HILDESHEIMER
INFORMATIK-
BERICHTE

Martin Schaaf & Klaus-Dieter Althoff (Hrsg.)

LWA 2006

Lernen – Wissensentdeckung – Adaptivität

9.–11.10.2006 in Hildesheim

ISSN 0941-3014
1/2006 (Oktober 2006)

Universität
Hildesheim

Institut für Informatik
Marienburger Platz 22
31141 Hildesheim

Aktualisierung September 2007

Vorwort

Die Workshop-Woche "Lernen, Wissen und Adaptivität 2006" (LWA 06) versteht sich als Forum, bei dem etablierte und neu auf einem Gebiet arbeitende Wissenschaftlerinnen und Wissenschaftler ihre aktuellen Arbeiten vorstellen und intensiv miteinander diskutieren können. Dies macht den besonderen Reiz dieser Veranstaltung aus, welche erstmals 1999 in Magdeburg stattfand. Der diesjährige Austragungsort war die Universität Hildesheim und wie in den Jahren zuvor wurden von verschiedenen Fachgruppen der Gesellschaft für Informatik e. V. (GI) eine Reihe interessanter Workshops organisiert. Dies waren im Einzelnen:

- FG-ABIS (Workshop der Fachgruppe „Adaptivität und Benutzermodellierung in interaktiven Softwaresystemen“)
- FG-IR (Workshop der Fachgruppe „Information Retrieval”)
- FG-KDML / AK-KD (Workshop der Fachgruppe „Knowledge Discovery, Data Mining und Maschinelles Lernen” sowie des Arbeitskreises „Knowledge Discovery”)
- FG-WM (Workshop der Fachgruppe „Wissensmanagement”).

Neben Einblicken in aktuelle Trends, Technologien und Anwendungen, die in den einzelnen Sitzungen geboten wurden, wurde auch der Austausch einzelner Fachgruppen entlang inhaltlicher Berührungspunkte intensiv betrieben. Zu diesem Zweck gab es einzelne Workshop-übergreifende Sitzungen, die durch zwei eingeladene Vorträge abgerundet wurden.

Wir wünschen den Teilnehmern der LWA Workshopwoche, dass die Vielzahl der angebotenen Vorträge eine gute Grundlage für Inspiration und Motivation zukünftiger Forschungsaktivitäten darstellt, sowie einen angenehmen Aufenthalt in Hildesheim.

Danksagung

Wir möchten zunächst ganz herzlich allen Workshop Organisatoren, Autoren und Gutachtern für ihren Beitrag zum Gelingen der LWA 06 danken. Die Workshops wurden dieses Jahr organisiert von

- Eelco Herder (L3S Research Center, Hannover), Dominik Heckmann (DFKI, Saarbrücken) (FG-ABIS)
- Alexander Hinneburg (Martin-Luther Universität Halle-Wittenberg), Andreas Hotho (Universität Kassel), Ralf Klinkenberg (Universität Dortmund) (FG-KDML / AK-KD)
- Thomas Mandl (Universität Hildesheim) (FG-IR)
- Alexandre Hanft, Martin Schaaf (Universität Hildesheim) (FG-WM)

Weiterhin geht unser Dank an den Herrn Präsidenten der Universität Hildesheim Prof. Dr. Wolfgang-Uwe Friedrich für die Eröffnung der LWA 06 und allen Kollegen und studentischen Mitarbeitern der Arbeitsgruppe „Intelligente Informationssysteme“ für ihren Einsatz und die Unterstützung vor Ort. Wir danken insbesondere Herrn Alexandre Hanft für das Zusammenstellen der Proceedings und Frau Martina Rosemeyer für ihren Einsatz bei der lokalen Organisation.

Der interdisziplinäre Aspekt der LWA wurde dieses Jahr nicht zuletzt durch zwei eingeladene Vorträge fokussiert, für die wir Herrn Dr. Ingwer C. Carlsen von Philips Research Europe in Hamburg sowie Herrn Dr. Joachim Baumeister von der Universität Würzburg gewinnen konnten. Auch ihnen möchten wir an dieser Stelle ganz herzlich danken.

Martin Schaaf & Klaus-Dieter Althoff
Hildesheim, im Oktober 2006

Inhaltsverzeichnis

Medizinische Bildverarbeitung - Bedarf an Modellierung und Adaptivität

I.C.Carlsen
Philips Research Europe - Hamburg

Workshop „Lernen, Wissen und Adaptivität“
09.-13.10.2006 Universität Hildesheim

Philips unterhält weltweit eine eigenständige Forschungsorganisation mit mehr als 2000 Angestellten. Die Forschungsthemen umfassen die Bereiche Medizintechnik, Life Style und Technologie. Forschung für die Medizintechnik spielt seit langem eine wichtige Rolle und erfährt gegenwärtig eine umfangreiche Erweiterung. Speziell die Philips-Forschung in Hamburg blickt auf eine mehr als 20-jährige Geschichte in der Verarbeitung und Analyse medizinischen Bildmaterials zurück.

Nach einer kurzen Einführung in die Forschungsorganisation von Philips wird ein Überblick über die Forschungsarbeiten im Hamburger Labor in den Bereichen

- multi-modale computergestützte Krebsdiagnostik,
- elastische Registrierung medizinischer Bilder,
- intelligente Akquisitionskontrolle in der Kernspinresonanztomographie,
- anatomische Modelle

gegeben.

In diesen Bereichen der Bildverarbeitung spielt klinisches Vorwissen in Form von Modellen eine zunehmend wichtige Rolle bei der Konfigurierung und Ablaufsteuerung. Anhand klinischer Beispiele wird der Bedarf an und der Einsatz von Vorwissen in der medizinischen Bildverarbeitung geschildert.

Knowledge Engineering in the Age of Communities

Joachim Baumeister
University of Würzburg
email: baumeister@ai-wuerzburg.de

Workshop „Lernen, Wissen und Adaptivität“
October 09-13, 2006 University of Hildesheim

For many years knowledge systems showed their usefulness in various business and academic domains. We report on some experiences we have made in own applications in medicine and biology. Due to the large body of necessary domain knowledge the systems were manually build by trained domain specialists. From these experiences an agile development process model was defined, since traditional knowledge engineering approaches failed or seemed to be not appropriate (e.g., small teams, motivating feedback, immediate and continuous quality assessment). We explain the building blocks of the agile methodology and we discuss when and why an agile development process can be more appropriate than traditional approaches.

Here, two important aspects are currently not considered: First, how to enable a distributed development within the methodology, and second, can knowledge engineering interfaces be simplified in a way such that general users can become ad hoc "knowledge engineers"? The success of Web 2.0 applications (e.g., Wikipedia, blogs, tag-based interactive applications) showed that general users are willing to actively participate in the creation of a new form of the WWW. As proposed by the Semantic Web vision the integration of knowledge-enabled services into web content will increase its utility. We discuss how knowledge systems can be combined with the concepts of Wikis, thus defining Knowledge Wikis.

A *KnowledgeWiki* retains the simple acquisition interface of traditional Wiki clones but extends its content types from text and multimedia to explicit knowledge fragments. We demonstrate a prototype system and show that a KnowledgeWiki intuitively provides the technology for a distributed knowledge engineering process through a standard web browser. We sketch some promising applications for Knowledge Wikis, and we give an outlook of interesting research directions arising from the implementation of Knowledge Wikis, e.g., the maintenance and evolution of heterogeneous knowledge (from text to rules), agile engineering techniques in a distributed environment, and advanced interfaces for ad-hoc knowledge engineers.

ABIS 2006 - 14th Workshop on Adaptivity and User Modeling in Interactive Systems

Eelco Herder
L3S Research Center
Hannover, Germany
herder@L3S.de

Dominik Heckmann
DFKI
Saarbrücken, Germany
heckmann@dfki.de

The ABIS Workshop

ABIS - *'Adaptivität und Benutzermodellierung in interaktiven Softwaresystemen'* - is the special interest group on Adaptivity and User Modeling in Interactive Systems of the German Computer Society.

The ABIS Workshop has been established as a highly interactive forum for discussing the state of the art in personalization and user modeling. Latest developments in industry and research are presented in plenary sessions, forums, and tutorials to discuss trends and experiences.

'Everyday Adaptivity'

This year's ABIS workshop is centered around the theme *everyday adaptivity*. In our modern times, it seems close to impossible to survive without a large variety of electronic devices and an Internet connection. As we are all individuals, we expect these devices and online services to serve our individual needs. Adaptable and adaptive systems can be seen the digital counterpart of the glove that molds into the form of one's hand.

System designers may want their users to initially marvel at the intelligence put in their systems. Eventually, it should be no more than logical, natural and obvious that the system is responsive to the users' preferences, interests, goals, needs and contexts. Adaptivity should become everyday and everywhere: personalized advice in online stores, individualized museum tours, personal agents that take over routine tasks and provide advice, adaptive learning systems, self-regulating household appliances.

Workshop Overview

The program committee received submissions from research and industry within the broad area of User Modeling and Adaptive Systems. Special emphasis of this year's workshop was on submissions in the following areas:

- design, implementation and/or evaluation of adaptive and personalized systems (e.g. mobile devices, e-learning, embodied agents, e-commerce, ambient intelligence)
- user modeling methods and techniques: social and collaborative approaches, usage mining, machine learning, reasoning mechanisms, evaluation techniques
- cross-sectional topics, such as security, privacy, trust, collaboration, adaptive system engineering

We strive to have a diverse group, varying from young researchers working on Master or PhD level to experts in the field. The workshop aims to provide a platform for exchanging fresh ideas and expertise, and for obtaining feedback on ongoing research. Therefore, we would like to encourage all participants to contribute to the workshop in discussions, so that all of us can enjoy another successful ABIS workshop.

Program Chairs

- Eelco Herder, L3S Research Center, Hannover, Germany
- Dominik Heckmann, DFKI, Saarbrücken, Germany

Program Committee

- Lora Aroyo, Eindhoven University of Technology, the Netherlands
- Mathias Bauer, DFKI, Saarbrücken, Germany
- Betsy van Dijk, University of Twente, the Netherlands
- Vania Dimitrova, University of Leeds, United Kingdom
- Nicola Henze, University of Hannover, Germany
- Andreas Jedlitschka, Fraunhofer IESE, Kaiserslautern, Germany
- Alexander Kröner, DFKI, Saarbrücken, Germany
- Tsvika Kuflik, Haifa University, Israel
- Andreas Lorenz, Fraunhofer Institut, Sankt Augustin, Germany
- Alexandros Paramythis, Johannes Kepler University, Linz, Austria
- Eric Schwarzkopf, DFKI, Saarbrücken, Germany
- Stephan Weibelzahl, National College of Ireland, Dublin
- Frank Wittig, SAP AG, Germany

Personal Reader Agent: Personalized Access to Configurable Web Services

F. Abel, I. Brunkhorst, N. Henze, D. Krause, K. Mushtaq, P. Nasirifard and K. Tomaschewski
Distributed Systems Institute, Semantic Web Group, University of Hanover
Appelstraße 4, 30167 Hanover, Germany
readerteam@kbs.uni-hannover.de

Abstract

The Personal Reader Framework enables the design, realization and maintenance of personalized Web Content Reader. In this architecture personalized access to web content is realized by various Web Services - we call them *Personalization Services*. With our new approach of Configurable Web Services we allow users to configure these Personalization Services. Such configurations can be stored and reused at a later time. The interface between Users and Configurable Web Services is realized in a Personal Reader Agent. This Agent allows selection, configuration and calling of the Web Services and further provides personalization functionalities like reuse of stored configurations which suit the users interests.

1 Introduction

Within the Personal Reader project we already developed Web Content Reader like the *Personal Publication Reader* [Baumgartner *et al.*, 2005] which allows browsing publications in an embedded context. We also utilized and extended the SWAD-E Semantic Portal software [Reynolds *et al.*, 2005] to provide a Personal Semantic Portal [Henze and Abel, 2005]. Whereas these approaches are fixed in terms of the type of data that is provided, we now introduce a more generic approach: *Configurable Web Services* and the *Personal Reader Agent*. The Personal Reader Agent is a Web Application which enables users to select, configure and call Configurable Web Services. These Semantic Web Services need a detailed description of how they can be configured and how they are accessible. According to this description the Personal Reader Agent generates an interface that allows to adjust the Web Services. Personalization functionalities, like reuse of stored configurations of Web Services which suit the users interest, lead to an adaptive, personal Agent.

2 Personal Reader Agent

The Personal Reader Agent is on the one hand a kind of wizard that allows to select, configure and call Configurable Web Services and on the other hand it enables users to manage and reuse their saved configurations (*personalized access support*).

2.1 Configurable Web Services

Each Configurable Web Service has a detailed RDF description which defines parameters that can be used to adjust the Web Service (*Configurable description*). An example is the *My Ear Music Web Service*: This service allows users to configure parameters like *music category* or *maximum duration of songs*. It results in a *podcasting feed* containing items that are aggregated from arbitrary feeds but fulfill the adjusted parameters. A formal definition of the Web Service's configurable parameters, thus a *Configurable Web Service*, has the advantage that the process of configuring the Web Service can be abstracted and different configurations can be stored and reused. These two aspects are covered by the Personal Reader Agent.

2.2 Demonstration

Within a normal workflow the Personal Reader Agent offers the following steps:

1. **Discovery and Selection:** In this step the Agent requests human readable descriptions of the Configurable Web Services that are registered at our UDDI. Afterwards these descriptions are prepared for a selection by the user.
2. **Configuration:** After the first step the Agent reads in the Configurable descriptions of the selected Web Services and generates HTML forms so that the user can perform the configuration (see figure 1).
3. **Web Service Call:** After all selected Web Services are configured without violating the restrictions defined in the corresponding Configurable descriptions (e.g. *maxNumberOfInputs*, *type*, ...), the Agent is ready to call the Web Services. In this step the user further has the opportunity to save the configuration (see figure 2).
4. **Presenting the results:** This step is not part of the Agent application but rather a task that can be done by a common RDF browser or an application that provides a special view for certain RDF data, e.g. *MyEar View* visualizes podcasting feeds (see figure 3).

Stored configurations and a corresponding user model build the fundament to enable users to:

- **reuse their own configurations:** In order to allow users a faster access to the Configurable Web Services they can call these services also with a saved configuration as illustrated in figure 4.
- **reuse recommended configurations of other users:** The Agent allows the listing of configurations that might be relevant for a user. To determine relevant configurations the Agent utilizes relations between users that are defined by an ontology that models persons and their involvements in working groups. If two

persons (*User A* and *User B*) are involved in the same working group then the Agent suggests that configurations made by *User A* are also interesting for *User B*.

3 Conclusion

In this demonstration paper we describe how the Personal Reader Framework has been extended by a Personal Reader Agent for configuring Web Services according to a user's needs. The Personal Reader Agent provides a user interface and corresponding functionality to discover, select, configure, call and access Web Services.
At present the Personal Reader Agent is deployed as a prototype accessible via http://www.personal-reader.de/agent/. More informations can be found at http://www.personal-reader.de/agent/documentation.pdf.

References

[Baumgartner *et al.*, 2005] R. Baumgartner, N. Henze and M. Herzog The Personal Publication Reader: Illustrating Web Data Extraction, Personalization and Reasoning for the Semantic Web. *European Semantic Web Conference ESWC 2005*, Heraklion, Greece, 2005.

[Henze and Abel, 2005] N. Henze and F. Abel. User Awareness and Personalization in Semantic Portals. *4th International Semantic Web Conference*, Galway, Ireland, 2005.

[Reynolds *et al.*, 2005] D. Reynolds, P. Shabajee, S. Cayzer and D. Steer. *Semantic Portals Demonstrator - Lessons Learnt*. SWAD-Europe deliverable 12.1.7, 2005.

A Figures

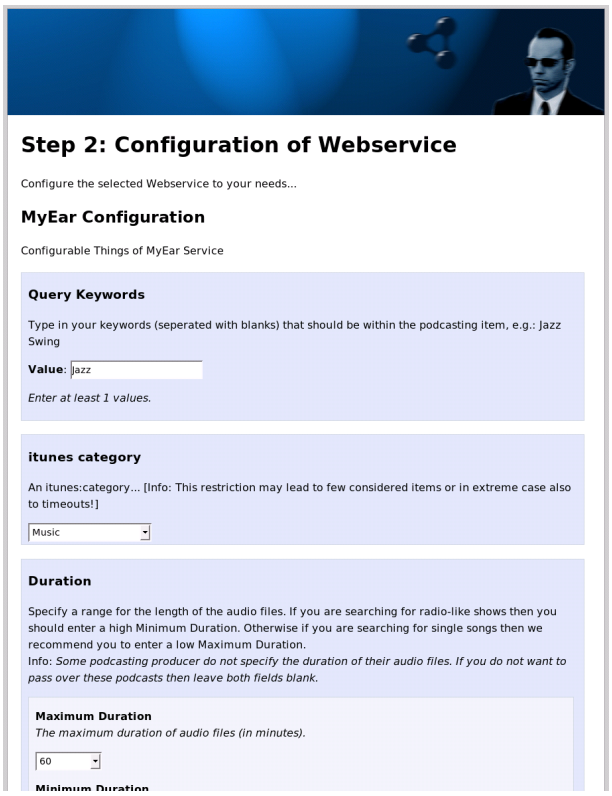


Figure 1: Configuration of Web Services

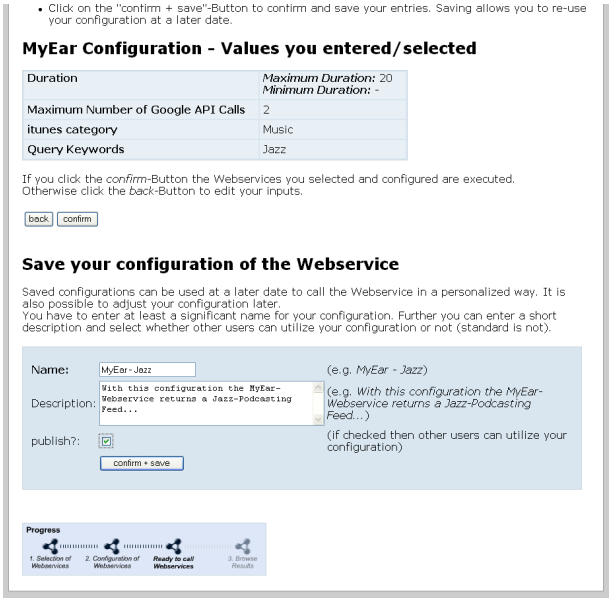


Figure 2: Saving Configurations - Entry of meta description about a configured Web Service

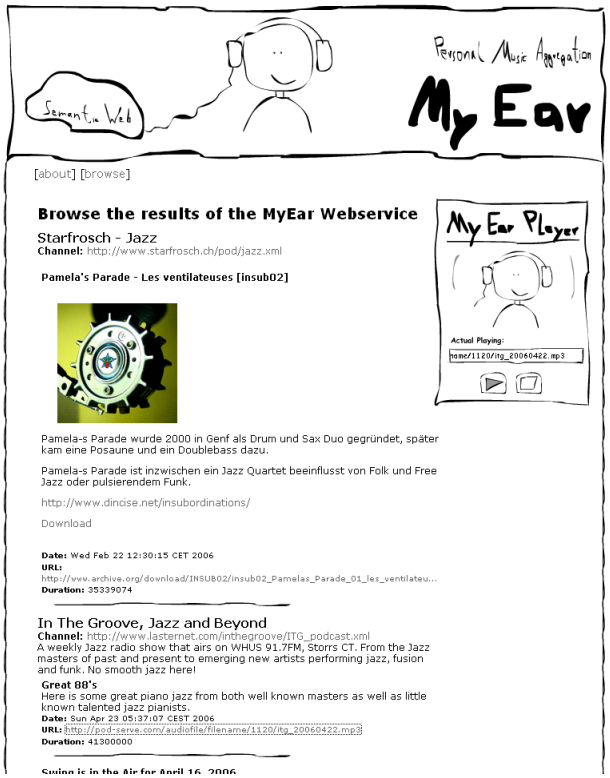


Figure 3: MyEar View

Figure 4: List of configured Web Services

User and Usage Profiling in a Multi-platform Service Environment

Armen Aghasaryan, Stéphane Betgé-Brezetz
Alcatel Research & Innovation, Marcoussis, France
armen.aghasaryan@alcatel.fr, stephane.betge-brezetz@alcatel.fr

Guillaume Raschia, Marc Gelgon
LINA Atlas-GRIM team, University of Nantes, France
guillaume.raschia@univ-nantes.fr, marc.gelgon@univ-nantes.fr

Abstract

In this paper, we present a beginning work on a behavioural profiling approach within a multi-service environment where the service usage and content consumptions data are collected on different service delivery platforms and/or on the user terminals. Therefore, it is necessary to be able to aggregate these potentially heterogeneous data so that a faithful user profile can be induced. The information structure used for profile data representation must be designed in a way to allow its utilization by a number of real-time multi-media personalized applications.

1 Introduction

The explosion of services offered by the telecommunications operators (voice services, video on demand, IPTV, etc.) and the diversity of access environments (terminals and networks) make more and more critical the capability of the operators to measure and to analyze in details the usage of the provided services by end-users. Indeed, the usage measurement allows the operators to be well informed on their best revenue generating services that ensure the success of their offer today and tomorrow.

In addition, a per user analysis aimed at the discovery of user's interest centres will also create the possibility for operators to propose more attractive applications adapted to the profile of each user. A good example is the targeted advertisement, a publicity selected for a particular user in accordance with his interest centers. While business analysts observe a certain saturation in the market of classical broadcast TV advertisement, the targeted ad is seen as a serious revenue relay for advertisers, see [Meyer, 2005].

A range of personalized applications have been successfully industrialized in the domain of Web technologies and e-commerce, e.g. Google AdWords, AdSense, or Amazon recommender system [Linden et al, 2003]. We note that these are "monolithic" solutions where personalization application and profile learning approach are very closely linked, or in other words, the profiling technology is dedicated to a particular personalization application.

In this context, the objective of our research project is twofold. First, we aim at extending the domain of profiling and personalization to the environments of next-generation network operators where multiple content and communications services co-exist. Second, we advocate a a generalized profiling approach where personalization and the profile learning techniques are clearly decoupled.

This paper is structured as follows. The section 2 presents the concept of a multi-platform / multi-application profiling module within a heterogeneous environment of modern large operators. In section 3, we describe the high level architectural approach. Finally, in section 4, both the user and content description as well as the scope of possible algorithmic approaches are presented.

2 Profiling in a multi-service environment

Telecom operators are now proposing content services (TV, music, …) in addition to their traditional communication services. Next stage will be the personalization of these services in order to better fit customer expectations and to get promising associated revenues. Various personalized services can be envisaged: targeted ad, personalized interactivity, personalized search, social networking applications, etc. Figure 1 illustrates such personalized applications in the context of IPTV (similar services can also be proposed in a mobile environment). At the initial step (a), the customers are watching a broadcasted TV program (all the viewers are watching the same program). Then, at the Ad time, each customer receives a targeted Ad depending on his profile: Alice - *who likes buying clothes!* - receives an Ad on dresses (b), John - *who likes cars!* - receives an Ad on cars (c). Furthermore, as shown in (b) and (c), interactivity menus can also depend on profiles.

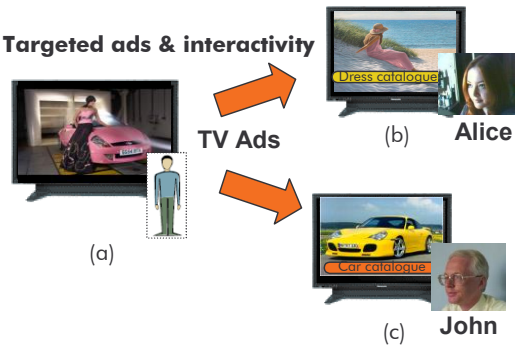


Figure 1 : Example of service personalization: targeted Ad and personalized interactivity on IPTV.

In order to propose such personalized services, operators need to have a reliable user profile, faithful to the actual user interests and expectations. This is the purpose of the multi-platform / multi-application user and usage profiling module that builds and updates the user profile taking into account the usage data on different services.

The principle of this module is shown in Figure 2. On one hand, it must integrate usage information of different Service Delivery Platforms (SDP): IPTV SDP for TV/video on fixed networks (e.g. xDSL), mobile video SDP for TV/video on mobile networks (e.g. either streamed on 3G or broadcasted on DVB-H networks) and IMS SDP (IP Multimedia Sub-system) for the next generation of communication networks. On the other hand, this generic profiling module should offer semantic user profile information allowing to address the various types of applications.

Figure 2. A multi-platform / multi-application profiling module.

In the sequel, we present the architecture of this module and the algorithmic approaches that can be used to aggregate the usage information into a consistent user profile.

3 Architectural approach for generic user profiling

3.1 Functional blocks

The overall architecture of the profiling module is depicted in Figure 3. As it was mentioned earlier, we advocate an approach where the personalization and the profile learning are clearly dissociated.

At first, we assume that multiple sources of raw data are used : server data (e.g. CDR – Call Data Records) coming from the different SDP (IPTV, mobile video and IMS), or data extracted from network devices, terminals or Set-Top-Boxes (e.g. cookies). Example of such raw data can be: the trace of a user session who has viewed *during 10mn a sport video on his TV*, or the trace of channel switching extracted from the DLSAM equipment.

The main components of the proposed architecture are the following (c.f. Figure 3):

- **User Profile Data-Base** that stores user profile information. In the scope of this paper, we are concerned with semantic user data such as the user interest domains (sports, entertainment, etc.), and his service habits (high/low user of video, service habits at home or on the move, etc.).
- **Explicit Profiling** allowing for the end-user to enter directly his user profile information (through a web site or a questionnaire). Explicit profiling is necessary, however – even if the end-user enters fairly these information it is not sure that it corresponds to the actual profile of the user. Therefore, having also an implicit profiling is mandatory.
- **Implicit Profiling** allowing to gather all the raw data, to learn and to update the user profile with respect to the real usage of services. Combination of such explicit and implicit profiling will allow the operator to have the most faithful and up to date profile.
- **Query Interface** used by the applications to offer customized services. Although the specific logic of personalization will be realized within each particular application, some frequent request patterns can be provided by the profiling module via its intelligent query interface. So, depending on the applications, various requests should be envisaged, e.g.: get all the users having certain profile criteria (e.g. interest in tennis, high video user), get the users having a profile "similar" to another one. This query interface (and the offered API) is critical as it will ensure the independence of the profiling module with respect to different personalized applications.

Note that all the SDPs as well as the profiling module are supposed to be owned by a single large operator. In the case where multiple operators are involved a profile sharing mechanism should be introduced, but this is out of the scope of this paper.

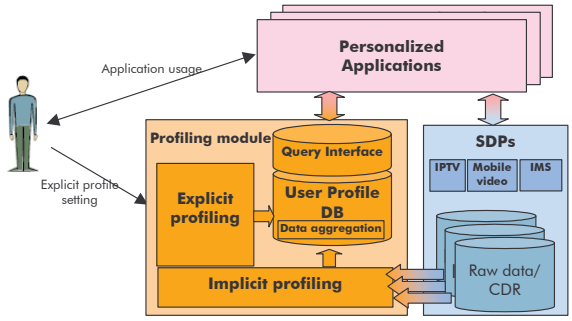


Figure 3. Main functional blocks for user profiling and personalization of services.

3.2 Non-Functional requirements

While defining a system of service usage measurement, an analysis of operational constraints of security, confidentiality, privacy, and scalability shall be taken into account especially for real-time multi-media applications. For example, the following aspects can be mentioned:

- limitations in terms of confidentiality guarantees, or the rights of the operator on the usage data of the customers. The related privacy legal constraints (national, European or international) have therefore to be guaranteed,
- capacity to support the profile evolution,
- performance issues related to temporal and spatial complexity and to the profile construction algorithms,
- possibility of evolution towards a distributed architecture for data acquisition, profile learning and service personalization.

4 Algorithmic profiling approaches

4.1 User profile and content description

The ETSI work on user profile management (ETSI STF265 EG 202 325 Document, oct. 2005) enumerates the very many aspects of user profile management entertaining to personal devices. As an example, it raises issues specific to the distributed infrastructure we are aiming at, such as the conflict of interest on how to disclose as little as possible from both the user profile and the content potentially delivered. Although user profile learning from their actions is out of scope of this ETSI standard, this issue can relate technically to how we model the user profile describing the user's interests.

Standardization has since long addressed the description of audiovisual content (programs and commercials)

[Evain *et al*, 2003.], including features such as 'genre' and 'channel' which are most amenable to learning user interests. It now also includes an MPEG-7 description form for users' interests and the history of user viewing actions, designed with similar goal as we focus on.

Descriptions conforming to this standard have been used in [Thawani *et al*., 2004] to suggest ads suited to a user profile. The matching criterion remains however limited to keywords, whereas the proposal involves a more powerful generalization/induction mechanism.

Hence, in our proposal, the user profile model is defined over the semantic compression technique proposed in [Saint-Paul *et al*., 2005]. Such an approach is somehow similar to that of [Pigeau, *et al*, 2003] applied to TV recommender systems.

4.2 Profiling algorithms

Specific requirements of the multi-platform service environment are addressed to propose a novel and well-suited approach to user profiling.

The main idea is to compute a classification tree of content services from their metadata and then, to mark subtrees corresponding to consumed services by a given user. Both tasks perform online and profiles are incrementally maintained. A conceptual clustering algorithm build concepts (or summaries) that represent classes of services thanks to their metadata. Profiles are then defined as a ranked collection of summaries associated to classes of services. Ranks are calculated as aggregated preferences taken from usage information. And the more the summary is high in the tree, the more it provides general information about the user profile.

Furthermore, using a fuzzy-set based compression technique, our approach relies on background knowledge built from an user-defined vocabulary with a high semantic level. Indeed, the fuzzy set theory provides a symbolic/numerical interface that leads to a numerical computation and a human-friendly interpretation of data structures. Each node from the classification tree is described by a set of linguistic labels on metadata attributes. Then it offers a straight way to query services from personalized applications. Such a querying mechanism has been defined in [Voglozin *et al*., 2006]. In the same time, this summary model allows the end-user to provide by hand an explicit profile with its own vocabulary. The system incorporates explicit summaries into the (implicit) list and resolves inconsistency following a given policy such as "explicit first" or "implicit first" or "middle-term".

Finally, such a system is expected to fulfil both the functional and non-functional requirements as stated in section 3: a content-based profiling approach thanks to services metadata and usage information, a joint explicit and implicit profile, a personalized access to services provided by a profile-oriented filtering mechanism, an online update of profiles, a linear temporal complexity of the algorithm w.r.t. metadata as well as a controlled memory footprint, and last, the possibility to deploy the solution onto distributed architectures (see [Saint-Paul *et al*., 2005] for details about complexity and distribution).

Technical and legal precautions should be considered to conform to privacy constraints, even if we claim that they do not involve scientific problems. However, privacy issues arising from building user profiles of TV viewers have recently been surveyed in [Spangler *et al*., 2006], especially when data mining is employed to analyse these profiles for marketing purposes.

5 Conclusion

In this paper, we have presented the architecture of a multi-platform/multi-application profiling module integrating heterogeneous service usage information and enabling various personalized applications (targeted ad, personalized search, social-based networking applications, etc.). Telecom operators should take benefit of this profiling module to offer these next generation of revenue generating applications.

However, several challenges have yet to be addressed before the deployment of this system in an operational environment. There are technical challenges, e.g. challenges related to the complexity and the exactitude of the profiling algorithms, their capacity to reflect faithfully the real interest centers of an user, or the possibilities of distributing these algorithms closer to the end-user for scalability purposes. But, other challenges are related to the human acceptance of such profiling and its intrusiveness in the private life of the customers. Even if privacy constraints are guaranteed, this dimension has to be taken into account in the very early design of both the profiling module and of the personalized applications. Nevertheless, one can observe that – even if all the users may not accept a deeper profiling - today they are more and more open to provide personal information on the web. Moreover, operators should also propose incentives for the users to be profiled (e.g. service discount). So, the latter should be also interested by more personalized services in return of their authorization to be profiled.

References

[Evain *et al*, 2003.] TV-Anytime Phase 1, J-P. Evain and H.Murret-Labarthe, EBU Technical Review, July 2003.

[Linden *et al*., 2003] G. Linden, B. Smith, and J. York, Amazon.com Recommendations, Item-to-Item Collaborative Filtering, Industry Report Amazon.com, 2003.

[Pigeau, *et al*., 2003] A. Pigeau, G. Raschia, M. Gelgon, N. Mouaddib, R. Saint-Paul, A Fuzzy Linguistic Summarization Technique for TV Recommender Systems, The IEEE International Conf. on Fuzzy Systems, 2003.

[Meyer, 2005] L. Meyer, Television 2015 - The future of TV financing in Europe, IDATE, DigiWorld Analysis, 2005.

[Saint-Paul *et al*., 2005] R. Saint-Paul, G. Raschia, N. Mouaddib, General Purpose Database Summarization, in VLDB'05, August 30-September 2 2005.

[Spangler *et al*., 2006] W. Spangler, K. Hartzel, M. Gal-Or, Exploring The Privacy Implications Of Addressable Advertising And Viewer Profiling, Communications of the ACM, May 2006.

[Thawani *et al*., 2004] A. Thawani, S. Gopalan, V. Sridhar, Context Aware Personalized Ad Insertion in an Interactive TV Environment, TV'04, Eindhoven, 2004.

[Voglozin *et al*., 2006] W.A. Voglozin, G. Raschia, L. Ughetto and N. Mouaddib. Querying a Summary of Database, Journal of Intelligent Information Systems (JIIS), Volume 26(1):59-73, January 2006.

A Personalization Service for Curriculum Planning

M. Baldoni [2], C. Baroglio [2], I. Brunkhorst [1], N. Henze [1], E. Marengo [2], V. Patti [2]

[1] L3S Research Center
University of Hannover
D-30539 Hannover, Germany
brunkhorst@l3s.de
henze@l3s.de

[2] Department of Computer Science
University of Torino
I-10149 Torino, Italy
matteo.baldoni@di.unito.it
cristina.baroglio@di.unito.it
viviana.patti@di.unito.it

Abstract

In this work we describe a "semantic personalization" web service for curriculum planning. Based on a semantic annotation of a set of courses, provided by the University of Hannover, reasoning about actions and change —in particular classical planning— are exploited for creating personalized curricula, i.e. for selecting and sequencing a set of courses which will allow a student to achieve her learning goal. The specific student's context is taken into account during the process: students with different initial knowledge will be suggested different solutions. The Curriculum Planning Service has been integrated as a new plug-and-play personalization service in the Personal Reader framework.

1 Introduction

In this work we describe the integration of a *Curriculum Planning Service*, for building personalized paths in a space of semantic learning resources —university courses— as a *plug-and-play personalization service* in the Personal Reader (PR), a framework for designing, implementing and maintaining personalization services. Each personalization service offers some personalization functionality, e.g. recommendations tailored to the needs of specific users, pointers to *related / interesting / more detailed / more general* information, and so on. Thus, users receive personalized views on Web contents. The characteristic of the PR framework is that it treats a personalization functionality as a *service*, and, within the framework, a user can select and combine —plug together— which personalized treatment he or she wants to receive. The user interface of the PR framework has to adapt to the selected functionality and display the results of the personalization services in a device-dependent manner. This framework has already been used for developing Web Content Readers that present online material in an embedded context, i.e. the Personal Publication Reader [Baumgartner *et al.*, 2005] and the Personal Reader for e-learning [Henze, 2005].

Recently the PR framework has been extended with the introduction of *configurable web services* and the *Personal Reader Agent* [Abel *et al.*, 2006]. These new services can process RDF data, and provide RDF in return. Additionally, they have an OWL-S description, and provide information on how they can be configured and in which way they can be accessed. An agent is used to provide an interface to manage the configurations and adapt them according to the users interests.

In this paper, we describe the integration of a new personalization service into the PR framework which uses *reasoning on semantically annotated data* about courses held at the University of Hannover, for enabling a personalization functionality, i.e. planning curricula, that are personalized w.r.t. the student's context and learning goal. The idea is to embed into this new service a reasoner, which is realized by means of actions techniques [Baldoni *et al.*, 2004a]. The new service is activated on demand by a component of the PR by means of a proper request, which includes the user's learning goal (expressed as a set of knowledge entities taken from a shared ontology) the user's context, i.e. what the user (supposedly) already knows. Moreover, since the reasoner applies planning techniques for performing the sequencing, we provided a semantic annotation of the set of courses with preconditions and effects. In fact, in the spirit of [Baldoni *et al.*, 2004a], we interpret each course as an atomic action, on the basis of prerequisites (what the student should know for understanding the course contents) and effects (what the student is supposed to learn by attending the course). Given such input data, the Curriculum Planning Service returns to the PR a set of possible personalized curricula, i.e. a set of linear plans. Then the PR is in charge to present these plans to the user as personalized sequences of courses to attend for reaching the desired learning goal.

The next section describes our approach and the components we have developed in order to implement the new personalized Curriculum Planning Service. Section 3 shows an example session with a simple demonstrator application. Conclusions follow.

2 Personalized Service for Curriculum Planning

In order to build the new personalization service for curriculum planning, multiple steps are necessary.

2.1 Extraction and Preparation of Metadata

Concerning the extraction of metadata, we used the Lixto [Baumgartner *et al.*, 2001] tool for extracting metadata about the available courses at the University of Hannover from the public HIS-LSF web pages (HIS-LFS provides infrastructure for managing all information relevant to higher education). This approach is similar to the one used in the *Personal Publication Reader* [Baumgartner *et al.*, 2005]. Other possible approaches include direct access to the back-end database which is providing the course information, or access to other repositories like e.g. Learning Management Systems. Unfortunately, the quality of most

of the information in these databases turned out to be insufficient, and many inconsistencies in the description of prerequisites and effects of courses could be noticed by analyzing the extracted data. As a consequence, we focussed on a subset of courses (computer science and engineering courses), and manually post-processed the data. Courses are annotated with prerequisites and effects, that can be seen as knowledge concepts or competences, i.e. ontology terms. After automatic extraction and manual post-processing, prerequisites and effects of each course were translated into English, a necessary step to combine it with material from our partners in the NoE REWERSE[1]. Metadata about credits, location and time schedule of courses were also automatically extracted. All the metadata is stored in an RDF document.

2.2 Reasoning on Metadata

Given a semantic annotation with preconditions and effects of the courses, classical planning techniques are exploited for creating personalized curricula, in the spirit of the work in [Baldoni *et al.*, 2004a; 2004b]. The curriculum planning task is accomplished by a reasoning engine, which has been implemented in SWI Prolog[2].

The interesting thing of using SWI Prolog is that it contains a semantic web library allowing to deal with RDF statements. Since all the inputs are sent from the PR to the reasoner in a *RDF request document*, it actually simplifies the process of interfacing the planner toward the Personal Reader. In particular the request document contains a) links to the RDF document containing the database of courses, annotated with metadata, b) the user's context c) the user's actual learning goal, i.e. a set of knowledge concepts that the user would like to acquire, and that are part of the *domain ontology* used for the semantic annotation of the actual courses. The reasoner can also deal with information about credits provided by the courses, when the user sets a credit constraint together with the learning goal.

Given a request, the reasoner runs the Prolog planning engine on the database of courses annotated with prerequisites and effects.

The initial state is set by using information about the user's context, which is maintained by the User Modeling component of the PR. In fact such user's context includes information about what is considered as already learnt by the student (attended courses, learnt concepts) and such information is sent to the planner as a part of the request document. The planning process is driven by the user's learning goal (and possibly by credit constraints), which is again provided by the PR in the request document. The Prolog planning engine has been implemented by using a classical depth-first search algorithm [Russell and Norvig, 1995]. This algorithm is extremely simple to implement in declarative languages as Prolog.

At the end of the process a *RDF response document* is returned as an output. It contains a list of plans (sequences of courses) that fulfill the users learning goals and profile. The maximum number of possible solutions to compute can be set by the user in the request document. Notice that further information stored in the user profile maintained by the PR could be used at this stage for sorting the list of plans with the aim of adapting the presentation of the solutions.

[1] `http://www.rewerse.net/`

[2] `http://www.swi-prolog.org/`

2.3 Embedding the Reasoner in a Web Service

In order to integrate the Curriculum Planning Service as a plug-and-play personalization service in the PR architecture we worked at embedding the Prolog reasoner into a web service. Figure 1 gives an overview over the components in the current implementation. The web service implements the Personalization Service (*PService* [Henze and Kriesell, 2004]) interface, defined by the Personal Reader framework, which allows for the processing of RDF documents and for inquirying about the services capabilities. The Java-to-Prolog connection runs the SWI-Prolog executable in a sub-process; essentially it passes the RDF document containing the request *as-is* to the Prolog system, and collects the results, already represented as RDF. While the request and response documents are transferred between the different components, these documents contain references to external resources, which are managed by other services in the framework, here the User Model and the Database of University Courses. The service itself can be accessed directly, as shown in Section (3), as well as being integrated as a PService into the Personal Reader Framework.

3 Demonstration

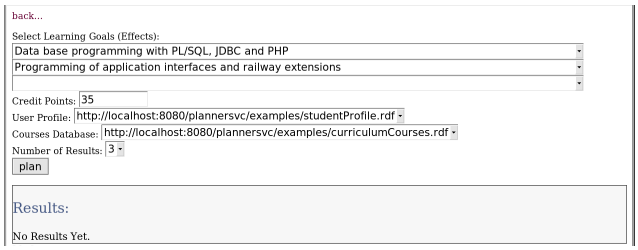


Figure 2: Starting page with two goals selected

As a proof-of-concept, we created a simple Visualization Servlet, which can be extended to become a Visualization Service (*VService* [Henze and Kriesell, 2004]) for the Personal Reader Framework.

Figure 2 shows a simple html form which allows the selection of learning goals as input for creating the curriculum sequences. Pressing the submit button "plan" sends a request to the servlet powering this interface, and an RDF request document will be created which then will be used to invoke the web service. The input data consists of 65 courses with 390 effects and 146 preconditions.

The returned RDF Response is parsed by the Servlet, and —in this prototype— a very simple List of Courses is displayed which fulfills the given goals.

4 Conclusion and Further Work

In this work we have shown the integration of a new semantic personalization web service for curriculum planning within the Personal Reader Framework. The goal of personalization is to create sequences of courses that fit the specific context and learning goal of individual students. Despite some manual post-processing for fixing inconsistencies, we used real data from the Hannover University database of courses.

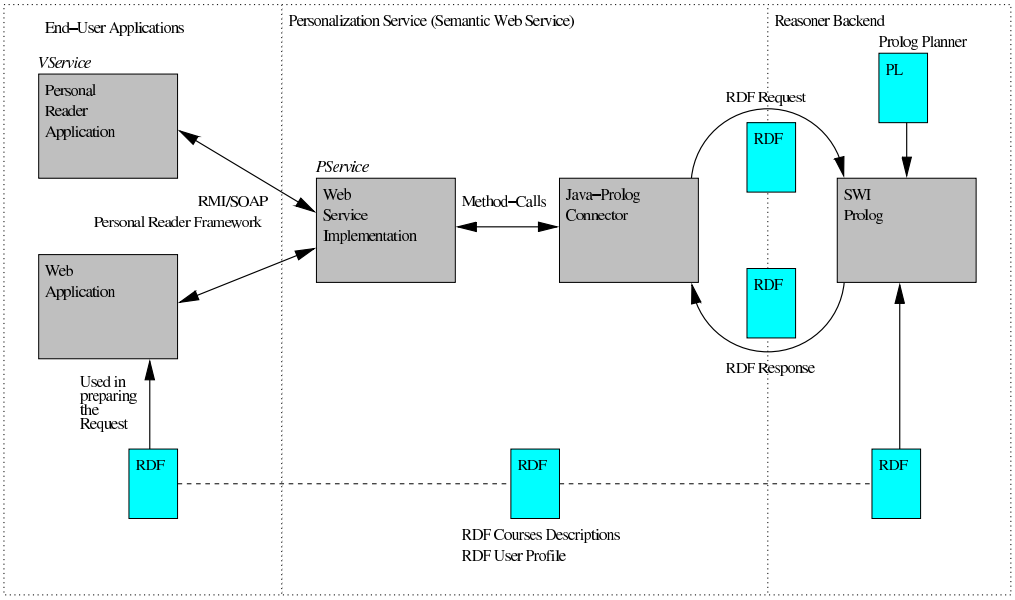


Figure 1: Curriculum Planning Web Service


Results:

Processing Time: 1133ms

Solution 1

- http://localhost:8080/plannersvc/examples/curriculumCourses.rdf#Complexity of algorithms
- http://localhost:8080/plannersvc/examples/curriculumCourses.rdf#Database systems I
- http://localhost:8080/plannersvc/examples/curriculumCourses.rdf#Database systems IIa
- http://localhost:8080/plannersvc/examples/curriculumCourses.rdf#Seminar to database systems
- http://localhost:8080/plannersvc/examples/curriculumCourses.rdf#Database systems IIb
- http://localhost:8080/plannersvc/examples/curriculumCourses.rdf#Datenbankpraktikum

Solution 2

- http://localhost:8080/plannersvc/examples/curriculumCourses.rdf#Complexity of algorithms
- http://localhost:8080/plannersvc/examples/curriculumCourses.rdf#Database systems I
- http://localhost:8080/plannersvc/examples/curriculumCourses.rdf#Database systems IIa
- http://localhost:8080/plannersvc/examples/curriculumCourses.rdf#Database systems IIb
- http://localhost:8080/plannersvc/examples/curriculumCourses.rdf#Datenbankpraktikum

Solution 3

- http://localhost:8080/plannersvc/examples/curriculumCourses.rdf#Complexity of algorithms
- http://localhost:8080/plannersvc/examples/curriculumCourses.rdf#Database systems I
- http://localhost:8080/plannersvc/examples/curriculumCourses.rdf#Database systems IIa
- http://localhost:8080/plannersvc/examples/curriculumCourses.rdf#Database systems IIb
- http://localhost:8080/plannersvc/examples/curriculumCourses.rdf#Seminar to database systems
- http://localhost:8080/plannersvc/examples/curriculumCourses.rdf#Datenbankpraktikum

Figure 3: Simple List of Results

The Curriculum Planning Service can be useful in many practical contexts. Exchanging Courses, and taking courses at different Universities becomes more and more common in Europe. Universities try to reduce costs and to cooperate in designing and integrating curricula. Through the Bologna Process initiative, the European Community aims at harmonizing the academic careers across Europe and curricula integration[3]. Curriculum planning might become a complicated task for students, who must build a reasonable path trough an enormous set of courses across the European countries. Especially in this scenario, there are further factors that the Curriculum Planning Service could take into account by exploiting further metadata concerning time and location of courses, that are actually already available in the course database as room-numbers, addresses and teaching hours. Such metadata could be used by the reasoner, besides the learning prerequisites and effects, in order to find a proper sequence of courses that is personalized also w.r.t. user's characteristics concerning time and location. We are actually investigating how to extend the application with a module of geo-spatial reasoning working on meta-data like floor-plans and locations.

More information about the Personal Reader can be found at the Homepage at `http://www.personal-reader.de/`, as well as for the Agent, located at `http://www.personal-reader.de/agent/` The Curriculum Planning Demonstrator is available at `http://semweb2.kbs.uni-hannover.de:8080/plannersvc`.

[3] `http://europa.eu.int/comm/education/policies/educ/bologna/bologna_en.html`

5 Acknowledgement

This work has been partially supported by the European Network of Excellence "REWERSE - Reasoning on the Web with Rules and Semantics".

References

[Abel *et al.*, 2006] Fabian Abel, Ingo Brunkhorst, Nicola Henze, Daniel Krause, Kai Mushtaq, Peyman Nasirifar, and Kai Tomaschweski. Personal reader agent: Personalized access to configurable web services. Technical report, Distributed Systems Institute, Semantic Web Group, University of Hannover, 2006.

[Baldoni *et al.*, 2004a] Matteo Baldoni, Cristina Baroglio, and Viviana Patti. Web-based adaptive tutoring: An approach based on logic agents and reasoning about actions. *Artificial Intelligence Review*, 1(22):3–39, September 2004.

[Baldoni *et al.*, 2004b] Matteo Baldoni, Cristina Baroglio, Viviana Patti, and Laura Torasso. Reasoning about learning object metadata for adapting SCORM courseware. In L. Aroyo and C. Tasso, editors, *Int. Workshop on Engineering the Adaptive Web, EAW'04: Methods and Technologies for Personalization and Adaptation in the Semantic Web, Part I*, pages 4–13, Eindhoven, The Netherlands, 2004.

[Baumgartner *et al.*, 2001] Robert Baumgartner, Sergio Flesca, and Georg Gottlob. Visual web information extraction with lixto. In Peter M. G. Apers, Paolo Atzeni, Stefano Ceri, Stefano Paraboschi, Kotagiri Ramamohanarao, and Richard T. Snodgrass, editors, *VLDB*, pages 119–128. Morgan Kaufmann, 2001.

[Baumgartner *et al.*, 2005] Robert Baumgartner, Nicola Henze, and Marcus Herzog. The personal publication reader: Illustrating web data extraction, personalization and reasoning for the semantic web. In Asunción Gómez-Pérez and Jérôme Euzenat, editors, *ESWC*, volume 3532 of *Lecture Notes in Computer Science*, pages 515–530. Springer, 2005.

[Henze and Kriesell, 2004] Nicola Henze and Matthias Kriesell. Personalization Functionality for the Semantic Web: Architectural Outline and First Sample Implementation. In *1st International Workshop on Engineering the Adaptive Web (EAW 2004)*, Eindhoven, The Netherlands, 2004.

[Henze, 2005] Nicola Henze. Personal readers: Personalized learning object readers for the semantic web. In *12th International Conference on Artificial Intelligence in Education, AIED05*, Amsterdam, The Netherlands, 2005.

[Russell and Norvig, 1995] Stuart Russell and Peter Norvig. *Artificial Intelligence: A Modern Approach.* Prentice Hall, 1995.

From Personal Memories to Sharable Memories

Nathalie Basselin and Alexander Kröner
German Research Center for Artificial Intelligence
Stuhlsatzenhausweg 3, 66123 Saarbrücken
{firstname}.{lastname}@dfki.de

Abstract

The exchange of personal experiences is a way of supporting decision making and interpersonal communication. In this article, we discuss how augmented personal memories could be exploited in order to support such a sharing. We start with a brief summary of a system implementing an augmented memory for a single user. Then, we exploit results from interviews to define an example scenario involving sharable memories. This scenario serves as background for a discussion of various questions related to sharing memories—and potential approaches to their solution. We especially focus on the selection of relevant experiences and sharing partners, sharing methods, and the configuration of those sharing methods by means of reflection.

1 Introduction

The tremendous growth of social software and associated concepts (from blogs to collaborative tagging and recommendation to reputation systems) demonstrates that people are willing to share *personal experiences*. In parallel, the huge number of websites offering forums, customer reviews, and customer-based recommendations proves the need to find *independent information*.

However, these technologies require people to spend efforts on reporting experiences, which is (beneath other issues such as privacy) one of the reasons why only a selected subset of experiences is shared this way. Another drawback of these common approaches to sharing is the lack of availability and context-awareness, which prevents their application for proactive and situated user support

These issues could be addressed by a mobile assistant, which supports the user in sharing personal information and in retrieving independent and relevant information. However, this raises lot of related research issues, for instance, the access and presentation of others' memories, or privacy issues. Our experience and a large-scale user study with a personal memory assistant offer some hints regarding these questions. In this paper we describe this assistant and report about our early reflection on how to extend this assistant for memory sharing: beneath an application scenario proposal for memories sharing, we describe our approach to solve issues such as the retrieval of relevant experiences in other people's memories, the selection of sharing partners, the handling of sharing occasions, and their exploitation for improving the system behavior.

2 Augmented Personal Memories

In SPECTER (cf. [Kröner et al., 2006a]) we conducted research on how augmented personal memories can be exploited for user modeling and decision support. The memories are created from a dense log of user experiences captured by an intelligent environment. Here, we think of an experience as an action, the context where this action took place, and annotations attached by user and system.

2.1 Building Personal Memories

The experience log is the result of an abstraction process, which begins on the level of sensor data. SPECTER may be connected to diverse sensors in order to capture information about the user's state and context. We experimented with a combination of GPS, IR (location tracking), biosensors (user feedback), web services (product-related services), and RFID (location tracking, smart objects). For a limited time, perceptions provided by these physical and virtual sensors are held in SPECTER's *short-term memory*, where inference processes and plan recognition are used to create a model of the user's current context.

In addition, all information gathered by the system is stored in a *long-term memory*, where machine-learning is applied in order to build a user model from behavioral patterns. The long-term memory provides beneath a plain record of perceptions an event-based organization, which combines each observed user action with its context. This so-called personal journal serves in the first place as "experience record" for the user, and is therefore an integral part of the user interface.

2.2 Accessing Personal Memories

The captured information about the user's activities is accessible to the user via diverse types of memory views.

The *chronological event list* (see the left-hand side of Figure 1) displays each observed user action in its context (e.g., place and time). The user can annotate each event with a written comment or adjust ratings (e.g., about quality) assigned by the system based on the user model.

An *object-oriented view* focuses on the recorded information without its context. It is typically applied to display query results about resources (e.g., products, places) and to exploit these for further application—e.g., for preparing a set of "examples" based on the memories, which may then be forwarded to services implemented by the environment (see the right-hand side of Figure 1).

Finally, a *function-oriented view* offers contextual functions for resources such as persons, objects, and locations stored in the memory. These functions make use of the

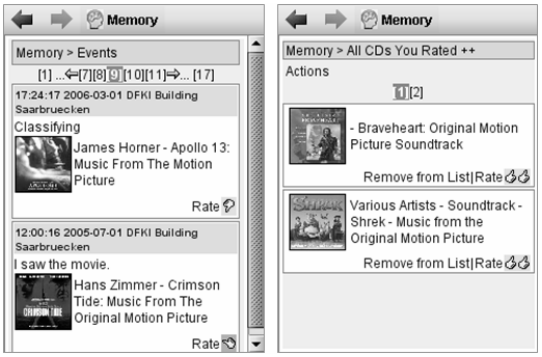


Figure 1: On the left-hand side: event-oriented view; on the right-hand side: object-oriented view.

memory (e.g., allow to retrieve objects or events related to some resource) and allow the user to exploit the current environment (e.g., allow to set up a query for similar products in the current shop). A typical dialog between user and system often involves several of these views. For instance, the event view grants access to the function view for objects involved in events, which allows setting up object selections displayed in the object view.

2.3 Decision Support

In order to describe the specific decision support provided by SPECTER, we coined the notion "Recomindation". This new paradigm for the exploitation of augmented memories blends "recommendation" and "reminder". "Recomindation" functions make use of the user's past experiences, of the current context, and of similarity algorithms to provide recommendations whose relevance is explained by the user's personal past experiences. For instance, when the user enters a CD store, the system offers among others the list of CDs she likes that are available in the store. Also when the user is looking at a CD, she can get a list of similar CDs that she knows. That way, she can remember similar CDs that she would have forgotten or learn more about an unknown CD (see Figure 2).

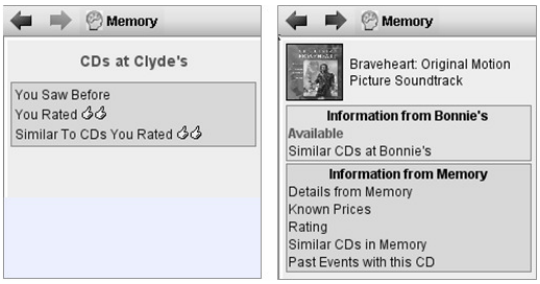


Figure 2: On the left: proactive situated services offer, when entering a store; on the right: proactive situated services offer, when looking at a CD in a store.

2.4 Reflection

Reflection on events recorded in the long-term memory allows the user to review past experiences, e.g., in order to prepare herself and/or the system for future actions. Guided by the system, the user adapts the system's decision support functions, for instance by customizing situational service triggers, or by correcting assumptions made by the system in the user model. In addition, the system offers automatically generated summaries of past actions. This aims at bringing elements (e.g., objects, locations) referenced by these events back into the user's mind and at assisting in their exploitation by offering services available on the Web (e.g., acquisition of extensive product reviews).

2.5 Evaluation

We used a CD shopping scenario for a summative evaluation study of the main aspects of our personal memory assistant: capture, storage and data presentation, exploitation, and control. We conducted the study with 20 participants in mock-up CD stores. Overall, participants were satisfied with the tested prototype and with the functions based on augmented memories; for a detailed description of the results, see [Plate *et al*., 2006].

Since the scenario of the study addressed a specific domain, we wanted to know if memory support was improving the user experience in this domain, and which other activities of the everyday life could benefit from augmented memories. Therefore, we asked participants to imagine scenarios where they would appreciate the functions of augmented memories; in addition, we asked for ideas and opinions regarding the sharing of information maintained by a SPECTER-like system. Most of the scenarios they imagined regarding sharing were shopping or tourism-oriented: "I am entering a bookstore. I would like to know the bestsellers as well as some people's opinions if I am interested in a given product. If I hesitate to buy a book, I might ask a friend who has tastes similar to mine."; "I would like to know if a given product is cheaper elsewhere."; "I would like to be warned when I'm about to buy a product which dissatisfied most people."; "I'm sightseeing but I don't know which places I should visit, whether this museum is worth its 25 euros entry fee, or how this hotel is.".

A shopping scenario is indeed a domain where experiences sharing functions are promising: the number of online customer reviews and forums about products proves that people are willing to learn about other people's opinions and experiences. In the scenarios they described, participants mentioned also several times that they are willing to know friends' opinions ("I might ask a friend who has tastes similar to mine."). Such functions could be provided by the sharing of user models and experiences of known people. However, shopping is quite limited regarding the kind of content shared: it consists mainly in sharing products attributes and their associated annotations. We therefore considered moving to a "shopping and cooking for guests" scenario including both grocery shopping and cooking in an instrumented kitchen. Since cooking involves recipes, i.e. processes, the sharing mechanisms will be more complex, as episodes, and not only perceptions, will have to be shared.

3 Towards Sharing Personal Memories

In the study, the information about the CDs' availability in the current mock-up store was provided by the CD store database; the similarity mechanism required for the "recomindation" functions consisted in calling the Amazon Web service corresponding to the function "Customers who bought this album also bought…". However, such information could be provided independently by a memory sharing mechanism. Someone looking at a product could access others' experiences to compare prices, to get customer opinions, or suggestions of alternative products.

Information provided by memory sharing is independent and not limited to the existing Web services. In addition, if subjects considered "recomindation" functions as time and money saving in our study, then we can expect users to find added value in querying others' past experiences since memory sharing has the potential to offer services like the ones mentioned above.

In other words, memory sharing has the potential to become a new medium for information exchange, which may complement traditional forums and online customer reviews. One advantage over those media could be the easiness to publish experiences. In addition the memory sharing principle used in a mobile and context-sensitive application would make the offered services accessible on site, either requested explicitly or provided proactively.

3.1 Field Study

We conducted a contextual inquiry with four participants who cooked for guests. They have been interviewed about their menu selection, observed while shopping and cooking, and interviewed.

Even if participants are equally either enthusiastic or skeptical regarding the application scenario, the observation proves that for each participant it is rich in sharing occasions. Some of the main sharing occasions that occurred are the following:

- Asking guests (or friends with same food habits and culture) about their tastes and constraints (religion, medical restrictions, vegetarianism),
- Asking friends/mother about menu suggestions, as well as recipe ingredients and directions,
- Asking the guests whether they may like the menu, whether there are ingredients that they do not eat,
- Getting specialized stores recommendations (Muslim or Asian grocery stores, for instance),
- Finding alternative solutions when ingredients are not available in a store,
- Estimating food and spices / salt quantities.

3.2 Example Scenario

According to our studies results, a scenario for memory sharing in the everyday life could be sketched as follows:

Barbara is at home, thinking about a menu that might please Jessica, her colleague, whom she has invited for dinner. She only knows that she likes chocolate a lot. She checks if Jessica has any food constraints, and indeed she is vegetarian. To find recipe suggestions, she queries the memories of unknown vegetarian people, paying more attention to vegetarians she trusts since she already followed their recommendations. She selects a starter and a main dish. She is not sure that Jessica will like the mushrooms in the main dish, so she queries Jessica's memory about mushrooms and finds out that she liked a lot most of the dishes with mushrooms. She now searches for deserts with chocolate. The system remembers Barbara of a given user who helped her a lot the last time she was looking for recipes with chocolate. She browses through the recipes with chocolate of this person and decides to prepare one of her new recipes: a chocolate fondue.

Barbara is not sure to find in her usual supermarket the specific spices which are used in the main dish recipe. She checks whether one ever bought such spices at her supermarket. Since no recent result is returned, she finds with the system where the person offering the recipe bought them. She buys the spices there and the other ingredients at her usual supermarket. There, she takes mozzarella for the starter. The system informs her that users complained about the awful quality of this mozzarella brand. So, she chooses another brand.

While she is cooking, Barbara gets a request from a friend, Paul, who would like to know if she likes sushi. She gives him access to her experiences with sushi. Later, someone asks her where she buys coffee. She does not reply since she's in a critical step of her recipe. Jessica arrives and Barbara finishes preparing the main dish: she takes the spices from the shelf and is informed that Jessica does not stand spices in high quantities. She thus uses less spice and asks Jessica to taste to know if the spice quantity is appropriate.

The next day, Barbara and Jessica review recently captured events. Jessica and Barbara rate the diner episode and Barbara decides to set it public, so that her friends can learn about her tastes for their next invitations and also to recommend the recipes she used. She also gives trust points to people whose experiences helped her for the diner preparation in order to use those people's memories in priority in the future. She also reviews missed sharing occasions and authorizes all SHAREDLIFE users to access her experiences with coffee in the future.

4 Approach

The above scenario illustrates actions relevant for sharing, including issuing sharing requests, handling sharing requests, and handling sharing responses. In the following, we will explain how these might be addressed by means of augmented personal memories.

4.1 Issuing Sharing Requests

"She checks if Jessica has any food constraints..."

A single user's augmented memory is a rich source of situations and artifacts. We developed in the SPECTER prototype a combination of different approaches, which provides the user with manual and proactive means of retrieving and browsing augmented memories. Extending these means for a sharing scenario is partly straightforward. Thus, it is easy to imagine how the various views on memories could be enriched in order to exploit such personal information as starting point for manual sharing actions—e.g., by adding functions for sharing via a ubiquitous user model (cf. [Heckmann, 2005]) or by attaching retrieval functions to objects (see Figure 3).

However, the quality of the selected experiences is directly related to the adequateness of the selected sharing partners: according to our application scenario, users might be willing to view experiences of a given kind of individuals or of given known people or of people similar to themselves or to their guests.

Thus, the user needs ways to select sharing partners relevant for the current situation. The system could automatically select people according to the current situation characteristics (all users having experiences with the mozzarella Barbara is looking at), however there are cases where the user knows better than the system whose memories she wishes to explore, for instance, because of information available in the user's natural memory, but not in the augmented memory: in the coffee aisle, the user does not know which coffee to buy, but she remembers

Figure 4: Possible design. The first screen would proactively appear when the user looks at a product in a store, offering annotation possibility and services involving other users' memories. The second screen shows other users' annotations about the product in a shopping context.

that her mother makes excellent coffee and she would like to explore her memories to know which brand she buys.

We therefore need an interface which supports the user in the selection of sharing partners. Since the number of individuals can be high, the interface should provide ways to express constraints on the available sharing partners, e.g., on food habits, tastes, health problems, religion, or homeland—information which can be offered (and protected with respect to the partner's privacy constraints) by a ubiquitous user modeling server.

While some constraints refer to Boolean variables (vegetarian: yes/no), other constraints address variables which can take different numeric values, which imposes additional requirements on the user interface—e.g., the option to sort sharing partners in order to identify interesting groups. Here, we believe that trust is an indispensable dimension when it comes to communities and recommendations. Physical proximity is also important for such a mobile and ubiquitous application. Additional dimensions like the number of experiences exchanged in both directions, the social distance (direct contact, friend of a friend, etc.), the profile similarity, the quantity and average quality of the experiences could also be taken into account. Potential sharing partners could not only be sorted according to those dimensions, but could also be restricted to people in a certain range of values of these dimensions. For instance, a student in Germany who invites a Chinese student from his campus could select Chinese people far away (in China) to get authentic Chinese recipes and then Chinese people two kilometers away to learn where these buy the required ingredients.

Distributing people in various *people categories* can also simplify search of relevant sharing partners: we consider categories such as buddies (a common approach used in chat applications, Movielens, and other social software), "familiar strangers" (unknown people with a trust level assigned by the user with time) and other unknown people. Each category could be shown or hidden.

We designed four prototypes for the selection of sharing partners. They all respect the principles described above but differ in the visualization of the community and the number of dimensions used at a time in the visualization. Two of our prototypes are shown in Figure 4: in the basic prototype on the top, the number of selected buddies, familiar strangers and unknown people can be reduced by Boolean constraints and sliders for interval selection for each numeric constraint. On the prototype at the bottom, sharing partners are distributed in a 2D graph

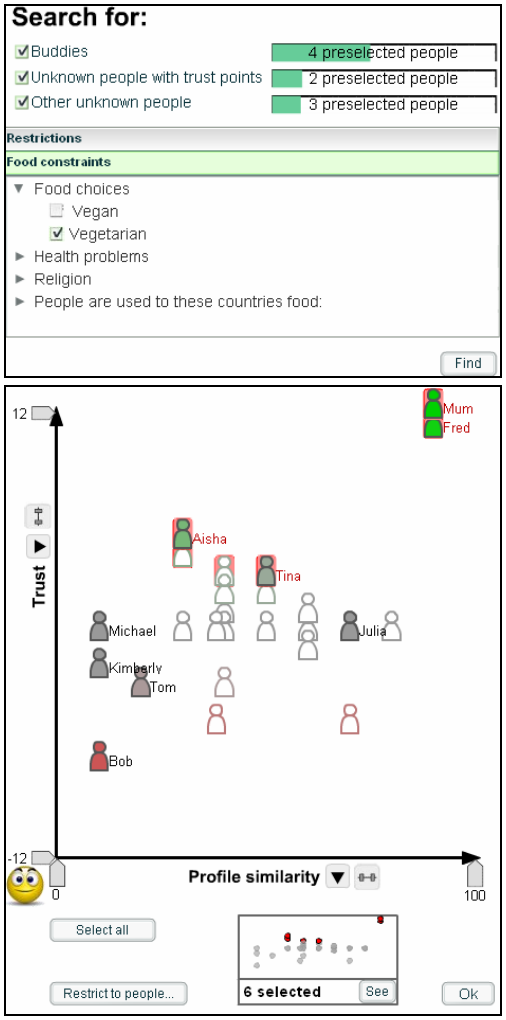


Figure 3: Two interfaces for selecting sharing partners.

according to the graph axes representing chosen dimensions. The results of an early study—whose extensive discussion would exceed the scope of this paper—indicate that the user might benefit from a combination of the two prototypes described here. We are currently working on the different possibilities of combination.

4.2 Handling Sharing Requests

"While she is cooking, Barbara gets a request…"

Our scenario includes many opportunities for sharing. Some are relevant for the user's current context (e.g., for Jessica, Barbara's request regarding her food constraints), some not (e.g., for Barbara, the request about where she buys coffee)—but probably these turn out to be relevant for future contexts. All these occasions will result in a large number of incoming sharing requests, which can hardly be handled by the user on his or her own.

Thus a straightforward approach which presents requests directly to the user is little promising—while it allows immediate reaction in urgent requests, the user might not be able (or willing) to verify all of them. In order to free the user from this burden, one could serve all incoming requests automatically based on a sharing policy specified in advance for the whole augmented memory. However, beside issues of privacy and trust in such

automatism, the unsupervised exchange of information might "overcrowd" the user's augmented memory with information never actually used.

An alternative way of handling sharing requests can be achieved by means of a mediator for user models. Thus, we explored in a prototype built upon SPECTER, how the user may exploit the facilities of an augmented memory to select explicitly data for sharing, attach situated access constraints, and then store these data on a ubiquitous user modeling server (cf. [Kröner *et al.*, 2006b]). There, default reasoning can be applied in order to infer additional privacy constraints. This way, the efforts required for specifying privacy constraints can be reduced since only a subset of information from the memory needs to be protected. And in addition, there is no need for the user to deal with reoccurring requests on data stored at the mediator. However, the whole process might turn out to be cumbersome if diverse requests enforce the user to submit again and again small pieces from the memories to the server, or unhandy if immediate response to a request on information not available at the mediator is required.

Therefore we propose to exploit the episodic nature of SPECTER's augmented memories for handling sharing requests. Following that model, the short-term memory enables an immediate analysis of and reaction on occasions of special relevance. In the case that an occasion is not relevant or ignored by the user, it is stored in the long-term memory, which enables the user to reflect later on these "missed" opportunities.

Reflection on Sharing

"Barbara and Jessica review recently captured events."

As discussed in Section 2.4, reflection on past events is a powerful means of exploiting augmented memories. This also holds for the reflection on sharing occasions, as illustrated by the following application examples.

Building a community: By evaluating recorded sharing occasions and actually shared experiences, the user may provide the system with feedback related to sharing partners (e.g., regarding privacy, trust, or expertise). Here, a trust level could be assigned with time: the user would give one trust point for each helpful experience or opinion which matches hers, or a negative trust point when one has an opinion opposite to hers.

Adding retrieval keys: By reflecting on experiences exchanged with others, the user may decide to add retrieval keys to her personal memory: comments, event ratings and retrieval helpers such as landmarks (cf. [Horvitz et al., 2004]) and collaborative tagging.[1]

Pending requests: A sharing request is not necessarily bound to a small time interval. For instance, a sharing partner might express a general interest in certain information. Therefore, reflection on such requests should allow the user to react to a request as long as the preconditions of the particular request are still valid.

Setting up sharing rules: Due to the sheer amount of requests there is always a risk that the user misses interesting occasions. To avoid such situations in the future, the recorded occasions can be exploited in order to configure the system's sharing behavior. If Barbara notices in her records requests about her (public) preferred coffee strength, she can set up a rule which lets the system reply on such requests automatically. Other rules might include situational elements; for instance, the user might want to trigger an anonymous sharing mode once a sharing partner is less the 50 meters away. Therefore, we want to assist the user in extracting from the records the characteristic features of the sharing occasion of interest and to bind these to services provided by the system. In order to achieve this goal, we will exploit (and extend, if required) an approach discussed in [Bauer *et al.*, 2005].

These applications of experience records are all affected by a specific problem: since recording sharing occasions will not reduce their mass, we have to provide the user with powerful means to filter and rank such records. In part, this issue can be addressed by regular GUI features (e.g., filters based on the user's buddy list); in addition we intend to introduce a measure for the value of sharing occasions, a work which has recently started.

4.3 Handling Sharing Responses

"Barbara finds out that she liked a lot most of the dishes with mushrooms."

Our goal is to respond in sharing requests not only with a snapshot-like excerpt of the user model, but also with related experiences. Sending experiences instead of user models serves several purposes. Thus, we expect that user models of sharing partners will often be incomplete or partially protected, which may prevent a system from inferring information of interest for the requesting user. Here, experience records allow the user to make assumptions about the course of events on his or her own. Similarly, the user is not forced to trust inferences drawn by a sharing partner's SHAREDLIFE since retrieved experiences allow for a re-interpretation by user and system.

The exchange of experiences allows also addressing the variety issue known from recommender systems (cf. [Jameson, 2006]): if people query their guest's memory to learn about her favorite dishes, she can get the same dishes at each invitation. Accompanying each dish recommendation by the guest's episodic memory involving it could address this issue: the frequency and dates when the guest ate the recommended dish are visible as well as the evolution of the dish rating over time.

Of course it may happen that no experiences are returned to a request—for instance, because the person never experienced for some reasons something matching the request or because the relevant experiences are protected by privacy constraints. In this case, the user can explore her guest's network to find people similar to her (see Section 4.1), or who have experiences related to cooking for this guest, or whom the guest trusts.

5 Related Work

Popular approaches related to our research are forum, Wiki, and in particular blogs. While these also provide means of sharing experiences, our work extends these in populating the experience base automatically, in assisting the user with proactive retrieval methods, and in allowing the specification of constraints on privacy and trust.

Thus, our work is also related to research on extending the blogging idea. For instance, FeedMap[2] allows for connecting blogs to locations and thus realizes a location-centered sharing approach, however, affected by the same

[1] A good example of applying collaborative tagging can be found at http://movielens.umn.edu/

[2] http://www.feedmap.net/

limitations regarding privacy and trust which apply for regular blogging. This issue is addressed by Moleskiing, which introduces trust on expertise to blog-like mechanisms. In addition, this work exploits reflection on past events in order to prepare experiences for (non-situated) sharing. [Avesani *et. al.*, 2005]

A well-known system related to augmented memories and sharing of memories is MyLifeBits. It assists its user in creating presentations from documents (e.g., photos, text files) collected over an individual's life; the documents may have attached automatically captured meta data (e.g., GPS). [Gemmell *et al.*, 2005]

Other related research addresses the unobtrusive capturing of meeting or classroom activities. These approaches often focus on creating a memory common to all participants in contrast to personal sharable memories. Studies showed that students in such settings were missing means to personalize the captured data and to retrieve it easily (cf. [Abowd *et al.*, 2000]). Another attractive scenario for research on sharing experience records are conference visits. Thus, such records can be exploited for initiating communication between participants (cf. [Müller *et al.*, 2004]), or, in combination with blogging, for sharing selected experiences (cf. [Numa *et al.*, 2006]).

Close to our research are the goals of a project started in 2006 by Nokia: SharMe[3] aims at recording input from mobile devices such as cell phones and at supporting the user in sharing that information with others.

6 Conclusion and Future Work

In this paper we tried to provide some answers to questions raised by memory sharing: we described combinable ways to provide access to others' experiences relevant to the user in the current situation. We described principles enabling to manually select relevant sharing partners. We also described how incoming sharing requests could be handled both automatically when sharing policies apply or manually if necessary and how missed sharing opportunities can be used to specify sharing rules. However, those concepts need to be designed and tested with users to find principles which will ensure the acceptance of memory sharing in context-sensitive software. Our field study and the user study about the community browser will be followed by other iterative sessions of design and evaluation with users. Because of the project's focus on sharing experiences collected over time, we will need to conduct the final evaluation over a long period with a consequent group of users made of known and unknown people.

Acknowledgments

This research is supported by the German Ministry of Education and Research respectively under grant 524-40001-01 IW C03 (project SPECTER) and 01 IW F03 (project SHAREDLIFE). Thanks to all projects members.

References

[Abowd *et al.*, 2000] Gregory D. Abowd and Elizabeth D. Mynatt. Charting past, present, and future research in ubiquitous computing. ACM Transactions on Computer-Human Interaction, 7(1):29–58, 2000.

[Avesani *et al.*, 2005] Paolo Avesani, Paolo Massa, and Roberto Tiella. A trust-enhanced recommender system application: Moleskiing. In SAC '05: *Proceedings of the 2005 ACM symposium on Applied computing*, pp. 1589–1593, Santa Fe, New Mexico. ACM Press, 2005.

[Bauer *et al.*, 2005] Mathias Bauer and Stephan Baldes. An Ontology-Based Interface for Machine Learning. In J. Riedl, A. Jameson, D. Billsus, and T. Lau (eds.), *Proceedings of the 2005 International Conference on Intelligent User Interfaces* (IUI 2005), pp. 314–316. New York: ACM, 2005.

[Gemmell *et al.*, 2005] Jim Gemmell, Aleks Aris, and Roger Lueder. Telling stories with MyLifeBits. In *Proceedings of the IEEE International Conference on Multimedia & Expo (ICME 2005)*, Amsterdam, The Netherlands, July 2005.

[Heckmann, 2005] Dominik Heckmann. Ubiquitous User Modeling. Ph.D. thesis, Saarland University, Department. of Computer Science, Germany, 2005.

[Horvitz *et al.*, 2004] Eric Horvitz, Susan Dumais, and Paul Koch. Learning Predictive Models of Memory Landmarks. In *Proceedings of the CogSci 2004: 26th Annual Meeting of the Cognitive Science Society*, Chicago, August 2004.

[Jameson, 2006] Anthony Jameson. Adaptive Interfaces and Agents. In J. Jacko & A. Sears (eds.), *Human-Computer Interaction Handbook,* Lawrence Erlbaum Associates, New Jersey, 2006.

[Kröner *et al.*, 2006a] Alexander Kröner, Dominik Heckmann, and Wolfgang Wahlster. SPECTER: Building, Exploiting, and Sharing Augmented Memories. In K. Kogure (ed.), *Proceedings of the Workshop on Knowledge Sharing for Everyday Life* (KSEL 2006). Kyoto, Japan, February 9–10, 2006.

[Kröner *et al.*, 2006b] Alexander Kröner, Dominik Heckmann, and Michael Schneider. Exploiting the Link Between Personal, Augmented Memories and Ubiquitous User Modeling. In: *Proceedings of the ECAI 2006 Workshop on Ubiquitous User Modeling* (UbiqUM 2006), Riva del Garda, Italy, 2006.

[Müller *et al.*, 2004] Christof E. Müller, Yasuyuki Sumi, Kenji Mase, and Megumu Tsuchikawa, Experience sharing by retrieving captured conversations using non-verbal features. In *Proceedings of the 1st ACM Workshop on Continuous Archival and Retrieval of Personal Experiences* (CARPE 2004), pp. 93–98. New York: ACM, 2004.

[Numa *et al.*, 2006] Kosuke Numa, Toshiyuki Hirata, Ikki Ohmukai, Ryutaro Ichise and Hideaki Takeda: Action-oriented Weblog to Support Academic Conference Participants. In *Proceedings of the IADIS International Conference on Web Based Communities (WBC2006)*, San Sebastian, Spain, 26–28, February 2006.

[Plate *et al.*, 2006] Carolin Plate, Nathalie Basselin, Alexander Kröner, Michael Schneider, Stephan Baldes, Vania Dimitrova, and Anthony Jameson. Recomindation: New Functions for Augmented Memories. In V. Wade & H. Ashman (eds.), *Adaptive hypermedia and adaptive web-based systems: Proceedings of AH 2006*. Berlin: Springer, 2006.

[3] http://research.nokia.com/research/projects/sharme

Predicting User Experiences through Cross-Context Reasoning

Shlomo Berkovsky[1], Lora Aroyo[2], Dominik Heckmann[3], Geert-Jan Houben[4], Alexander Kröner[3], Tsvi Kuflik[1], Francesco Ricci[5]

[1]University of Haifa, Israel
{slavax@cs, tsvikak@is}.haifa.ac.il

[2]Technical University of Eindhoven, The Netherlands
l.m.aroyo@tue.nl

[3]German Research Center for Artificial Intelligence
{heckmann,kroener}@dfki.de

[4]Vrije Universiteit Brussel, Belgium
Geert-Jan.Houben@vub.ac.be

[5]Free University of Bolzano/Bozen, Italy
fricci@unibz.it

Abstract

The existing personalization systems typically base their services on general user models that ignore the issue of context-awareness. This position paper focuses on developing mechanisms for cross-context reasoning of the user models, which can be applied for the context-aware personalization. The reasoning augments the sparse user models by inferring the missing information from other contextual conditions. Thus, it upgrades the existing personalization systems and facilitates provision of accurate context-aware services.

1 Introduction

The overwhelming size of nowadays information world, jointly with limited processing capabilities of the users pose a need for developing and exploiting personalization approaches allowing an easier navigation and access means. Personalization research yielded a number of techniques, such as collaborative [Herlocker et al., 1999] and content-based filtering [Morita and Shinoda, 1994], item-to-item collaborative filtering and others. These techniques facilitate adapting the services provided to the user to his/her actual interests and needs, as expressed by the User Models (UMs) [Kobsa, 2001] that constitute an essential input for every personalization technique.

Despite an intensive research, aimed mainly at improving the prediction accuracy of the personalized recommendations provided to the user, personalization techniques suffer from a severe limitation. The provided personalization typically relies on a UM, which has been tailored for an application, characterized by specific personalization algorithms and a specific application domain. Moreover, user needs represented in the UM are generally valid only in a specific context, which is typically ignored by the state-of-the-art personalization systems.

Taking into account various contextual conditions may be beneficial and even essential for providing accurate and efficient personalization. For example, consider an everyday task of recommending radio music for a user during his/her daily driving from home to work. Although the user's music preferences are quite steady, different types of music may be recommended as a function of his/her mood, presence of other people, traffic conditions and even weather conditions. Hence, there is an emergent need for *slicing* the general preferences represented by the UM according to various contextual conditions. This will allow considering the contextual aspects and providing the user context-aware personalization.

On one hand, providing the user context-aware personalization may significantly improve the accuracy and the usefulness of the provided personalization service. On the other hand, the information stored in the UMs may not suffice for providing accurate context-aware personalization. This will happen due to the above slicing of the general UMs that will split the available information about the user according to the appropriate contextual conditions. Hence, any attempt of inserting the context-awareness dimension into the state-of-the-art personalization systems should involve developing a reasoning mechanism, which will facilitate inferring the essential parts of the UMs across various contextual conditions.

This position paper focuses on developing mechanisms for cross-context reasoning of the UMs, which can be applied for the purposes of the following context-aware personalization. The core element of these mechanisms is referred to as *user experience*, or for the sake of brevity, just experience. By experience we denote an explicit or implicit feedback provided by a user as a result of experiencing a certain content (or item) in a certain context. Figure 1 schematically illustrates the experience compo-

nents. For example, a user John Doe may rate pop-music radio program listened when driving alone on a rainy morning by assigning it 4 stars on a 5-stars scale. In this case, the experience of 4 stars is given by the user John Doe to the content of pop-music radio program in the context of a rainy weather and being alone. The union of such experiences is considered as the UM. Given a set of past experiences represented by the UM, the goal of the above cross-context reasoning mechanism is inferring the essential parts of the UM for the purposes of generating accurate context-aware personalization for future experiences.

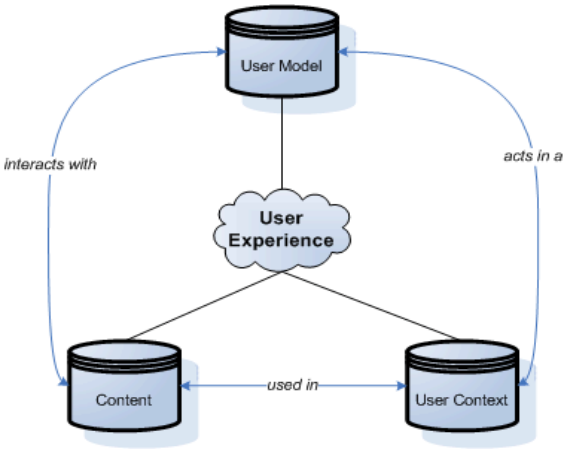


Figure 1: Representation of User Experiences

Our approach is based on semantically-enriched descriptions of the experiences. This means that all the components affecting the experience, i.e., users, contents and contexts, are described using semantic schemata. These schemata facilitate defining various cross-context reasoning mechanisms, which will augment the sparse parts of the UM by inferring the missing information from past experiences in other contextual conditions.

Moreover, cross-context reasoning may be integrated with other personalization approaches, such as cross-user (i.e., collaborative) and cross-content (i.e., content-to-content, or item-to-item) reasoning. For example, applying the reasoning mechanisms on the experiences of similar users on the required content will lead to collaborative cross-context reasoning, while applying them on the experiences of the given user on similar contents will lead to item-to-item cross-context reasoning. Also, we consider applying an advanced cross-context hybrid reasoning, integrating both cross-user and cross-content reasoning

Hence, the contribution of our work is two-fold. First, we provide a high-level framework for semantic representation of context-aware user experiences on contents. Second, we exploit this framework for defining various reasoning mechanisms for (1) inferring the essential parts of context-aware UMs, and (2) providing context-aware personalization. This upgrades the capabilities of the state-of-the-art personalization systems and facilitates provision of accurate context-aware personalization services.

The rest of the paper is structured as follows. In section 2 we overview the related works on semantic-based and context-aware personalization approaches. In section 3 we briefly describe an example scenario that will be used for the following semantic representation and reasoning. In section 4 we discuss the semantic data representation. In section 5 we discuss the proposed reasoning mechanisms. Finally, in section 6 we conclude the paper and discuss several veins for future research.

2 Related Work

Rich context models are of special value for user support outside of a typical desktop scenario. For instance, when guiding its user through a museum or during a sightseeing tour, an assistant may adapt its personalization with respect to contextual information such as the visitor's interests, location, available time, financial limitations, mobility constraints, and local weather conditions (see, for instance, [Davies et al., 2001] and [Cinotti et al, 2004]). Here, the user stays in variations of a single context (the tour); other scenarios combine such rich context models with adaptation to diverse tasks. For instance, [Kuwahara et al., 2003] describe a context-aware assistant, which aims at avoiding nursing accidents in hospitals. The system has to distinguish between diverse context models of various nursing tasks and has to predict actions before their actual occurrence. This approach includes modeling across contexts in various dimensions; however, even in this case the set of supported contexts is of limited size and is known in advance.

The previously described works make use of UMs, which provide information about the user in diverse contexts. Such contextualized user modeling is a research area on its own. As pointed out in works such as [Harvel et al., 2004] and [Kern et al., 2006], context-based user modeling may already be performed on the level of sensor data. Our work aims at a higher level of abstraction, in particular, at a UM built from semantic structures. An instance of this approach is provided by [Mehta et al., 2005], who propose the use of a common ontology-based user-context model as a basis for the exchange of UMs across applications. In their approach the context is modeled as an extensible set of facets representing the characteristics of the user and his current context. Ubiquitous user modeling [Heckmann, 2005] extends this idea by continuously modeling the user by means of situational statements, which enables modeling of the user in (ideally) any context. However, if the user is in a context not experienced before, the question arises which information from previous contexts could be exploited for user support. Therefore, we propose in this article a reasoning mechanism which allows for assembling a UM for a given situation based on previous experiences.

The use of context for adapting user support is subject of considerable research efforts in recommender systems research. For instance, [Herlocker and Konstan, 2001] presents a task-focused recommender, which first retrieves items similar to items associated with some task, and then applies collaborative filtering in order to rank the items based on the interest prediction. In this case, context is defined by the task only. [Adomavicius et al., 2005] discusses the ways for achieving a more complex context model for recommendations by means of a multi-dimensional data warehousing approach. However, while the latter allows providing context-aware recommendations, it does not deal with projecting user modeling information between various contextual conditions.

In [Ricci et al., 2003] a case-based recommendation approach has been used to model a travel recommendation session as a case. Here, a case is indexed by various contextual features, such as the type of travel, the group composition, the distance from the target location, the travel season and so on. These features, among others, contribute to determining what stored recommendation sessions must be retrieved to influence the ranking of the items

considered by the user. Hence, similarity-based reasoning is exploited to make cross-context deductions.

3 Example Scenario

Everyday life is composed of various events where users request and use information. This information should suit the individual characteristics of the users and the specific context of that user. As an example, let us consider two days that contain simple traveling scenarios. The two days differ in that one is defined as a work day while the other is defined as vacation day.

Using the above characteristics, here are two different but somewhat similar scenarios, with different context that may help illustrate the need for context. In both scenarios, let us assume that our user travels alone for the whole day, leaves home in the morning and returns in the afternoon.

Let us say now that our traveler is married with two little kids, likes country music, likes nature, outdoor sports, including water sports and especially surfing and likes Italian food and coffee. In the following scenarios we highlight the context-aware personalization service provided by the system in form of recommendations.

1. Working day: Traveling from home to a city nearby for a business meeting. The meeting is planned to start at 10:00, end at 12:00 and our traveler is expected back in the office at 14:30 for another meeting
2. Vacation day: Traveling from home to the same city for a vacation day. During that day our traveler will go to the lake, spend some time there and return home sometime afternoon after enjoying preferred water sport and lunch.

The driving distance from home to the other city is about an hour, depends upon traffic conditions. There is a selection of roads – highways and secondary scenic routs.

In the first scenario, the traveler has a meeting at 10:00, since driving time is about an hour a recommendation for traveling is to **leave home at 08:30** (after rush hour), allowing some time for traffic congestions and planning to arrive a bit early. The travel context is that the traveler travels alone, the time is morning, the season is summer, weather is nice, means of transportation is a private car, travel goal is work, and travel time is about an hour. This traveling context requires information and recommendations about the road conditions, traffic and parking place at the end (parking place, next to the meeting place, where the traveler will get receipt for parking). During the trip, there is another recommendation task: music selection out of a choice of radio stations and CD player. Our traveler drives on the **highway**, listening to a favorite **country** singer, gets to the meeting place about 15 minutes early, parks in a short walking distance from the meeting place. There are 15 minutes to wait, so the system recommends a third task: **having coffee** at a nearby bar. The meeting finishes at 12:30. As our traveler needs to get back to the office by 14:30, the system suggests having a **Pizza** for lunch (fast Italian food) at a near by Pizza stand (also within the expenses budget of our traveler). Our traveler starts driving back to work at 13:30, at this time the system suggest **taking the highway** (shortest path). Regarding music, the system recommends **favorite, but not relaxing** (relaxing music may make the traveler sleepy) country music from one of the local radio stations.

In the second scenario, there are no time constraints, so the system suggests **leaving at 09:00** to avoid traffic, taking a **scenic road** to the lake (the city is near a lake), parking in a free parking area, a bit away from the city, but where surfing equipment can be rented and where there are also some restaurants. During the trip to the lake, the system suggests a favored **country** CD. Our traveler gets to the lake, surfs, swims a little and breaks for lunch at 13:00. The system recommends an Italian **restaurant** near the beach. Our traveler decides not to accept the recommendation. Instead he/she decides to start heading home. The system recommends **taking a scenic road** back and stopping in a good Italian **restaurant** along the way, about 15 minutes drive from the lake. Our traveler follows the recommendation. After lunch, the system recommends **favorite but not relaxing** (relaxing music may make the traveler sleepy) country music from one of the local radio stations.

The above two scenarios, detailed for the same users in two different contexts: leisure and work, almost identical in most of the details, demonstrate the idea of context-awareness. Work context is different from leisure context (in this specific example, due different time and budget constraints), and the recommendations are also different. Even within the same general context (work-day context for instance) there are different sub-contexts. For example, restaurant recommendation may be different given the availability of time: if the meeting ended early, there is more time to get to a restaurant, but if the meeting ended late, there is time to grab a Pizza at the nearest Pizza stand and go back to the office.

4 Data Representation

The fundamental problem related to data representation is "how can this heterogeneous situational information be represented in a uniform, efficient and semantically-enriched fashion"? We addressed it basing our approach on so-called s*ituational statements* [**Error! Reference source not found.**] that serve as integrating data structures for user modeling and context-awareness.

The basic idea behind situational statements is to apply predefined meta-level information in an extended RDF representation with OWL ontologies. These ontologies provide a shared and common understanding of a domain allowing communication between heterogeneous widely spread application systems. The newly defined general UM ontology GUMO [**Error! Reference source not found.**] is collecting the user's dimensions modeled within user-adaptive systems, e.g., the user's age, and occupation. Furthermore, it also facilitates representing the user's interests and preferences.

Similarly, GUMO facilitates modeling in RDF various dimensions of context, e.g., day time, season, companions, motivation (for the traveling scenario) and others. Figure 2 illustrates partial representation of context in GUMO. In the same manner, also the items can be modeled in RDF.

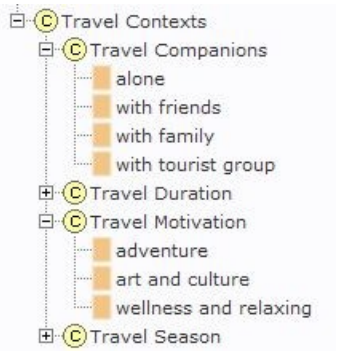


Figure 2: Context Representation in GUMO

5 Reasoning Rules

In the previous section we have seen how we can syntactically represent the data from user models in RDF. As we have explained in the introduction, our interest in this paper is to infer essential parts of a context-aware UMs or provide context-aware reasoning. It means that we are interested in combining context-specific user data and infer context-aware user data.

To explain this inference mechanism, we need to consider the UM data in more detail. A user experience was previously defined as the combination of user feedback for a certain content item in a certain context. For this context to be captured, we use (here in a simplified syntax) situational descriptions, such as:

`context.motivation=work` or
`context.time=afternoon`

With the aid of all the situational statements that we have at our disposal, we should understand what the relevant contextual aspects are. For example, in an experience we will have a combination of a situation, an item, and a rating. Here, we give the details of a concrete example:

```
context.motivation=work
context.time=afternoon
item.meal.price=moderate
rating=0.8                                    (*)
```

For the moment, we have assumed that these experiences are simply registered and explicitly stored like that. As we have explained in the motivation, it can be the case that we need a UM that deals with the context-awareness in a more efficient way by "inferring the essential parts". Likewise, from the perspective of the personalization system, we can have a case, where a number of experiences are available, but in a new situation no experiences are available to base the recommendation on. To help resolving this problem of inferring the essential parts, we can exploit the inference mechanisms in the data structures. For this, it might be necessary to define rules that indicate how the different aspects of the situations relate to each other. We now sketch a number of illustrative cases, with rules that help to define how we obtain the (cross-context) inferred knowledge.

In the first scenario is we derive knowledge about a more generic situation from a more specific one by discarding some contextual information. For example, a rule:

```
context.motivation and context.time
implies context.time
```

could help to aggregate the detailed knowledge with a certain knowledge referring to `context.motivation` into more coarse-grained knowledge referring to `context.time` only. This could define the factors that are more important for the context-awareness and help to deal with a situation such as

```
context.time=afternoon
```

by inferring that for this situation, the above rating (*) can be used as a basis for recommendation. Rules like this would help to define how the different contextual aspects are related to each other, such that also for a situation

```
context.motivation=leisure
context.time=afternoon
```

some user modeling information will be available, even if no previously experiences have been recorded for this situation. Note that if there would have been experiences recorded for this situation, applying the above rule would result in multiple ratings being available for consideration in the personalization stage.

In the previous scenario, we have dealt with rules that concern the presence or absence of aspects in the situational statements. In the following scenario, we exploit knowledge about the domain of values for our situation aspects. For example, consider a situation:

```
context.motivation=work
context.time=4pm
```

Knowing the rating (*), we would be able to use this, if we would know that 4pm is a time in the afternoon. So, with a rule like:

```
4pm implies afternoon
```

we would be in the position to keep in the UM only the essential statements for the experiences, and still be able to infer the relevant situations. This scenario fits perfectly with our RDF/OWL-based approach where we can rely on the fact that the value domains are represented through ontological structures that facilitate this kind of inference.

So, the first type of rules is associated with the presence of situation aspects (the genericity of the situations), whereas the second type is associated with the structure inside the domains and domain knowledge for the situation descriptors. Needless to say that it is also possible to define rules that combine the above two types.

As a result of these rules, whenever we are in a situation *S* for which we want to provide personalization, we can infer all those experiences that "hold", i.e. are considered relevant (because they have a situation implying *S*).

We would like to stress that in previous inferences we considered the situation and item parts of the experiences, and not the ratings. Obviously, we could also include the ratings in the rules and exploit them by the inference mechanisms. For example, consider two experiences:

```
context.motivation=work
context.time=afternoon
item.meal.price=moderate
rating=0.8
```

and

```
context.motivation=leisure
context.time=afternoon
item.meal.price=moderate
rating=0.2
```

Availability of ratings in the experiences allows supporting different kinds of reasoning. For example, if the value of `context.motivation` is unknown, some probabilistic model can produce a prediction of:

```
context.time=afternoon
item.meal.price=moderate
rating=0.6
```

We would like to stress that the above examples all relate to one and the same user and item in different situations, i.e., new situations on the basis of situations from previous experiences on the same items. Mainly, the rules help to define how we can infer knowledge based on how the situations are structured. This is therefore an example of pure cross-context reasoning.

We point out here that in the above examples, we did not refer to the item in question (e.g., meal). It is obvious that in the same line we could also have included the items from the experiences in the rules, yielding cross-context item-item reasoning from past experiences on other items in other contexts. In the same spirit, including the users in the rules yields cross-context collaborative (cross-user) reasoning from past experiences of other users in other contexts. Finally, both of the above methods could be integrated, yielding a hybrid reasoning. At this stage, we just point out these possibilities, without exploring them in depth.

Once the required UM data is inferred, the following stage of the context-aware personalization actually deals with generating the recommendations. For this purpose, any state-of-the-art recommendation technique may be applied.

6 Conclusions and Future Research

This paper motivates the need for cross context personalization and suggest an initial model for it. It also integrates it with the ideas adapted from the state-of-the art personalization techniques in order to provide a complete framework for context-aware personalization. Future research will focus on formalizing the model, integrating it with known representation and reasoning techniques and demonstrating it in everyday scenarios as an initial proof of concept.

References

[Adomavicius et al., 2005] Adomavicius, G., Sankaranarayanan, R., Sen, S., and Tuzhilin. A. (2005). Incorporating Contextual Information in Recommender Systems Using a Multidimensional Approach. In: ACM Transactions on Information Systems, vol. 23, (1).

[Cinotti et al., 2004] Cinotti T.S., Garzotto, F., Muzii, R., Malavasi, M., Galasso, S., Raffa, G., Roffia, L. and Varlese, V. (2004). Evaluating Context-aware Mobile Applications in Museums: Experiences from the MUSE Project. In: Proceedings of the Eighth Annual Conference on Museums and the Web (MW 2004).

[Davies et al., 2001] Davies, N., Cheverst, K., Mitchell, K., Efrat, A. (2001). Using and determining location in a context-sensitive tour guide. In: IEEE Computer Society Press 34(8).

[Kuwahara et al., 2003] Kuwahara, N., Noma, H., Kogure, K., Hagita, N., Tetsutani, N., Iseki, H. (2003). Wearable auto-event-recording of medical nursing. In: Proceedings of the Ninth IFIP TC13 International Conference on Human-Computer Interaction (INTERACT 2003).

[Harvel et al., 2004] Harvel, L., Liu, L., Abowd, G.D., Lim, Y.X., Scheibe, C., Chatham, C. (2004). Context Cube: Flexible and Effective Manipulation of Sensed Context Data. In: Proceedings of the 2nd International Conference on Pervasive Computing.

[Heckmann, 2005] Heckmann, D. (2005). Ubiquitous User Modeling. PhD thesis, Department of Computer Science, Saarland University, Germany.

[Herlocker et al., 1999] Herlocker J.L., Konstan, J.A., Borchers, A., and Riedl, J. (1999). An Algorithmic Framework for Performing Collaborative Filtering, In: proceedings of the SIGIR Conference.

[Herlocker and Konstan, 2001] Herlocker, J. L. and Konstan, J. A. (2001). Content-independent, task-focused recommendations. IEEE Internet Computing, vol.5.

[Kern et al., 2006] Kern, N., Schmidt, A., and Schiele, B. (2006). Context Annotation for A Live Life Recording. In: Journal of Personal and Ubiquitous Computing - Special Issue on Memory and Sharing of Experiences, vol. 10, Springer.

[Kobsa, 2001] Kobsa, A.. (2001). Generic User Modeling Systems. In: User Modeling and User-Adapted Interaction, vol. 11(1-2).

[Mehta et al., 2005] Mehta, B., Niederée, C., Stewart, A., Degemmis, M., Lops, P., and Semeraro, G. (2005). Ontologically-enriched unified user modeling for cross-system personalization. In Proceeding of the User Modeling.

[Morita and Shinoda, 1994] Morita, M., and Shinoda. Y. (1994). Information Filtering Based on User Behavior Analysis and Best Match Retrieval. In: proceedings of the SIGIR Conference.

[Ricci et al., 2003] Ricci, F., Venturini, A., Cavada, D., Mirzadeh, N., Blaas, D., and Nones, M. (2003). Product recommendation with interactive query management and twofold similarity. In: proceedings of ICCBR 2003, the 5th International Conference on Case-Based Reasoning.

[Wasinger et al., 2003] Wasinger, R., Oliver, D., Heckmann, D., Braun, B., Brandherm, B., and Stahl, C. (2003). Adapting Spoken and Visual Output for a Pedestrian Navigation System, based on given Situational Statements. In: proceedings of the GI-Workshop on Adaptivity and User Modeling in Interactive Software Systems.

Can Log Files Analysis Estimate Learners' Level of Motivation?

Mihaela Cocea & Stephan Weibelzahl
National College of Ireland
Mayor Street, Dublin 1
[mcocea, sweibelzahl]@ncirl.ie

Abstract

The learners' motivation has an impact on the quality of learning, especially in e-Learning environments. Most of these environments store data about the learner's actions in log files. Logging the users' interactions in educational systems gives the possibility to track their actions at a refined level of detail. Data mining and machine learning techniques can "give meaning" to these data and provide valuable information for learning improvement. An area where improvement is absolutely necessary and of great importance is motivation, known to be an essential factor for preventing attrition in e-Learning. In this paper we investigate if the log files data analysis can be used to estimate the motivational level of the learner. A decision tree is build from a limited number of log files from a web-based learning environment. The results suggest that time spent reading is an important factor for predicting motivation; also, performance in tests was found to be a relevant indicator of the motivational level.

1 Introduction

Logging the users' interactions in educational systems gives the possibility to track their actions at a refined level of detail. Log files are easy to record for a large number of users, they can capture a large variety of information and they can even be presented in an understandable form. Thus, these data are a potentially valuable source of information to be analyzed and used in educational settings. Automatic analysis of log data is usually used to detect regularities and deviations in groups of users, to provide more information to tutors about the learners, to offer suggestions for further actions – mostly for the "deviation" cases.

A particularly important type of "deviation" is low motivation behavior usually associated with drop-out [Martinez, 2003]. Thus, identifying the low motivated learners and finding remedial actions would result in lower rates of drop-outs. We are interested in finding regularities in the user's behavior that could indicate their general motivational level. The preliminary investigation presented in this paper is looking at the possibility to predict the engagement / disengagement of learners from common log files data.

The paper is organized as follows. Section 2 discusses previous work related to the use of log files analysis in education, with a particular interest in approaches to motivation. It also includes a brief description of our research approach on motivation. Section 3 describes the information contained in the log files used for analysis and the indicators refined from the basic log data. The actual analysis and possible interpretations are described in Section 4. Section 5 concludes the paper with a summary and implications for further work.

2 Previous work

Automatic analysis of interaction data is used in research areas such as educational systems, data mining and machine learning. Educational systems can benefit from data mining and machine learning techniques by giving meaning to click-through data and associating these data with educational information.

Log files analysis has been used for a variety of purposes: provide information to tutors to facilitate and make more accurate the feedback given to learners [Merceron and Yacef, 2003], monitor group activity [Kay *et al.*, 2006], identify benefits and solve difficulties related to log data analysis [Heiner *et al.*, 2004], use response times to model student disengagement [Beck, 2004], infer attitudes about the system used, attitudes that affect learning [Arroyo *et al.*, 2004], developing tools to facilitated interpretation of log files data [Mostow *et al.*, 2005].

In relation to research on motivation, activity tracking has also been considered as a source of information for assessing users' motivation. Thus, there a number of approaches have been presented trying to infer motivational states from the learners' interactions with the systems: 1) a rule-based approach to infer *relevance*, *confidence*, *satisfaction* (from ARCS model [Keller, 1987]), *effort* and *sensory/ cognitive interest* [de Vicente and Pain, 2003], 2) inferring *confidence*, *confusion* and *effort* from: the learner's focus of attention, the current task and expected time to perform the task [Qu *et al.*, 2005], 3) inferring *attention* and *confidence* from the learner's actions, using factor analysis to group the actions that indicate the two motivational states [Zhang *et al.*,2003]

The previously presented approaches related to motivation try to infer automatically different motivational states by connecting the learner's actions (reading a page, solving a quiz, etc) and the time to perform them, with performance, which is typical information for educational systems.

Using the same type of information, rather then inferring such well refined motivational states, we are interested in finding a general indicator for motivational level as a starting point for further investigation about the learner's motivation [Cocea, 2006]. Thus, after finding this general indicator of motivation, an assessment of mo-

tivational characteristics will be conducted for the disengaged students, in order to have more detailed and accurate information about their level of motivation and, thus, pursue a more efficient intervention. This approach has the advantage of identifying the low motivated learners and focusing on them for further assessments and interventions because they are the potential drop-out students. Motivated students can also benefit from motivational assessment and intervention, but our main concern is for low motivated students as this is a problem in e-Learning.

We present here the results of the analysis of a limited number of log files from an online-course called HTML-Tutor. The purpose of this analysis is to investigate if commonly logged data can be used for predicting a general level of motivation. If indeed log file analysis can provide information about the motivational level, then potentially a motivational module could be included in educational systems that log the learner's interactions.

3 Log files description

HTML-Tutor is an interactive learning environment which offers an introduction to HTML and publishing on the Web; it is online and can be accessed freely. We don't have any information about the users except the data from the log files. They could be of any age and using the system for different purposes.

3.1 The logged parameters

The logged information is described in Table 1. Each event is recorded with a timestamp.

Table 1. Information included in HTML-Tutor log files

Event	Properties/Description
Login/logout	User ID
Goal	The purpose of using HTML-Tutor
Preferences	Different options can be changed by the user (e.g. frames/no frames, link annotation/no link annotation etc.)
Page access	PageID
Test	TestID, result: Correct/False
Hyperlink	The Page ID of the triggered page from the link
Manual	Looking for help about the system
Help	Looking for help about the learning content
Glossary	Word looked up
Communication	Access to a discussion lists and if a comment has been made
Search	Terms searched
Remarks	User's Remarks
Statistics	Users can see statistics about their activity, such as: time spent from the last login, percentage covered in a certain chapter, percentage of correctly answered tests etc

3.2 The analysis parameters

From the basic log data presented in Table 1, five indicators/ attributes with higher level of information have been calculated: performance on tests, the time spent reading, the number of accessed pages, the time spent solving tests and level of motivation: engaged / disengaged. A description of these attributes and the way they were calculated is presented in Table 2. These derived indicators are used in the analysis presented in Section 4.

Table 2. Derived attributes to be used in the analysis

Attribute	Description
UserId	A unique identifier per each user
Performance	Percentage of correctly answered tests (calculated as number of correct tests divided by total number of performed tests)
TimeReading	Time spent on pages (calculated as the sum of the time spent on each page accessed) in a session
NoPages	The number of accessed pages
TimeTests	The time spent performing tests (calculated as the sum of time spent on each test)
Motivation	Engaged / Disengaged

The information was aggregated in order to create the database with the same indicators for every user and to give meaning to the click through data. Basically only two events with their average times are considered (reading and taking tests) because none of the other events were registered in the log files considered for analysis.

The time spent reading refers to the total time spent reading in a session. The last attribute in Table 2, motivation, has been inferred from the log-files data using the rules presented in Table 3. The time thresholds mentioned in the table were established on the basis of estimated time required for reading a page or performing a test.

Table 3. Rules for motivational level assignment

Disengagement	Engagement
Click-through pages (consecutive access-page events) with short time per page (less then 20 seconds)	Click-through pages (consecutive access-page events) with an average of at least 60 seconds per page
Very long time spent on a page/ test (above 10 minutes)	Reasonable time spent per page/test (between 1 and 10 minutes)
Automatic logouts from the system due to inactivity (for 30 minutes)	Lack of automatic logouts

We are aware that this way of assigning a motivational level to learners is a limitation for the results of the analysis. An external measure of the engagement or disengagement of the learner would be a more accurate base for prediction. In our further work an experiment will be conducted in order to externally validate the prediction.

There is an overlap between the indicators used for prediction and the indicators used for estimating engagement/ disengagement: time is used in both cases. Given the fact that in the first case the average time for reading and testing is used and that in the second case time thresholds for each page/test are used, the two types of time indicators are almost independent.

4 Analysis

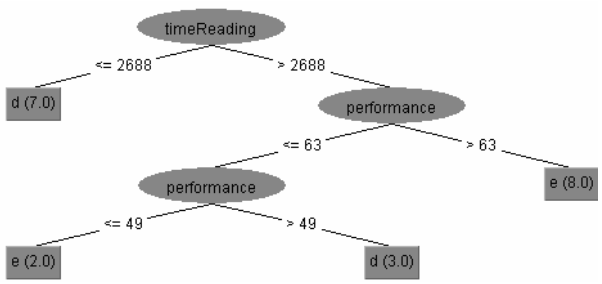
A number of 24 log files were randomly chosen for analysis and four of them were excluded due to very little information contained.

In order to perform the analysis, the Waikato Environment for Knowledge Analysis (WEKA) [Witten *at al.*, 1999] was used. The chosen method was decision trees based on C4.5 algorithm [Quinlan, 1993]. Other methods could be used, such as naïve Bayes classifier or regression. I chose decision trees and C4.5 algorithm because it provides classification and prediction, and also intelligible output in a graphical representation. Thus, the users' motivation can be characterized in terms of the attributes generated from the log files data (classification) and the predictability can be examined in order to see if such log file data can be used for motivation prediction.

In order to use the data for decision tree learning, each user has been assigned a motivational "state": engaged or disengaged. The criteria used for this assignment was described in the Table 3. The distribution of the 20 learners comprised 10 engaged and 10 disengaged.

4.1 The decision tree

The decision tree generated by WEKA for characterizing motivation is shown in Figure 1. The most important attribute for predicting motivation is, according to this decision tree, the time spent reading (timeReading): the users that spend less then 2688 seconds (approximately 45 minutes) are classified as disengaged; if the time spent reading exceeds 2688 seconds, performance is the second attribute to be used in classifying learners. Thus, if performance ratio is above 63%, users are classified as engaged. Otherwise, the same attribute, performance is used to classify learners as engaged if the ratio does not exceed 49% or as disengaged, otherwise.



e = engaged
d = disengaged

Figure 1.Decision tree for motivation

Summarizing the information from the decision tree, four categories of learners have been identified:

- learners who spend less then approximately 45 minutes reading; they are classified as disengaged;
- learners who spend more them 45 minutes reading and with a performance that exceed 63%; these learners are classified as engaged;
- learners who spend more then 45 minutes reading and with a performance between 49% and 63%; they are classified as disengaged;
- learners who spend more then 45 minutes reading and with a performance below 49%; they are classified as engaged.

4.2 The confusion matrix

The confusion matrix is presented in Table 4. It shows the quality of the decision tree and it has been produced by using fourfold cross-validation.

Table 4. The confusion matrix with fourfold cross-validation

		Predicted	
		Engaged	**Disengaged**
Actual	**Engaged**	8	2
	Disengaged	3	7

The elements in the matrix show the number of test examples for which the actual class is the row and the predicted class is the column. The diagonals of the confusion matrix indicate 75% of correctly classified examples and 25% on examples classified incorrectly. Thus, we can state that the quality of the decision tree is quite good.

Looking at the disengaged learners as they are our main interest, we see a lower rate of correct classification: 70% of the disengaged students are correctly classified.

4.3 Interpretation of results

Since the decision tree was derived only from a small set of examples, the results cannot be generalised in a straightforward manner. Another limitation is the way in which a motivational level, engaged/disengaged, was assigned to each user. It would have been ideal to have an external measure for this.

However, some interesting remarks can be made. The decision tree finds a particularly refined category of disengaged students: learners who spend a considerable time reading (above 45 minutes) and with a performance between 49% and 63%. Trying to give some meaning to these figures, a possible interpretation is the following: the fact that these learners have an average performance gives them a medium level of confidence; they go on reading, as they know they could improve their knowledge and performance, but knowing that they already have a medium or good knowledge level makes them invest less effort in learning. On the other hand, the results outline two categories of engaged learners that spend considerable time reading (over 45 minutes):

- The learners with a performance lower than 49%;
- The learners with a performance greater than 63%.

The engagement in both cases could be explained by the learners' desire to acquire more knowledge or just a better performance. From this perspective, it would be interesting to investigate the type of goal orientation of the learner (mastery / performance).

The results cannot tell anything about the users' level of motivation within the first 45 minutes. According to the decision tree, a user could be qualified as engaged or disengaged only after 45 minutes and, by that time, a demotivated user would have probably already logged out. Thus, it is of no benefit to know this information if there is no possibility to intervene. So, in order to be able to intervene on time, it is required to have information about the level of motivation in less time. This is also supported by the known fact that motivation can fluctuate at short periods of time.

In order to address the above mentioned aspects we intend to: 1) conduct a more detailed analysis using the data from the log files instead of derived indicators and 2) analyze the user's activity for short time periods – 10-15 minutes and extract the level of motivation for those specific times. By this approach information about the level of motivation would be updated at every 10-15 minutes and thus, have the possibility to intervene before the user would log out.

5 Summary and implications

We presented in this paper some results from a log files analysis. This analysis included a limited number of entries (20) and, thus, the results can't be generalised. However, it confirmed that a general indicator of the motivational level could be predicted from very basic data commonly recorded in log files.

This implies that a prediction module could be included in educational systems that record learners' actions. Looking at the two indicators found as predictors in our analysis – time spent reading and performance – the question that needs to be answered is if they depend on the system.

The threshold used for the time spent reading is approximately 45 minutes and the thresholds found for the performance were 49% and 63%. Because of the limited data used, the accuracy of the possible interpretations needs to be investigated in further work.

Another aspect to be investigated that emerged from our analysis is the level of motivation for short periods of time – 10-15 minutes – that would bring benefits in terms of intervention on time (before the user logs out) and taking in consideration the fluctuant nature of motivation.

Further work includes a larger scale and a more refined level of detail in the analysis, including the data from 150 log files. Also, an experiment will be conducted in order to externally validate the predicted motivational level.

References

[Arroyo *et al.*, 2004] Ivon Arroyo, Tom Murray, Beverly P. Woolf. Inferring unobservable learning variables from students' help seeking behavior. *In Proceedings of the workshop Analyzing Student-Tutor Interaction Logs to Improve Educational Outcomes*, pages 29-38, Maceio, Brasil.

[Beck, 2004] John E. Beck. Using response times to model student disengagement. In *Proceedings of the ITS2004 Workshop on Social and Emotional Intelligence in Learning Environments*, August 2004.

[Cocea, 2006] Mihaela Cocea. Assessment of motivation in online learning environments. In *Proceedings of Adaptive Hypermedia and Adaptive Web-Based Systems*, pages 414-418, Dublin, Ireland, June 2006.

[de Vicente and Pain, 2003] Angel de Vicente and Helen Pain. Validating the Detection of a student's Motivational State. In S. A. Cerri, G. Gouarderes, F. Paraguacu (Eds.), *Proceedings of the Sixth International Conference* on Intelligent Tutoring Systems, pages 933-943

[Heiner *et al.*, 2004] Cecily Heiner, Joseph Beck, Jack Mostow. Lessons on Using ITS Data to Answer Educational Research Questions. In Proceedings of the workshop *Analyzing Student-Tutor Interaction Logs to Improve Educational Outcomes at ITS 2004*, pages 1-9, Maceio, Brasil.

[Kay *at al.*, 2006] Judy Kay, Nicolas Maisonneuve, Kalina Yacef, Osmar Zaïane. Mining patterns of events in students' teamwork data. In *Proceedings of the Workshop on Educational Data Mining at the 8th International Conference on Intelligent Tutoring Systems (ITS 2006)*, pages 45-52, Jhongli, Taiwan.

[Keller, 1987] John Keller. Development and use of the ARCS model of instructional design. *Journal of Instructional Development*, 10(3): 2-10, 1987

[Martinez, 2003] Margaret Martinez. High Attrition Rates in e-Learning: Challenges, Predictors, and Solutions. *The e-Learning Developers' Journal*, June 2003.

[Merceron and Yacef, 2003] Agathe Merceron and Kalina Yacef. A Web-Based Tutoring Tool with Mining Facilities to Improve Learning and Teaching. In *Procidings of the 11th International Conference on Artificial Intelligence in Education AIED 2003*, pages 201-208, IOS Press.

[Mostow *et al.*, 2005] Jack Mostow, Joseph Beck, Hao Cen, Andrew Cuneo, Evandro Gouvea, and Cecily Heiner. An Educational Data Mining Tool to Browse Tutor-Student Interactions: Time Will Tell! In *Educational Data Mining: Papers from the 2005 AAAI Workshop*, ed. Joseph E. Beck, pages 15-22. Technical Report WS-05-02. American Association for Artificial Intelligence, Menlo Park, California

[Qu *et al.*, 2005]Lei Qu, Ning Wang, Lewis Johnson: Detecting the Learner's Motivational States in an Interactive Learning Environment. *Artificial Intelligence in Education*. C.-K. Looi et al. (Eds.), IOS Press, 2005, pages 547-554

[Quinlan, 1993] Ross Quinlan. *C4.5: Programs for Machine Learning*, Morgan Kaufmann Publishers, San Mateo, CA, 1993

[Witten *at al.*, 1999] Ian H. Witten, Eibe Frank, Leonard E. Trigg, Mark Hall, Geoffrey Holmes and Sally Jo Cunningham. "Weka: Practical machine learning tools and techniques with Java implementations." *Proc ICONIP/ ANZIIS/ANNES99 Future Directions for Intelligent Systems and Information Sciences*, pp. 192-196, Dunedin, New Zealand, November 1999.

[Zhang *et al.*, 2003] Zhang, G., Cheng, Z., He, A., Huang, T. A WWW-based Learner's Learning Motivation Detecting System. In *Proceedings of International Workshop on "Research Directions and Challenge Problems in Advanced Information Systems Engineering*, Honjo City, Japan, September 16–19, 2003.

Unwanted Behavior and its Impact on Adaptive Systems in Ubiquitous Computing

Michael Fahrmair, Wassiou Sitou, Bernd Spanfelner
Technische Universität München – Department of Informatics
Boltzmannstr. 3, 85748 Garching (Munich), Germany
{fahrmair|sitou|spanfeln}@in.tum.de

Abstract

Many ubiquitous computing applications so far fail to live up to their expectations. While working perfectly in controllable laboratory environments, they seem to be particularly prone to problems related to a discrepancy between user expectation and systems behavior when released into the wild. This kind of unwanted behavior of course prevents the vision of an emerging trend of context aware and adaptive applications in mobile and ubiquitous computing to become reality.

In this paper, we present examples from our practical work and show why for ubiquitous computing unwanted behavior is not just a matter of enough requirements engineering and good or bad technical system verification. We furthermore provide a classification of the phenomenon and an analysis of the causes of its occurrence and resolvability in context aware and adaptive systems.

1 Introduction

The main intention of ubiquitous computing (Ubicomp) is the use of functionality in as many situations as possible [Fahrmair, 2005; Dey, 2000; Weiser, 1991; Schmidt, 2002]. Context adaptation in this setting is an enabling technology for ubiquitous computing since it allows a technical system to change its structure, functionality or implementation at runtime to adapt to situation depending conditions. *Context* in this scope means the sufficiently exact characterization of a system's situation by means of perceivable information that is relevant for the adaptation of the system (a model of a situation). *Adaptation* again is a term to describe the general ability to fit to different conditions or circumstances given by the environment in a certain situation. *Context adaptation* is therefore shortly defined as an automated adjustment of the observable behavior or the internal states of a system to its context [Fahrmair *et al.*, 2006].

While classic non ubiquitous applications are used like hammers and screwdrivers and let the user decide how to put up a picture, ubiquitous applications are comparable to discreet servants. Our experiments clearly showed however that this paradigm shift seems to intensify problems related to user acceptance with such systems and their applications. Identifying and tracking back the causes, we observed that besides technical errors there was a new class of failure type involved. Sometimes the system even behaved exactly as intended by the developer, however for a certain combination of user, his situation and the environment, this behavior could be seen as wrong and even disadvantageous despite other users might have been perfectly satisfied.

To better distinguish this effect from failures that are undisputable an error, no matter what perspective or special situation you look at it from, we introduced the term *unwanted behavior* (UB). The main difference of this unwanted behavior to, for example, specification/implementation errors is that unwanted behavior can not be systematically detected by means of typical verification techniques like theorem proofing or model checking. In other words it is possible to have a 100% correct system that is still completely useless for a specific user in a specific situation. However since the system can work perfectly for most of all users most of the time, it is also not just a question of good or bad requirements engineering. Moreover since this phenomenon especially happens when releasing an application from the safety of the development lab (with its controllable environment) out into an infinite complex and unpredictable real world, this is also not a matter of just finding the right test cases during the development phase.

Context awareness is fine in theory, and the research issue is figuring out how to get it to work. Therefore, two applications, Grapevine and Rendezvous, developed and deployed by IBM, have revealed the key challenges in making context-aware computing a reality [Christensen *et al.*, 2006]. Grapevine and Rendezvous are services offered to IBM employees as a means of looking into the promise and perils of context-aware computing. The majority of users, however, did not find application activity context useful for a variety of reasons. For example, users often were not comfortable with others knowing what they were doing. The Grapevine [Richards and Christensen, 2004] service provided complete control over who could observe which elements of context, and users commonly blocked all others from viewing their computer activity all of the time. Although the service allowed observer-by-observer blocking, it was rarely used. The IBM Rendezvous service allows people to talk in small groups using telephones (to "rendezvous" on the phone) and/or computer applications that provide a telephone function. This is similar to audio-conference calls, and the service appears to be a layer on top of audio-conferencing. Instead of calling directly into an audio-conference, however, a user of the IBM Rendezvous service in effect phones his or her corporate calendar, selects a meeting from it, and enters into a multi party conversation with the people invited to that meeting.

In this paper we put all these and other observations and assumptions on a firmer theoretical and practical basis and discuss their technical and methodical resolvability for the domain of software and systems engineering:

- We give a brief overview (Section 2) why unwanted behavior is an intensified problem for ubiquitous applications that are realized with context adaptation. We describe both examples of existing real life as well as academic prototypes and concepts to illustrate the above mentioned unwanted behavior problems (Section 3)
- Although it is superficially possible to classify the described examples into three different types of failure reasons, our analysis (Section 4) shows that the real cause for UB is always a divergence between two models of reality.
- There are exactly three sources for such model divergences. We explain why such discrepancy can survive most technical quality measures, stay hidden in the system for a long time or even spontaneously arise while running the system (Section 5). We also do a short discussion on what constructive or methodical measure can be taken to avoid unwanted behavior and how these strategies worked out in our practical applications.

2 Impact of Unwanted Behavior (UB) on Ubiquitous Computing

The main idea behind all these concepts in particular and of ubiquitous computing in general is a more flexible system understanding, whereby the thought of the system as a tool moves into the background and the needs and wishes of the user step into the foreground. Generally, these needs and wishes of a given user vary according to his current situation.

During the last six years we have developed and experimented with several prototypes of applications based on that idea of systems that can automatically recognize wishes and needs of its users (or other stakeholders) and adapt themselves accordingly by means of reconfiguration. Among them were a mobile community based search engine [Fahrmair *et al.*, 2000], an in-house navigation and information assistant for a campus [Amann *et al.*, 2004] and a one-year long self experiment with a smart home environment [Fahrmair, 2005].

From the introducing description of UB, there seems to be no reason why UB cannot arise in non ubiquitous systems. In fact this assumption is true. However ubiquitous applications usually

C1: are multi functional and more complex, and operate in heterogeneous environments,

C2: work, at least to some extent, invisibly in the background, and

C3: are technically based on automation (context adaptation)

Because of C1, Ubicomp systems are much less transparent for their users, especially regarding technical limitations of possible functions. Due to C2, there is no such thing like an operating error in Ubicomp. Therefore the user rightly insists that a Ubicomp application fulfills his wishes and needs as promised and not just does what he has explicitly commanded. This difficulty even gets worse if the system relies on automation (C3) to fulfill its ubiquity goal, even in situations with very limited user interaction possibilities (e.g. while driving a car).

This is because, even if UB can also occur in normal systems, they usually have a spontaneous hull [Raasch, 1993] that can help to prevent or compensate for negative UB experience. A spontaneous hull in short is some kind of interactive influence sphere around the actual technical system core. This user influence can be used to compensate for known or anticipated UB for example by manually modifying input and output values, finding operational workarounds for bugs etc. With increasing automation and decreasing user interaction resources in ubiquitous systems, this mechanism however can no longer compensate UB below a tolerable level.

3 Examples of Unwanted Behavior

In this section we provide a couple of examples illustrating the occurrence of unwanted behavior while dealing with software systems, particularly if the systems possess certain automatic (i.e. context adaptive behavior). Some of these examples are constructed. Since not many ubiquitous systems have been released, the examples aim to illustrate unwanted behavior also in common applications. We should point out that in Ubicomp the occurrence of UB is however amplified by the ubiquitousness of the applications.

3.1 Microsoft Office Spell Checking

A well-known, yet very simple, example of unwanted behavior is the automatic spell checker incorporated in MS Office Word. After a full stop, the first letter of the following word is automatically capitalized. In fact, this function cannot differentiate between abbreviations with point and the real ending of a sentence. If this fact is not familiar to a given user, the written text could hold some surprises during a read-after-write check. This is an example for situations, where the system is not able to predict the users intentions. Of course no one would care about this if it occurs once in a while but, as Ubicomp applications are meant to support the user in as many situations as possible with automated decisions, such small glitches can sum up to a big annoyance.

3.2 Smart Kitchen

A nearly inexhaustible source for effects of unwanted behavior are smart home applications as described in [Fahrmair, 2005]. This occurs particularly if the user can not understand the exact technical realization of the system. A typical example concerns smart kitchens with their underlined automatic food order systems. A lowbrow user for instance could develop the impression that the ordered food is only based on the editable purchasing list. In fact this observation is however only a coincidence, which arises as a result of the fact that the user does not transact any additional purchases, which would be registered by the system over the RFID labels and considered for new orders. If the user suddenly changes his relevant behavior, for example by adjustment of a weekly poker party, for which the guests bring some food and put it in the refrigerator, this dependence hidden so far can lead to terrible surprises. The system would register the regular consumption of additional goods and would adapt the automatic order accordingly. This example highlights problems that occur because of a misunderstanding of the systems inner behavior. More accurate or better understandable documentation in this case is not necessary sufficient, because Ubicomp systems adapt their structure and functionality at runtime.

3.3 Navigation System

Lets consider a common GPS-based navigation system. Such a system has as primary task to guide its user from a location A to another location B. At the beginning of the guidance, the system computes the route. Thereby it considers in addition to the current input of the user, also his preferences, and guides him to the desired destination. If the user gets lost during the guidance, the system recomputes the route from the current position to the destination. In our example, a building site that recently began is on the computed route. This building site however is not yet taken up into the street guide. The system is therefore unable to recognize this new situation and guides the user into this dead-end street. The user is unsatisfied, turns 400m back and takes another way. After a while, the user lets the navigation system guide him again to his destination with the hope the detour would cause the system to choose another route. Once again, the system guides the user to the temporary dead-end street at the building site. The user had to go back even further to let the system definitively avoid the building site. Of course there are already new navigation systems that allow the user to exclude specific streets. This requires additional user interactions in a situation where the user is already busy driving the car. This example illustrates unwanted behavior, where the system, in contrast to the user, does not have the necessary abilities to detect exceptional situation.

3.4 Further Examples: Air-conditioning, Bank Service

Other examples for unwanted System behavior would be a climate control that does adapt to the cultural background and a user that has guests from abroad and suddenly is disappointed about the system adapting to the guests, due to the majority criterion of the adaptation logic, and not to him. Here the user was used to a certain behavior that apparently changes spontaneously. A further example is an automatic savings function for the bank account that saves money above a certain threshold and accidentally saves money that someone has assigned on the account to pay a certain anticipated bill.

4 Characterization of Unwanted Behavior

We use the term *unwanted behavior* to designate the phenomenon where the behavior of a given system, while free of errors, still differs from the expectations of its current user.

4.1 Criteria for the Occurrence of UB

Analyzing the above mentioned motivation and examples, we summarize that unwanted behavior occurs if the following criteria are all together fulfilled:

- There exists an observable divergence between user expectation and system behavior.
- The system behaves correctly regarding its specification
- The system specification complies with the collected requirements.

The existence of the observable divergence between user expectation and system behavior is with reference to the above mentioned observations undisputable. The idea of drawing a distinction between user model and systems model is similar to Norman's canonical elucidation of the role of mental models in the design process [Norman, 1988]. Norman states that the designer's goal is to design the system image such that the user's mental model of the system's operation coincides with the designer's mental model of the same. The system image represents those aspect of the implementation with witch the user interacts. Yet the above mentioned discrepancy, i.e the observable divergence between user expectation and system behavior, could not be seen as system construction failure, since the system exactly behaves as specified by the engineers. The main cause of the occurrence of such unwanted behaviors therefore seems to be a lack in collecting and processing the users needs and wishes.

We derive from these criteria that an unwanted behavior occurs if, on the one hand, the user is not aware of the system's abilities and thus develops expectations that are unrealizable by the system. On the other hand, UB occurs if the needs of the user are altered due to external influences, which are unrecognizable by the system. Over and above that, UB could be registered if situations (or context) of use arise at runtime, which were not predictable at development time. The system then proceeds from assumptions that might not be valid any longer and thus behaves suddenly incorrectly from the users point of view. In this way UB-occurrences are events that individually usually are not critical for the overall functionality of a system. In larger quantities, UB can become a growing annoyance though. This can lead to rejection by the user and therefore to a replacement of a system.

4.2 Cause of Occurrence of UB

Events that meet our definition of UB can be traced back to three reasons (see Figure 1).

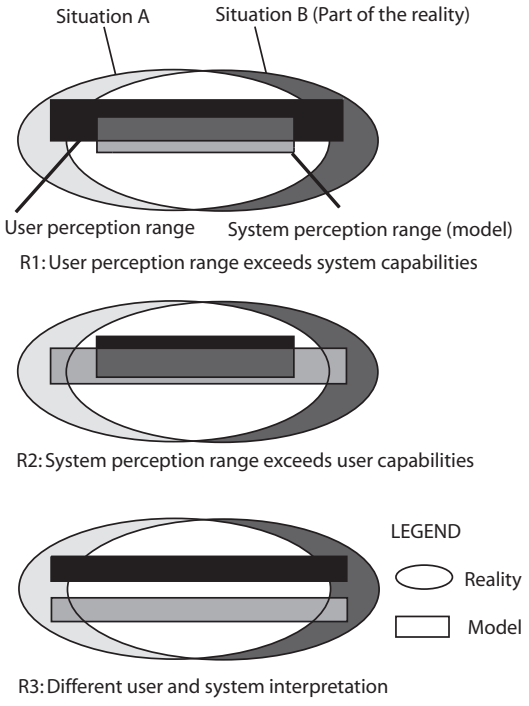


Figure 1: Differences in the Perception of Different Situations

These reasons are sometimes sole responsible for a UB-Phenomenon but can also occur in combination:

R1: The user is able to differentiate two situations, but the system is not.

R2: The system is able to differentiate two situations, but the user is not.

R3: Both, user and system are able to differentiate the situations but they have different interpretation or perspective of the situation's changes.

R1 is typical for ubiquitous systems that lack needed sensors to identify a certain situation. This can either be the case if the situation was not considered during the requirements engineering and hence a required sensor was not integrated in the system. Another possibility is that sufficient exact sensors are not available, or even if so, they are too expensive etc. In either case the context information that is available to the system is not sufficiently exact to characterize all relevant situations. Since context is an abstraction of the reality where irrelevant details are dropped, a third possibility for R1 could be that the system model is too abstract and details that are important for the user may have been dropped.

R2 is the exact opposite of R1. If a system uses sensors that observe the environment in terms that are beyond the perception of the user, it is possible that the system will identify a new situation where the user is not aware of any changes. Another possibility could be that the system model is too detailed. Sometimes little changes in the environment are perceptible by a user per se, but in normal life these details are filtered out. If the system model observes such details (for example in discrete rules) it is likely that the system identifies a change of the situation but the user does not. Also the system designer could be of the opinion that a certain change in the environment leads to a different situation but the user does not share this opinion. Since ubiquitous systems are commodity, it is likely that some user's opinion about different situations differ from the designer's opinion. In contrast to a too detailed model where the designer did not bear in mind that certain details are irrelevant to a user, here there is a basic difference in identifying situations.

R3 is a bit related to the different interpretation of a situation. Here the user as well as the system designer had different interpretations of a situation. The main difference compared to the last reason is that in the former argument the difference lies in the importance to distinguish between two situations whereas now both agree that a new situation is recognizable but they differ in the interpretation of the new situation. This could be the case if, for instance, the cultural backgrounds of the designer and the user are different.

Despite the reason for UB phenomenon differs, there is a common fact that is equal to all three reasons: The user's reality model, from which he derives his wishes and needs, differs from the system model that represents the reality model of the designer.

5 Model Discrepancies (MD) and their Origins

Now that we have identified MDs as the main reason of UB in an otherwise technical error free system, the interesting question is how and when such MDs are created and whether they can be detected, removed or at least avoided or compensated for.

To understand the difficulty of this question we have to make clear that this question is not about comparing two technical models, which is often done in software verification (e.g. checking design specification model against implementation) and is at least theoretically feasible. The UB problem is about comparing models with reality, i.e. checking whether the assumptions that were used to *create* the model in an abstraction process have been valid in the first place and are still valid. In ubiquitous computing this means comparing a black box model of yet unknown type and structure inside the users head with a similar model of the developer of the system. While the model of the developer is at least partially visible in form of the system implementation, the user model is not. To make it more difficult, the developer's model representation is frozen at a certain time during the development phase while the user's world model constantly changes and should be updated to reflect any unforeseeable changes in reality, his growing experience etc. Yet the user's model is still only a projection of reality from a certain perspective and of course this perspective is not necessarily the same as the developer's.

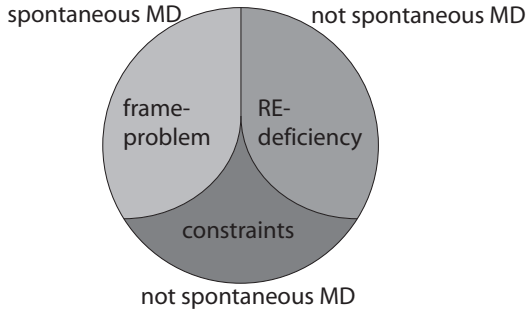


Figure 2: Three Reasons for Model Discrepancies

It follows from this that MDs either already exist at development time (hidden MDs) or (spontaneously) emerge during the system lifetime (see Figure 2). The latter possibility is not as obvious so we first take a look at hidden MDs that are introduced somewhere within the development process. Our analysis of several case studies clearly showed that these hidden MDs can have two possible origins.

The first possibility is that the UB is caused by a MD that originated from a *conscious decision by the system designer*. For example, the designer realized that among 1000 possible usage contexts there might be 2 with use cases and requirements conflicts that would need an uneconomical amount of effort to be resolved. So the decision was made to deliberately drop support for this situation. However the first problem with such decisions in a ubiquitous application is that ubiquitous systems usually are extended, recombined or depended on in an unforeseeable way. Even if a failure rate of 0.2% seems to be pristinely tolerable, such effects can propagate and multiply due to service composition (typical for Ubicomp systems) rendering later applications of the system pretty useless. It could be argued that this problem is more or less a matter of writing and reading proper manuals. Yet the problem is that even if this decision was explicitly made, even documented, this information does not always make it through the development process. Even if it would in Ubicomp (because of its dynamic nature) there is usually no printed manual.

The second possibility for hidden MDs is *deficits in requirements engineering* (RE). In our terminology we deliberately speak of deficits instead of errors because we distin-

guish between real errors (like registering and documenting a requirement in a document that is later on overwritten by another version or skipping that annoying interview etc.) and "did not know it better at that time" effects. So the reason is that state of the art RE methods seem to be not fully suitable for the construction of complex Ubicomp applications that work in highly heterogeneous dynamic environments.

As mentioned before there is a third possible reason for MD. This one is especially though to handle because the MD does not exist during development time but arises later. This last condition however is not undisputable. The whole matter is under heavy discussion for more than 20 years especially in the field of AI (known there as the frame problem [Dennett, 1984]). However there seem to be a lot of good arguments that this problem is not generally, and now less then ever practically, solvable at least until it is possible to give a machine at least limited abilities in mind reading and fortune telling. Another possibility would be to avoid using classical model representations (with extrinsic semantic) at all. Using a model representation based on self contained semantic (like the mathematic language is for mathematics) to reflect a significant part of the real world however would most likely mean constructing a machine with higher complexity than our universe.

We therefore assume that there are certain random events within reality or the user's thoughts and ideas that can cause MDs at a later point in time than the development of the system (see Figure 3). Moreover because of the symbol grounding problem [Harnad, 1990] such MDs can not be detected from inside the model without any outside help. This means MDs can stay hidden until they cause UB or at least become imminent.

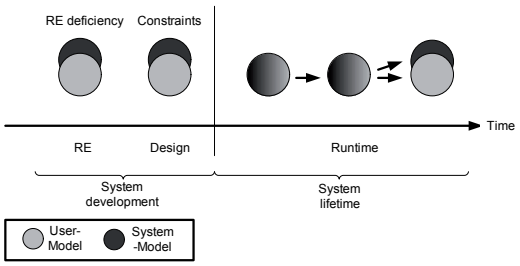


Figure 3: Model Discrepancies and their Origins

6 Related Terms

While dealing with the phenomenon described above, the question arose of whether we really need a new identifier? And is there any relation to terms like classical system error, operating error, feature interaction, automation surprise or software aging?

6.1 Classical System Error

An often asked question regarding the whole issue of unwanted behavior caused by model divergences is, whether this is not just some kind of classic error. Especially because for instance, an implementation error could also lead to wrong results in a software system, which is of course also disliked by the user.

The main difference however is that such errors usually are unacceptable for all users no matter what situation they are in. Also a system with errors is usually regarded as incorrect compared to a formal specification, which makes errors at least theoretically possible to dispel at development time by various means of different verification techniques.

Our definition of unwanted behavior therefore purposely excluded such technical errors because we differentiate between failures that can be prevented by means of verification and failures that can not be systematically detected during development time. Basically this is inspired by the fact that there can be systems that are 100% correct but can still be completely useless in a certain situation. This is also why MDs can not always be detected at development time. Unlike classical system errors, MDs are hidden in implicit assumptions (in other words the abstractions) that were used to create a model in the first place.

6.2 Operating Error

The only other form of errors not covered by this differentiation between UB and specification errors are operating errors. These describe failure situations where the system would depend on correct user interaction to produce a correct output behavior. Ubiquitous applications however usually work invisibly in the background, at least to some extent and therefore have to be foolproof by definition, at least regarding any remaining direct user input. This is why we neglect operating errors as a source of UB provided that they are not already covered by MDs in general, for example when the user has developed a wrong idea about the capabilities of the system.

6.3 Feature Interaction and Automation Surprises

Besides classical system and operating errors there is yet another term that is often confused with MD created UB. *Feature interaction* however relates to a large group of mostly technical issues around interaction and combination of features (for a good overview see [Pulvermüller *et al.*, 2002]). Inside this group especially unwanted feature interactions not caused by specification/implementation errors come quite close to unwanted behavior, but MDs are not limited to a combination or interaction of features. Therefore feature interaction problems that are not related to errors are a subset of our definition of UB.

The same is true for *automation surprises* or so called *mode errors* [Hourizi and Johnson, 2001]. These are also small subsets of UB that especially concentrate on user interfacing aspects. Moreover automation surprises only concentrate, on system behavior non-transparency and makes no difference between unexpected and unwanted behavior. Mode errors again require at least some sort of direct and conscious user interaction, which is not always the case for ubiquitous systems. We therefore regard both terms as very special cases of UB. In fact they describe rare cases of non ubiquitous applications where UB can have a larger impact (e.g. in avionic systems).

6.4 Software Aging

Finally the term *software aging*, while seeming quite related especially to MDs accrued during the system's lifetime, usually focuses on increasing difficulties to change an old system to new requirements (or to cope with an increasing number of such changes afterwards [Parnas, 1994]). The reasons that requirements can become obsolete of course are the same as MDs appearing during runtime of an ubiquitous system. The main difference between MD based UB and software aging however is that the latter

needs several years or even decades of lifetime to evolve relatively stable changes in the wishes and needs of their users while spontaneous MDs in ubiquitous applications can happen during runtime and can be instable. Also they usually heavily depend on the specific user and his situation, while software aging usually refers to a collective change in requirements (e.g. like supporting a new technology, platform or process).

7 Conclusion

In this paper, the phenomenon of unwanted behavior (UB) has been characterized and examples from our practical work have been given. We have also provided a classification of the phenomenon and analyzed the causes of its occurrence and resolvability, particularly in the filed of context awareness and ubiquitous computing.

While researching user acceptance problems in real life ubiquitous computing applications, there always has been one question at issue: Is the problem of unwanted behavior really a new kind of problem? Is it a specific drawback of ubiquitousness or a question of good (or bad) requirements engineering? Unfortunately the answer is more complicated than the question. Can good-enough requirements engineering prevent from UB? Yes and no. Yes, because the main goal of RE is to analyze the user wishes and needs and prepare them to be translated into a technical specification for system design. No, because wishes and needs change over time depending on necessities of reality. Therefore, while analyzing requirements, at least enough resources, a good clairvoyance and adequate methods would be needed. But, what is exactly "good enough" RE? What is it all about? For ubiquitous computing this means the more wide the application domain gets (both in lifetime and complexity), the more probable it is that there exist discrepancies between user expectation and system behaviors. Each time this happens, a UB will be created. This is a user experience problem in Ubicomp because of the claim to fulfill user wishes and needs without necessary direct or conscious interaction. Unconscious usage means automation and hence less possibility to mediate or real-time correction by the user.

However the system model could have been wrong from the beginning due to deficits in RE, but also MDs can arise later due to framing problems. Because the user is also usually a customer of some sort, for him it is not his concern if this UB is caused by a bad RE analyst or a tricky twist in reality. Therefore it is necessary to at least introduce another compensation mechanism to replace any real-time compensation possibilities the user had in non-automatically acting applications. This mechanism is called calibration and described in [Fahrmair, 2005; Fahrmair *et al.*, 2006; Newberger and Dey, 2003]. Once again, this mechanism should be no excuse for relaxed RE. There is a number of reasons why as much effort should be spent on RE as economically possible to reduce the number of UB caused by failed RE since the calibration is not a totally preventive mechanism (every engineer probably used to sit in an airplane from time to time). However there are also remarkable indications for not delivering ubiquitous application without calibration support. And last but not least there are also a number of good reasons to not scrap ubiquitous computing all together because for instance it allows computers to extend their full potential from direct human machine interactions to sub- or semi-conscious secondary usage in almost every situation.

References

[Amann *et al.*, 2004] K. Amann, T. Reichgruber, and M. Roming. *Personalisierung kontextadaptiver Dienste*. Student Project, Technische Universität München, 2004.

[Christensen *et al.*, 2006] J. Christensen, J. Sussman, S. Levy, W. E. Bennett, T. V. Wolf, W. A. Kellogg. *Too Much Information*. HCI, ACM Queue 4(6), 2006.

[Dennett, 1984] D. C. Dennett. *Cognitive Wheels: The Frame Problem of AI* . Ed.:C. Hookway, Minds, Machines, and Evolution. Cambridge University Press, Cambridge, 1984.

[Dey, 2000] A. K. Dey. *Providing Architectural Support for Building Context-Aware Applications*. PhD Thesis, College of Computing, Georgia Institute of Technology, 2000.

[Fahrmair, 2005] M. Fahrmair. *Kalibrierbare Kontextadaption für Ubiquitous Computing.* Dissertation, Department of Informatics, Technische Universität München, 2005.

[Fahrmair *et al.*, 2000] M. Fahrmair, C. Salzmann, and M. Schoenmakers. *Verfahren zur Vorauswahl mobiler Dienste*. IPR DE0010024368A1 [DE], DPMA, 2000.

[Fahrmair *et al.*, 2006] M. Fahrmair, W. Sitou, and B. Spanfelner. *An Engineering Approach to Adaptation and Calibration*. Modeling and Retrieval of Context MRC 2005, Ed.: T. Roth-Berghofer, S. Schulz and D. Leake, LNCS 3946, 2006.

[Harnad, 1990] S. Harnad. *The Symbol Grounding Problem*. Physica D 42, 1990.

[Hourizi and Johnson, 2001] R. Hourizi, and P. Johnson. *Beyond Mode Error: Supporting Strategic Knowledge Structures to Enhance Cockpit Safety*. Joint Proc. HCI 2001 and ICM 2001.

[Newberger and Dey, 2003] A. Newberger and A. K. Dey. *Designer Support for Context Monitoring and Control*. IRB-TR-03-017, Intel Research Berkeley, 2003.

[Norman, 1988] D. A. Norman *The Psychology of Everyday Things*. Basic Books, New York, 1988.

[Parnas, 1994] D. L. Parnas. *Software Aging*. 16th Int. Conf. on Software Engineering (ICSE-16), 1994.

[Pulvermüller *et al.*, 2002] E. Pulvermüller, A. Speck, J. O. Coplien, M. D'Hondt, and W. DeMeuter. *Feature Interaction in Composed Systems*. LNCS 2323, 2002.

[Raasch, 1993] J. Raasch. *Systementwicklung mit Strukturierten Methoden. Ein Leitfaden für Praxis und Studium*. Hanser, 3. Auflage, München, Wien, 1993.

[Richards and Christensen, 2004] J. Richards and J. Christensen. *People in our Software*. ACM Queue 1(10), 2004.

[Schmidt, 2002] A. Schmidt. *Ubiquitous Computing - Computing in Context*. PhD Thesis, Computing Department, Lancester University, U.K., 2002.

[Weiser, 1991] M. Weiser. *The Computer for the 21st Century*. Scientific American, 1991.

User Profiling and Privacy Protection for a Web Service oriented Semantic Web

Nicola Henze and Daniel Krause
Distributed Systems Institute, Semantic Web Group, University of Hannover
Appelstraße 4, 30167 Hannover, Germany
{krause,henze}@kbs.uni-hannover.de

Abstract

In a Web Service-based Semantic Web long term usage of single Services will become unlikely. Therefore, user modeling on Web Service's site might be imprecise due to a lack of a sufficient amount of user interaction. In our Personal Reader Framework, the user profile is stored centrally and can be used by different Web Services. By combining information about the user from different Web Services, the coverage and precision of such centralized user profile increases. To preserve user's privacy, access to the user profile is restricted by policies.

1 Introduction

Adaptation has been proven to be able to massively improve users' satisfaction with online services. An expressive example is Amazon, which extensively uses personalized recommendations and became one of the largest online bookshops. One very important part of all advanced adaptation methods are the – as precise as possible – information about the user in a user profile. Today two classes of methods for generating such a user profile are widely used: Profile learning techniques using observations about the user to implicitly model the user, or information which has been directly provided by the user, for example via a questionnaire.

If a user interacts over a long time with an online system, both techniques perform well: On the one hand profile learning approaches get enough input from the user to generate an appropriate user profile. On the other hand users are more willing to fill in a questionnaire after they attained confidence in a system by using it over a longer period of time.

If we think about a Web Service-oriented Semantic Web, this long term usage of single Web Services will not be the standard case any more. Users are looking for Web Services that fulfil their actual requirements and immediately want to use them. After their task is performed users may never use this Web Service again. In such a high dynamic environment single Web Services do not have the time and sufficient users input to generate an appropriate user profile on their own.

According to this assumption we present a framework for a Web Service-accessible centralized user profile allowing different Web Services to collaborate in the task of user modeling. By storing user profiles in a trustful independent system, this approach also allows the user a comprehensive policy-based control of his user profile to retain his privacy.

2 The Personal Reader Framework

The Personal Reader Framework [Abel *et al.*, 2005; Baumgartner *et al.*, 2005; Henze and Kriesell, 2004] provides users with a unique access point and single login to a Web Service-based Semantic Web and preserves privacy protection by offering a policy-based usage of the sensitive user information. Web Services thus can – if trustworthy enough – share information about the user, but still the user is in full control of the shared data and can anytime restrict or extend the access to the data on a per-Web Service base. This results in better user comfort as eventually required initial user profile creation period takes time only once.

2.1 Architecture

The Personal Reader Framework is divided into four main components:

- Syndication & Visualization
- User Modeling Service
- Connector
- Personalization Services: Web Services that offer a certain personalization functionality

The syndication (for short *SynService*) and visualization components are responsible for combining and integrating content generated by the Personalization Services, for visualizing the content in an appropriate User Interface, and for assisting the user during the discovery, selection and configuration of Personalization Services, for short *PServices*. The Connector (*CService* handles the communication flow between all stakeholder in the architecture. The User Modeling Service (for short *UMService*) is responsible for obtaining user's privacy by restricting access to the user profile by policies. Thus, only authorized Web Services can access the user profile.

2.2 Syndicated access to Personalization Web Services

For provide a unique access point to Personalization Services, a discovery process for appropriate, available PServices is necessary. This is realized by the centralized CService, which accesses one or several UDDI broker to obtain Web Service descriptions (including a RDF description of their provided functionality, and a list of invocation parameters and their description).
The descriptions of the available PServices are used by the SynServices to generate a portal were users can select those

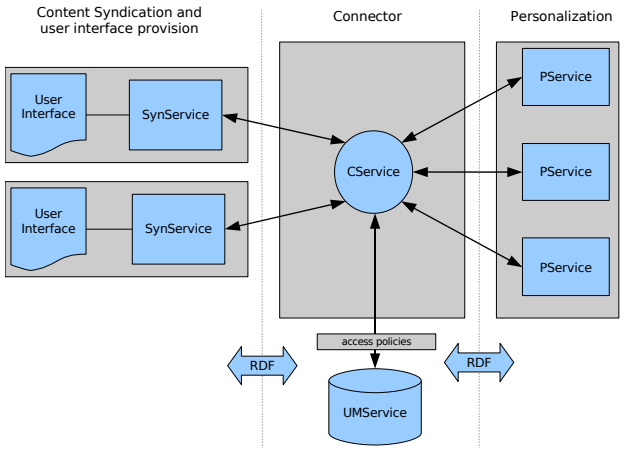


Figure 1: Simplified architecture of the Personal Reader Framework

Services which suit best their requirements. Thus, this portal represents a single access point to the available Web Services.

Negotiation

Before a user can invoke the Web Services he selected from the portal, their invocation parameters must be set. The SynService tries to set these invocation parameters automatically by setting them according to values stored in the user profile. Therefore, the SynService sends a request $requested(W, P)$ for every invocation parameter P of the Web Service W to the UMService. The UMService should return the value V of parameter P together with a semantic description of the value (for example the value is three at a scale from one to five where one expresses highest interest in P, see [Heckmann, 2005]). To preserve privacy, the User Profile Manager evaluates this request according to an Event-Condition-Action Rule (ECA) [Bailey *et al.*, 2004] and returns the requested value only if the condition is fulfilled:

r1: *on* requested(W,P)
if readAccessAllowed(W,privacyProtection(P)) $\wedge$ confidence(P,V) $>$ threshold
do return(P,V)

On represents the event, if the condition and do the entire action.
This rule expresses that value V of parameter P is returned if the confidence in (P, V) in the user profile is higher than $threshold$ and W is allowed to access P. $PrivacyProtection(P)$ expresses the policy representing access restrictions to P. By using policies the user can describe his privacy restrictions very detailed and is able to group several Web Services and invocation parameters, too.

For example a user can specify in his policies that all Web Services that were certified by some trusted authority can access his user profile. Or a per-parameter-base access can be realized, where access to invocation parameter P is granted to Web Services that already have access to a similar invocation parameter P'.

User interaction

If the access is denied, a SynService has different options to handle this:

- Ask the user whether he wants to grant access or not. After the user made a selection, the policy of the according invocation parameter P is adjusted to automatically allow or deny further accesses to P from this Personalization Service.
- If alternative PServices are available whose invocation parameters can be automatically filled, use only those Services.
- If denied invocation parameters are all marked as optional, try to invoke the PService without these parameters. If the user dislikes the result ask him to grant access.
- Deny access.
- Other user defined actions.

These different options enable the user to choose whether he wants to be disturbed in order to adjust policies or not (with the fact of loosing some content), and are important to preserve the usability and trust in the whole Personal Reader tool.

According to the specified user policies, there are three cases in which an invocation parameter P cannot be accessed by a Web Service W:

1. The policy denied access to P from W
2. $threshold$ is defined and confidence of P is lower than $threshold$
3. P does not exist in the user profile

Every case leads to the action that the SynService will take care on the missing invocation parameters as described above. If all invocation parameters are configured, PServices are invoked and their delivered contents are syndicated and visualized in an appropriate representation for the end user.

2.3 Collaborative Access to User Profiles

The access policies in combination with the above defined ECA rule require user interaction, if the user accesses unknown new PServices. In this case the automatic discovery of user profile informations fails and the user is asked how to proceed. If the user accesses these new PServices often – as we expect – it can lead to usability disadvantages as a result of frequent user interaction.

Our solution to cope with this issue is to let users define other users they trust in. If these trusted users U' allow a Web Service W to access their user profile to receive an invocation parameter P, the UMService can automatically allow access to this data from the user profile of User U, too. In this case ECA rule *r1* is extended to:

r2: *on* requested(W,P)
if [readAccessAllowed(W,privacyProtection(P)) $\vee$ userProfile(U').readAccessAllowed(W,privacyProtection(P))] $\wedge$ confidence(P,V) $>$ threshold
do return(P,V)

Additionally, we can share user profiles between different users by such a collaborative approach. This can be established in the same manner as the shared trust:

r3: *on* requested(W,P)
if readAccessAllowed(W,privacyProtection(P)) $\wedge$ [confidence(P,V) $>$ threshold $\vee$ userProfile(U').confidence(P,V) $>$ threshold]
do return(P,V)

For discovering possible candidates for collaboration we use FOAF (friend-of-a-friend) files to construct a social network. Furthermore, we can apply user profile matching techniques to find similar users for collaboration.

2.4 User Profile Maintenance

User profile maintenance is handled by the UMService, this includes tasks like storing users' information and metadata like which PService was responsible for which changes/information. Furthermore, the UMService restricts access to the sensible user profile information for unauthorized Web Services by applying access policies. These restrictions are divided into read access where Web Services try to read some information from the user profile, and write access where Web Services try to update existing user profile content, or create new user profile content.
The following two ECA rules – *r4* for read access and *r5* for write access – express these access policies:

r4: *on* readAccess(W,P)
if readAccessAllowed(W,privacyProtection(P))
do return(P,V)

r5: *on* writeAccess(W,P,V)
if writeAccessAllowed(W,privacyProtection(P))
do createEntry(W,P,V) ∨
updateEntry(W,P,V)

If a Web Service tries to write to or read from the user profile, the User Profile Manager checks if the necessary access privileges do exist. If this is not the case and default access privileges are not sufficient to return the requested information, access is denied.

Our approach allows to store user profiles in a centralized place. Every Web Service can access the user profile via the UMService. As this storage is not placed on Web Service's site, the user is – at every time – in full control of his personal information. Furthermore, the collected information can be used also long term, because they are kept in the user profile even if a Web Services disappears. As a result, even a high dynamic environment, where new Web Services appear and old Web Services disappear frequently, does not cause loss of user information.

Distributed User Modeling

The user profile contains domain-specific information, for example a music recommender will probably store information about music objects, another higher-class music recommender stores information of inferred user's preferences and a third Web Service, an e-learning Service, stores information about learning objects. This domain-specific content of the user profile makes it hardly possible to do centralized user modeling. A centralized approach would need a user modeling component that has domain-specific knowledge of all known Web Services. But this would limit capabilities of easily integrating new Web Services of unknown domain as they would require the update of the user modeling component.

So our approach relays on a per-Web Service-user-modeling: Each Web Service can gain write access to the user profile: it can use it's own user modeling techniques to derive new information about the user, and write the results directly to the central user profile. For storing these informations a techniques like proposed in [Heckmann, 2005] will be used. Other problems that occur in a centralized user profile, like conflict handling, can be solved

The advantage of this approach is that the complete implementation of the User Profile Manager is domain-independent, and enables any kind of Web Service to interact with the Personal Reader Framework without updating its components.

3 Proof-of-Concept

Assume a user is searching for music recommendations in the genre rock. The Connector has discovered two different music recommender Web Services the user does not know:

- the first Web Service returns non-adaptive rock music recommendations
- the second Web Service provides adaptive common music recommendations

As the user is only interested in rock music recommendations he considers the rock music recommender Web Service as most appropriate and invokes it. This Web Service returns in its response a list of recommendations. While the user browses through the list of recommendations and listen to music he likes, the rock music recommender Web Service assumes that the user likes songs a, b and c most. To share these information the Web Service tries to store the following information in the user profile:

User likes songs a, b and c

The rock music recommender is not known to the user, and we assume that the user has set as a default "no write access" for all unknown Web services. In consequence, rule *r5* denies write access to the user profile. Thus, the user is asked if the Web Service should be allowed to alter his user profile. The user accepts this and further write access and, as a consequence, the according policy is updated to allow further write accesses automatically.

For whatever reasons, the user decides to invoke the common music recommender, too. This Web Service first tries to access the user profile to get an answer for the query:

Which music style is preferred?

The access is blocked by default (rule *r1*) and again the user is asked if he allows access to his data, and again he grants access. But no information about the preferred music style of the user is stored in the user profile. Therefore, the music recommender tries to get this information on another way by querying the user profile again

Which songs are preferred?

As the Web Service already has the permission to access similar parameters to those requested in the new query, and the user-controlled policy automatically allows access to similar parameters (rule *r1* with additional constraint), the user is not asked again whether the Web Service should be allowed to access the requested information. Because the Web Service gets the answer 'a, b and c are preferred songs', it can infer from its internal knowledge base that these music titles belong to the rock genre. Thus, it adapts its results by recommending only rock music. Later on, the music recommender might infer from its observations of user interaction that this assumption is true. Now it sends a write request to the User Profile Manager to insert the fact that the user likes rock music (rule *r5*). And, after negotiations with the user, the profile is updated.

This example shows that the second Service used the observations of the first Service to adapt its content according

to user's interests. Additionally, new generated information from the second Web Service is stored in the user profile and can be accessed by succeeding Web Services.

3.1 Demonstration

A working demonstration is presented in [Abel *et al.*, 2006]. In this demonstration we have already implemented a configurable Web Service, called MyEar. MyEar is a podcast recommender that can be configured according to:

- keywords in podcast description
- duration of podcast
- genre

We have implemented a visualization template, called MyEarView, which is used by the Syndication & Visualization Component to visualize the results of MyEar. At the moment, the user modeling is limited and stores only the previously made configurations of Web Services. Thus, the user does not have to set invocation parameter again if he uses an already configured Web Service. We are currently working on the extension of the user profile manager according to the ideas described in this paper.
The demonstration application is accessible via:
http://www.personal-reader.de/agent/

4 Related Work

Research for user-driven access to the Semantic Web currently focusses on two different approaches. The first approach visualizes RDF files without taking into account their content. Examples are Piggy Bank[1], Longwell[2] or Brownsauce[3]. These browser are more appropriately called RDF browser. Other projects focus on providing Semantic Web access in a small (DynamicView [Gao *et al.*, 2005], mSpace [Shadbolt *et al.*, 2004]) or larger (Haystack [Quan and Karger, 2004], SEAL [Hartmann and Sure, 2004]) domain.
In terms of personalization different adaptive systems [Cheverst *et al.*, 2002; Bra *et al.*, 2002] implement user modeling directly in their systems. Thus the change of application domain requires an adjustment of user modeling (open corpus problem [Brusilovsky, 2001]). [Henze and Nejdl, 2004] proved for the domain of educational learning that user modeling can be separated from the adaptive system.
An overview of privacy issues for distributed user profile usage is given in [Clauß *et al.*, 2002]. A framework for exchanging personal data is introduced in [Berthold and Köhntopp, 2000] which is used in [Koch and Wörndl, 2001] to share personal data between different applications. Our contribution to this related work is to benefit from distributed user modeling strategies and combine them with a centralized user profiling manager in the highly dynamic environment of the Semantic Web, where classic user modeling methods cannot be applied.

5 Conclusion and Further Work

We presented the Personal Reader Framework that offers personalized, user-driven access to the Web Service-based Semantic Web. To enable user modeling in such a highly dynamic environment we presented a centralized user profiling approach. A user's profile can in parts be accessed and eventually modified by the Web Services the user trusts; According access rights for Web Services are maintained by privacy policies in the User Profile Manager, thus centralized and under full control of the user.
At this time, we have not implemented the user profile yet. Our current work focuses on combining different user profiles that were developed for and maintained by single Web Services. Future work will be the integration of more Personalization Services into our Personal Reader Framework to demonstrate the advantages of a shared user profile in the Semantic Web.

[1] http://simile.mit.edu/piggy-bank/
[2] http://simile.mit.edu/longwell/
[3] http://brownsauce.sourceforge.net/

References

[Abel *et al.*, 2005] Fabian Abel, Robert Baumgartner, Adrian Brooks, Christian Enzi, Georg Gottlob, Nicola Henze, Marcus Herzog, Matthias Kriesell, Wolfgang Nejdl, and Kai Tomaschewski. The personal publication reader, semantic web challenge 2005. In *4th International Semantic Web Conference*, nov 2005.

[Abel *et al.*, 2006] F. Abel, I. Brunkhorst, N. Henze, D. Krause, K. Mushtaq, P. Nasirifard, and K. Tomaschewski. Personal reader agent: Personalized access to configurable web services. In *Proceedings of the Fourteenh GI- Workshop on Adaptation and User Modeling in Interactive Systems (ABIS 06)*, Hildesheim, Germany, 2006.

[Bailey *et al.*, 2004] James Bailey, George Papamarkos, Alexandra Poulovassilis, and Peter T. Wood. An event-condition-action language for xml. In Mark Levene and Alexandra Poulovassilis, editors, *Web Dynamics*, pages 223–248. Springer, 2004.

[Baumgartner *et al.*, 2005] Robert Baumgartner, Nicola Henze, and Marcus Herzog. The Personal Publication Reader: Illustrating Web Data Extraction, Personalization and Reasoning for the Semantic Web. In *European Semantic Web Conference ESWC 2005*, Heraklion, Greece, May 29 - June 1 2005.

[Berthold and Köhntopp, 2000] Oliver Berthold and Marit Köhntopp. Identity management based on p3p. In *International Workshop on Design Issues in Anonymity and Unobservability*, 2000.

[Bra *et al.*, 2002] P. De Bra, A. Aerts, D. Smits, and N. Stash. AHA! Version 2.0: More Adaptation Flexibility for Authors. In *Proceedings of the AACE ELearn'2002 conference*, October 2002.

[Brusilovsky, 2001] Peter Brusilovsky. Adaptive Hypermedia. *User Modeling and User-Adapted Interaction*, 11:87–110, 2001.

[Cheverst *et al.*, 2002] Keith Cheverst, Keith Mitchell, and Nigel Davies. The role of adaptive hypermedia in a context-aware tourist guide. *Commun. ACM*, 45(5):47–51, 2002.

[Clauß *et al.*, 2002] Sebastian Clauß, Andreas Pfitzmann, Marit Hansen, and Els Van Herreweghen. Privacy-enhancing identity management. In *The IPTS Report, Special Issue: Identity and Privacy*, pages 8–16, 2002.

[Gao *et al.*, 2005] Z. Gao, Y. Qu, Y. Zhai, and J. Deng. Dynamicview: Distribution, evolution and visualization of research areas in computer science. In *Proceeding of International Semantic Web Conference*, 2005.

[Hartmann and Sure, 2004] Jens Hartmann and York Sure. An infrastructure for scalable, reliable semantic portals. *IEEE Intelligent Systems*, 19(3):58–65, 2004.

[Heckmann, 2005] Dominik Heckmann. *Ubiquitous User Modeling*. PhD thesis, Department of Computer Science, Saarland University, Germany, November 2005.

[Henze and Kriesell, 2004] Nicola Henze and Matthias Kriesell. Personalization Functionality for the Semantic Web: Architectural Outline and First Sample Implementation. In *1st International Workshop on Engineering the Adaptive Web (EAW 2004)*, Eindhoven, The Netherlands, 2004.

[Henze and Nejdl, 2004] Nicola Henze and Wolfgang Nejdl. A Logical Characterization of Adaptive Educational Hypermedia. *New Review of Hypermedia*, 10(1), 2004.

[Koch and Wörndl, 2001] Michael Koch and Wolfgang Wörndl. Community support and identity management. In *Proceedings Eurpean Conference on Computer Supported Cooperative Work (ECSCW 2001)*, 2001.

[Quan and Karger, 2004] D. Quan and D. Karger. How to make a semantic web browser. In *Proceedings of the 13th International Conference on World Wide Web*, pages 255–265, 2004.

[Shadbolt *et al.*, 2004] N. R. Shadbolt, N. Gibbins, H. Glaser, S. Harris, and m. c. schraefel. CS AKTive space or how we stopped worrying and learned to love the semantic web. *IEEE Intelligent Systems*, 19(3), 2004.

Validating Navigation Time Prediction Models for Menu Optimization

Vera Hollink, Maarten van Someren
Faculty of Science, University of Amsterdam
Kruislaan 419, 1098 VA Amsterdam, The Netherlands
{vhollink,maarten}@science.uva.nl

Abstract

Authors of menu optimization methods often use navigation time prediction models without validating whether the model is adequate for the site and its users. We review the assumptions underlying navigation time prediction models and present a method to validate these assumptions offline. Experiments on four web sites show how accurate the various model features describe the behavior of the users. These results can be used to select the best model for a new optimization task. In addition, we find that the existing optimization methods all use suboptimal models. This indicates that our results can significantly contribute to more effective menu optimization.

1 Introduction

Hierarchical navigation menus are a popular medium to allow users of web sites access to the site's contents. These menus consist of hierarchies of categories with the content pages located at the leaf nodes. To reach their target information users navigate top-down through the hierarchy by selecting categories.

The initial design of menus is often far from optimal as designers do not know the goals and strategies of their future users. Moreover, even with a good initial structure navigation can become less efficient when the user population or the contents of the site change over time.

Various authors have attempted to overcome these problems by presenting methods to automatically adapt the structure of a menu towards the site's actual user population, e.g. [Witten and Cleary, 1984; Fisher *et al.*, 1990], or to the behavior of individual users, e.g. [Smyth and Cotter, 2003; Hollink *et al.*, 2005]. These methods address the optimization of menus with a purely navigational function. In these menus the hierarchical structures do not provide information, but are only means to navigate to the content pages on the terminal nodes. Consequently, the optimal menu is the one that minimizes the average time users need to reach their target pages.

All menu optimization techniques involve adaptation of menu structures and evaluation of the adapted structures. The techniques define a set of possible adaptations that can be made to a site's original menu. They choose which adaptations are performed on the basis of an evaluation metric that expresses the quality of the adapted structures.

The evaluation of adapted menu structures is always done offline as online evaluation of all possible adapted menus is not feasible. In an online evaluation all menus need to be placed on the site for some time until a sufficiently large number of users have used the menu. This would not only take an unacceptable amount of time, but also would mean that the users face a continually changing menu that is often even worse than the initial menu. In an offline evaluation the efficiency of the adapted structures is not measured directly, but predicted on the basis of a model of the user population.

If we compare the models of existing menu optimization methods, we find large differences in the underlying assumptions about the users' targets and navigation strategies. For example, some models assume that users read all available menu items before making a choice, while others assume that users stop reading as soon as an acceptable item is encountered. The assumptions behind the models are seldom mentioned explicitly and even more seldom validated. We feel this is a great deficiency as the used model specifies the direction of the optimization and thus determines for a large part the success of the optimization.

In this work we review the assumptions behind models that predict average navigation time. The various models are validated offline on real log data of four web sites with hierarchical menus. The contributions of this paper are threefold. 1) We make the assumptions behind navigation time models explicit, so that for a particular application one can select a model whose assumptions hold. 2) If in the experiments certain assumptions appear to be inherently better than others, this reduces the scope of the models that need to be considered when optimizing menus of new sites. 3) We provide a method to find for a new site the best fitting model among the potentially optimal models.

Section 2 discusses the models underlying various optimization methods. In section 3 we present a framework to compare the available models. Section 4 explains the procedure that we use to validate the models and in section 5 we apply the models to log data of four web sites. The last section contains conclusions and discusses the results.

2 Twelve navigation time prediction models

We examined the models for predicting expected navigation time of twelve menu optimization methods. Below, we briefly describe the context of the methods and their main properties.

One of the first menu optimization methods was developed by Witten and Cleary [1984]. They optimized the hierarchical index of a digital phonebook using the access frequencies of the phonenumbers. A limited time prediction model was used that assumes that the choice lists (the lists of categories located under the same items in the hierarchy) have equal and non-adaptable numbers of items.

Lee and McGregor [1985] explicitly sought to quantify the relation between menu structure and navigation time. They assumed users always searched for only one page and all pages had equal probability of being sought. Later Landauer and Nachbar [1985] extended their model to menus where the choice lists were ordered alphabetically. Paap [1986] added the possibility that the choice lists themselves were categorized. Fisher *et al.* [1990] improved the Lee and McGregor model by adding frequency based page probabilities. Moreover, they invented an algorithm to optimize menus on the basis of the improved model. A limitation of this algorithm is that it can only find structures that can be formed by removing intermediate nodes from the original hierarchy.

Bernard [2002] presented another model for predicting navigation time: the Hypertext Accessibility Measure (H_{HAI}). Like the Lee and McGregor model, the H_{HAI} measure predicts the expected navigation time solely on the basis of the menu structure.

The ClickSmart system [Smyth and Cotter, 2003] adapts WAP menus to the behavior of individual users. The time prediction model that is used is called the click-distance. This model is in fact an instantiation of the model introduced by Fisher *et al.*. To circumvent the problem of creating labels for new menu items, the optimization algorithm can only make hierarchies flatter and not deeper.

In [Hollink *et al.*, 2005] we presented a system that adapts web menus to individual users. We used a model that was similar to Fisher's model but, unlike Fisher's model, our model assumes that users sometimes make navigation mistakes. The applicability of the algorithm is restricted to situations in which the pages are labeled with keywords that can function as labels for the menu items.

The MESA model [Miller and Remington, 2004] is to our knowledge the only quantitative model that links the probability of making navigation mistakes to the quality of the items' labels. The connection between label quality and mistake probability seems natural, but the practical applicability of this model is limited as for all labels quality assessments need to be provided by experts.

Allan and Bolivar [2003] provide three models to assess the quality of a document hierarchy created through hierarchical clustering: the minimal travel cost, the expected travel cost and the expected accumulated travel cost. The models are not designed for predicting navigation time in web menus, but as they predict the amount of time users need to locate documents in a hierarchy, they can be used for this purpose without modification.

3 Time prediction framework

In this section we provide a framework that allows us to systematically compare the available navigation time prediction models. The core of the framework is formed by the dependencies shown in Figure 1. The time users need to navigate through a menu depends on the paths that they follow through the menu and the strategy they use to follow these paths. The followed paths in turn are a consequence of the users' targets, the nature of the menu and the strategy the users use to search the menu for their targets.

All time prediction models that we encountered follow this general schema. The differences between the models lie in their assumptions about the factors that determine navigation time. Below we review the target set features and the strategy features that are used in the optimization methods introduced in the previous section. In addition,

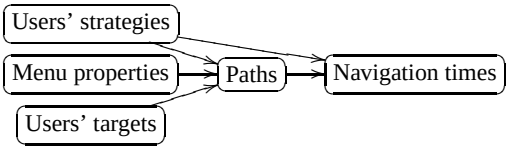


Figure 1: Causal dependencies between a menu structure, the users' targets, the users' strategies, the paths they follow through the menu and the time they spend navigating to their target pages.

we discuss the assumptions that the features represent and the circumstances under which these assumptions are justified. Table 1 lists the features, and positions the twelve navigation time prediction models in the framework.

Due to space limitations, the features that concern the characteristics of the menus are left out of the discussion. Some models only apply to menus with certain characteristics, for instance menus with equal numbers of items in all choice lists. However, these features can be observed directly from the menus, so that validating whether these features apply to the situation is trivial.

Most models in Table 1 actually represent *classes of* models rather than individual models. The models in these classes share the same features, but some of the features have parameters that need to be determined anew for each site. For example, the fact that a model uses page probabilities is a feature and the parameters of this feature are the probabilities of the pages of a particular site. In this work we evaluate model classes and not individual models. The word 'model' will be used to refer to both model classes and models.

3.1 Users' targets

A user comes to a site to fulfill certain information needs. The pages that together fulfill these needs we call the user's *target pages* or his *target set*. We distinguish three features of the users' target sets (see Table 1). First, most models assume that any set of pages can be a user's target set. Only the travel cost models [Allan and Bolivar, 2003] make use of predefined topics that form the possible target sets. According to these models a user is interested in exactly one topic and searches for all pages on this topic. The travel cost models are developed for assessing document hierarchies. In this setting the topics form the gold standard for the clusters at the lowest level of the hierarchy. The second feature is the size of the target sets. Most models assume each user searches for exactly one target. The travel cost models allow for the possibility that a user has multiple targets, namely all pages belonging to one topic.

The third feature is the probability distribution over the target sets. The models that are explicitly developed to predict navigation time all assume that the target sets have equal probability of being sought (uniform). They compute expected navigation time as the unweighted average of the times needed to reach each of the targets. All models used in optimization algorithms assume that the probabilities are proportional to the frequency of the sets in the log files. This extension has a clear value for menu optimization, as it causes algorithms to place more frequently accessed pages at a more prominent position in the hierarchy.

3.2 Navigation strategies

Table 1 contains seven features that concern the users' navigation strategies, five of which influence the prediction of the users' paths. The first feature, the users' search strategy, involves the order in which users open hierarchy nodes. Most models assume that users use a greedy depth

Table 1: Properties of navigation time prediction models

Model	Features of targets			Features of users' strategies						
	Target set	Target set size	Target set probabilities	Users' search strategy	Multiple target search	Mistake probability	Users' stop condition	Users' choice strategy	Node choice function	Node opening function
Witten and Cleary [1984]	all	one	frequency	greedy	-	0	all targets	-	0	linear
Lee and McGregor [1985]	all	one	uniform	greedy	-	0	all targets	read all/ until found	linear	linear
Landauer and Nachbar [1985]	all	one	uniform	greedy	-	0	all targets	read all	logarithmic	linear
Paap and Roske-Hofstrand [1986]	all	one	uniform	greedy	-	0	all targets	read until found	logarithmic	linear
Fisher *et al.* [1990]	all	one	frequency	greedy	-	0	all targets	read until found	linear	linear
Hypertext accessibility measure [Bernard, 2002]	all	one	uniform	greedy	-	0	all targets	read all	logarithmic	logarithmic
Click-distance [Smyth and Cotter, 2003]	all	one	frequency	greedy	-	0	all targets	read until found	linear	linear
Hollink *et al.* [2005]	all	multiple	frequency	greedy	separate	fixed	all targets	-	0	linear
MESA model [Miller and Remington, 2004]	all	one	uniform	greedy	-	label quality	all targets	read until found	linear	linear
Minimal travel cost [Allan and Bolivar, 2003]	predefined	multiple	uniform	greedy	continual	0	best category	read all	linear	linear
Expected travel cost [Allan and Bolivar, 2003]	predefined	multiple	uniform	exhaustive	continual	0	all targets	-	0	linear
Expected accumulated travel cost [Allan and Bolivar, 2003]	predefined	multiple	uniform	greedy	continual	0	all targets	-	0	linear

first strategy. According to these models, users perform a depth first search to their target pages. The users base their choices on the items' labels and only open items that lead to targets. For users with a single target page this means that they take the shortest path. The expected travel cost model assumes a different strategy. According to this model users perform an exhaustive depth-first search visiting all nodes until they happen to hit their targets. This means that in the worst case a user traverses the whole tree.

Users following the greedy strategy only open items that lead to targets. The second strategy feature, the users' choice strategy, concerns the way the users select these items from the choice lists. A user can read all labels and then select the best node or start reading at the top of the list and open an item as soon as an acceptable item is read.

The third feature concerns the behavior of users with more than one target. The simpler models assume that these users search for each target separately, in other words, that they go back to the starting point after a target is found. More complex models include a continual search pattern which means that users surf from the starting point to the first target and from this target to the second target, etc.

The fourth navigation strategy feature is the probability that users using a greedy strategy make navigation mistakes. Here making a mistake means selecting an item that does not lead to a target page. Most models simply assume users never make mistakes or make random selections with a small but fixed probability. The MESA model uses the quality of the items' labels to determine the probability of a user selecting an item erroneously.

The fifth element of the users' strategy is their stop condition. The minimal travel cost assumes that users stop navigating once they have reached the menu item under which most target pages are located. All other models assume users keep searching until all target pages are found.

The final two strategy features are the node opening function and the node choice function. They specify the relation between the path followed through the site and the navigation time. Navigation time is determined by two properties of the path: the number of menu items a user has opened ($|Path|$) and for each navigation step the number of items in the choice list that the user has read ($\#choices$):

$$Time = \beta.f(|Path|) + \Sigma_{\{n \in Path\}} \alpha.g(\#choices(n))$$

Here f is the node openings function and g is the node choice function. α and β are parameters that respresent respectively the time users need to read an item from the choice list and the time users need to open a menu item. The value of $\#choices(n)$ depends on the choice strategy of the users. As mentioned before, one can assume that users read all items of a choice list or that they stop reading when an acceptable item is found. For both functions f and g three variants appear in literature: a linear function, a logarithmic function and a null function, meaning that the factor has no influence. A linear relation between navigation time and the number of item openings means that opening an item takes equal time at each level of the hierarchy. A linear choice function implies that users go top-down through the choice lists and need equal time to read each item. A logarithmic choice function is justified when the list of items is ordered and people do not need to read every item to find the one they need. Finding a known item in an alphabetic list of n items can be done by making a series of binary splits, which results in reading only $log_2(n)$ items. A logarithmic opening function, which is

used in the H_{HAI} model, can not be justified in this way, as one always has to open all items on the path.

4 Validating time models

The many differences between the twelve models make clear that choosing a navigation time prediction model for a menu optimization task is a non-trivial problem. For the menu features one can simply check whether they apply to the menu at hand, but the users' strategy and targets are not so easily observable. Some of the feature values are equivalent variants such as logarithmic and linear choice functions. Others are merely extensions of each other. For instance, a model with uniform target probabilities is in fact a simplified version of a model with frequency based probabilities. To find the best fitting model one needs to determine which of the variants model the situation best and whether the extensions lead to significant improvements.

Below we systematically test all valid combinations of features (including combinations that do not appear in the models in Table 1) to determine the relative importance of the various features. We create instantiations of the models for four web sites (i.e. we set the parameters of the model features). We apply the instantiated models to the sites' menus and log files and measure how well the models predict the users' paths and navigation times. These experiments lead to recommendations for using the more complex or the simpler features. For the features for which the optimal choices differ per site we provide a method to determine for a given site which choices are optimal.

The evaluation consists of three parts: first we validate the assumptions about how users with a given target set choose a path, then we validate the ones that determine navigation time given a path and finally we validate the assumptions about the users' targets. The following sections describe the procedures for validating the assumptions. In section 5 these procedures are applied to four web sites.

4.1 Data preprocessing

From the log files we restore the sessions of individual users. All requests coming from the same IP address and the same browser are contributed to one user. When a user is inactive for more than 30 minutes, a new session is started. The sessions include both target and non-target pages. We determine the most likely targets on the basis of the time the user spent on the pages. All pages with a reading time longer than or equal to the median reading time of the hierarchy's end pages are marked as target pages. The other pages form the paths to the targets. The rationale behind the use of the median reading time is that target pages are pages to which a user pays more than usual attention.

The median reading time is a crude criterion for selecting targets. However, in our experiments we found that choosing higher or lower time thresholds changed the absolute scores of the various models, but not their relative performance. Nevertheless, it is questionable whether reading time is at all a good criterion to select targets. It is plausible that on average users spent more time on target pages than non-target pages, but clearly this does not hold for every individual page view. Without prior knowledge reading time is the only source of information. However, on many sites characteristics of the pages can be used to make a more informed estimation of the users' targets, for example using the page characterizations used in the WUM method [Spiliopoulou and Pohle, 2001].

4.2 User strategies for predicting paths

Table 1 contains five features that influence the paths that users with a given target set follow through a menu. We determine the impact of the users' search strategy, the users' choice strategy, the search for multiple targets and the users' mistake probability. The stop condition is not used as we have no means to determine whether users would have liked to find more pages besides the ones they visited. We only test fixed mistake probabilities, because label quality assessments are generally not available.

Each combination of features is combined into a partial model that predicts paths. The partial models are evaluated by comparing the predicted paths to the paths that the users actually followed on the site. For each target set in the log files the models predict a path along all targets. In the end we count how many of the predicted page transitions actually occurred in the users' sessions. The models are compared on precision and recall. Here precision is the number of correctly predicted transitions divided by the total number of predicted transitions. Recall is the number of correctly predicted transitions divided by the number of transitions in the users' sessions. We focus on the page transitions rather than the visited pages themselves, because the transitions determine the navigation time, as we will see below.

4.3 User strategies for predicting times

We evaluate all features that determine the predicted navigation times: the users' choice strategy, the node opening function and the node choice function. Partial models that predict navigation times are formed for all combinations of features. For each path to a target page in the log files we compute the time it took the user to traverse the path. In addition, we count the number of menu items the user opened along the way and the number of choices he had in each step. To these data the time prediction models are fitted in such a way that the mean of squared errors is minimized. This results in optimal parameter settings for the models (values for α and β, see section 3.2).

A 5-fold cross-validation is used to evaluate how well the models predict navigation times of future users. The models are fitted to the training sets and evaluated on the test sets. As evaluation measure we use the R-square measure, which expresses the proportion of the variance in the users' navigation times that is explained by a model.

4.4 Predicting target sets

We validate models with various values for the target set size and the target set probabilities. All models assume that all target sets are possible. Models with predefined topics are not considered, as in general it is not possible to find a division in topics that applies to all visitors.

Again we split the log data in test and training sessions. The training data is used to compute the target set probabilities. During training each target set model produces a collection of target sets that simulates the targets of the actual users. The simplest model is the single uniform model. It assumes users search for single targets and all targets have equal probability. Its target set collection is a list of all pages of the site. The single frequency model also assumes users search for single targets, but now the target probabilities are based on the number of times each page occurs as a target in the training sessions. The multiple frequency model consists of target sets with more than one page. Its target set collection is a list of all target sets occurring in the

Table 2: Properties of the four sites that are used for evaluation.

Site	Log Period	Number of sessions	Number of menu items	Maximal menu depth
SG	9 months	51,567	92	3
RN	9 months	23,995	100	6
GH	1 month	22,788	59	6
HI	4 days	2,062	288	4

training set. The collection of the multiple uniform model would comprise all possible target sets (the power set of the site's pages), but the computation of this collection is not tractable for sites with more than a few pages.

The purpose of the test sets is to evaluate how well the target set collections of the three models reflect the targets of the actual users of the site. For each target set in each collection we estimate the time users need to locate the target pages using the path and time models that scored best in the previous evaluations. The expected navigation time of a collection is the weighted average time over all targets in the collection. The expected navigation times are compared to the average time that users from the test set really needed to locate a target. As evaluation measure we use the relative error: the difference between the expected navigation time and the real average navigation time as percentage of the real average navigation time.

5 Experiments

We applied the method described in the previous section to log data of four Dutch web sites. The sites are from different domains and their menus vary in size and structure. The SeniorGezond site (SG)[1] gives information about the prevention of falling accidents. It provides many different navigation means one of which is a hierarchical navigation menu. The Reumanet site (RN)[2] contains information about rheumatism. GHadvies (GH)[3] is a site about layoff compensation. HoutInfo (HI)[4] contains pages about the properties and applications of various kinds of wood. Features of the sites' log files and menus are given in Table 2.

The partial models for path prediction were applied to the four sites. The results of the experiments are given in Table 3. There are only two models with exhaustive strategies, because with this strategy there is no difference between the two choice strategies. The exhaustive models predicted extremely long paths which resulted in moderate recall, but very low precision. The greedy models resemble the true strategy of the users much better: 40-55% of the predicted transitions were actually followed. No large differences were found between the two choice strategies. Possibly, this is because both strategies were used by large user groups. In all cases the continual target search models worked much better than the separate search models.

In a second set of experiments we added fixed mistake probabilities to the greedy models with multiple targets. Including navigation mistakes did not improve the models: both precision and recall decreased almost linearly with increasing mistake probability.

In conclusion, when optimizing a menu, the best choice is a greedy model without navigation mistakes. Either one of the choice strategies can be used. In addition, the model should take into account that users with multiple targets do not start over each time a target is found.

[1] http://www.seniorgezond.nl/

[2] http://www.reumanet.nl/

[3] http://www.goudenhanddrukspecialist.nl/

[4] http://www.houtinfo.nl/

Table 3: Precision and recall of the path prediction models. E is exhaustive strategy, G is greedy strategy, C is continual target search, S is separate target search, A is read all choices, and U is read until good item found.

Data set		Path model ES	EC	GSA	GCA	GSU	GCU
SG	precision	0.010	0.016	0.240	0.442	0.240	**0.443**
	recall	0.234	0.196	0.301	0.308	0.301	**0.309**
RN	precision	0.012	0.028	0.184	0.414	0.184	**0.417**
	recall	0.219	0.203	0.284	0.316	0.284	**0.318**
GH	precision	0.022	0.062	0.196	0.530	0.196	**0.535**
	recall	0.338	0.298	0.308	0.359	0.308	**0.363**
HI	precision	0.007	0.015	0.335	**0.500**	0.335	0.499
	recall	**0.517**	0.376	0.407	0.343	0.407	0.342

Table 4: Average R-square of the time prediction models. L is logarithmic, S is linear (straight), 0 is zero, and U is read until good item found. The first character is the node opening function and the second character the choice function.

Data set	Time model 00	S0	SSU	SLU	L0	LSU	LLU	H_{HAI}
SG	-0.01	0.88	**0.88**	0.88	0.73	0.84	0.85	0.74
RN	-0.00	0.67	0.68	0.68	0.69	0.73	0.72	**0.74**
GH	-0.00	0.78	0.79	**0.79**	0.64	0.75	0.74	0.72
HI	-0.00	0.84	0.86	**0.86**	0.62	0.75	0.76	0.80

The results of the experiments with time prediction models are given in Table 4. All figures are averages over the 5 folds. Due to space limitations the table only shows the models with *read until found* choice strategies. The results of the *read all* choice strategies are very similar as was the case in the path prediction experiments. The results of the H_{HAI} model [Bernard, 2002] are shown separately. This model is basically a double logarithmic (LLA) model, but with some small modifications.

The results of the time experiments are less clear than the results of the path experiments. Nevertheless, some observations can be made. Models that use both the number of node openings and the number of choices perform better than models that disregard the number of choices (00, L0 and S0) or the number of node openings (not shown). Apparently, both elements influence navigation time. As expected, on three of the four data sets linear node opening functions gave better results than logarithmic opening functions. Only on the Reumanet data set the logarithmic opening functions worked best, but on this data set all models performed low. Apparently, navigation times were more noisy on the Reumanet site. A possible explanation is that the site is visited frequently by people with rheumatism for who clicking links is more difficult.

The difference in performance between models with logarithmic and linear choice functions is small. We expected to find a preference for linear choice functions, because the sites have unordered choice lists. Apparently, visitors manage to select items without reading all preceding items. This can be a learning effect: when a user has opened an item before, he remembers where the item is located.

The values of the parameters α and β depend on the complexity of the labels and the experience of the users and differ per site. For the LLU model we found that β should be between 2 and 5 times as large as α. This coincides with the values used in the MESA model [Miller and Remington, 2004] $\alpha = 0.25$ and $\beta = 0.5$. In the click-distance model [Smyth and Cotter, 2003] selecting and clicking links takes equal time, but these values are meant for WAP users who navigate via mobile phones.

For a new menu optimization task, we recommend to use a linear node opening function, because this function tends

Table 5: Relative error of the target set prediction models in combination with the GCU path model and the SLU time model.

	Target set model		
	Single	Single	Multiple
Data set	Uniform	Frequency	Frequency
SG	2.16	1.14	**0.15**
RN	2.46	1.11	**0.29**
GH	3.70	2.13	**0.04**
HI	3.10	1.69	**0.10**

to outperform other models and has better theoretical foundations. The best node choice function is strongly site dependent and should be determined anew for each site. This can be done offline in the same way we performed the time model experiments. At the same time these experiments will yield the optimal parameter settings.

In the target set evaluations we used the GCU path model and the SLU time model. Table 5 shows the error of the prediction of the expected navigation time when various target set models are used. The use of target set frequencies considerably improved the prediction. Furthermore, for all sites the model using target sets with multiple targets outperformed the models with singleton target sets. This confirms our earlier conclusion that it is important to model the behavior of users with more than one target.

In summary, in our experiments we found clear evidence that greedy continual search path models and multiple target frequency target set models are the best choices. If we compare these to the models in Table 1 we see that none of the optimization methods uses the optimal model class. This suggests that menu optimization can be improved by using the optimal average navigation time model.

6 Conclusion and discussion

In this work we gave an overview of the assumptions that are explicitly or implicitly used in navigation time prediction models. We presented a method to validate the assumptions offline using a site's log files. The method was applied to the menus of four web sites with hierarchical menus. In our experiments several model features appeared to be inherently better than others. These findings limit the set of models that need to be considered when the optimal model is sought for a new menu optimization task.

For the optimization of a menu in a new domain the path and target set models that performed best in our experiments can be used directly. We found that the optimal features of the path and target set models are the same for all sites. The best choice for the time prediction features is site dependent. Therefore, for a new domain the best time features needs to be determined from the log data. This can be accomplished with the method described in this work.

With the presented methods we can fit a limited set of models to a site's log file and make a well-funded choice for a navigation time prediction model. Using the right model is essential for menu optimization, because an accurate model of the users' behavior makes sure that one optimizes towards the menu with the shortest average navigation time. Comparison of our findings with the models used in menu optimization methods shows that all methods use suboptimal models. Thus, selecting the right models with the presented procedures can make menu optimization much more effective.

To obtain generally valid result we used web sites from different domains and with different characteristics. Nevertheless, it is possible that in other domains with yet other characteristics other models become optimal. Moreover, more relevant features may exists that are not present in any of the examined optimization methods.

Another limitation of the procedures described in this work is that they evaluate the models only on log data produced by users who used the sites' original menus, while the purpose of the models is to predict the average navigation times of menu structures after they have been adapted. To see how well the models generalize to new structures one needs log data created with different menus for the same site. Therefore, the next step of our research will be to incorporate the best performing model in an optimization tool. The tool will be applied to menus of real web sites and the resulting menus will be placed online. Comparison of the users' navigation times before and after the optimization allows us to evaluate the accuracy of the time predictions.

References

[Allan and Bolivar, 2003] J. Allan, A. Feng, and A. Bolivar. Flexible intrinsic evaluation of hierarchical clustering for TDT. *Proc. of the CIKM 2003*, pp. 263–270, New Orleans, USA, 2003.

[Bernard, 2002] M. L. Bernard. Examining a metric for predicting the accessibility of information within hypertext structures. *PhD. Thesis Wichita State University*, Wichita, USA, 2002.

[Fisher *et al.*, 1990] D. L. Fisher, E. J. Yungkurth, and S. M. Moss. Optimal menu hierarchy design: syntax and semantics. *Human Factors*, 32(6):665–683, 1990.

[Hollink *et al.*, 2005] V. Hollink, M. van Someren, S. ten Hagen, and B. Wielinga. Recommending informative links. *Proc. of the IJCAI-05 Workshop on Intelligent Techniques for Web Personalization*, pp. 65–72, Edinburgh, UK, 2005.

[Landauer and Nachbar, 1985] T. K. Landauer and D. W. Nachbar. Selection from alphabetic and numeric menu trees using a touch screen: depth, breadth and width. *Proc. of the SIGCHI conf. on Human Factors in Computing Systems*, pp. 73–78, San Francisco, USA, 1985.

[Lee and MacGregor, 1985] E. Lee and J. MacGregor. Minimizing user search time in menu retrieval systems. *Human Factors*, 27(2):157–162, 1985.

[Miller and Remington, 2004] G. S. Miller and R. W. Remington. Modeling information navigation: implications for information architecture. *Human-Computer Interaction*, 19:225–271, 2004.

[Paap and Roske-Hofstrand, 1986] K. R. Paap and R. J. Roske-Hofstrand. The optimal number of menu options per panel. *Human Factors*, 28(4):377–385, 1986.

[Smyth and Cotter, 2003] B. Smyth and P. Cotter. Intelligent navigation for mobile internet portals. *Proc. of the IJCAI'03 Workshop on AI Moves to IA: Workshop on Artificial Intelligence, Information Access, and Mobile Computing*, Acapulco, Mexico, 2003.

[Spiliopoulou and Pohle, 2001] M. Spiliopoulou and C. Pohle. Data mining for measuring and improving the success of web sites. *Special issue on applications of data mining to electronic commerce, Journal of Data Mining and Knowledge Discovery, 5(1-2):85–114, 2001*.

[Witten and Cleary, 1984] I. H. Witten and J. G. Cleary. On frequency-based menu-splitting algorithms. *Int. Journal of Man-Machine Studies*, 21:135–148, 1984.

Personalisation in German Smart Sensor Web

Sandro Leuchter, Dirk Mühlenberg & Rainer Schönbein
Fraunhofer Institute for Information and Data Processing (IITB)
Fraunhoferstr. 1, D-76131 Karlsruhe, Germany
firstname.lastname@iitb.fraunhofer.de

Abstract

German Smart Sensor Web (GSSW) is an experimental system for the German Federal Armed Forces. Its purpose is to provide a secure integration infrastructure for networked sensors. GSSW has a middleware based on ontologies and software agent technology. It uses a semantic representation of sensor data and other information in the area of intelligence, surveillance and reconnaissance (ISR) to feed "smart" symbolic AI based assistance functions. Interface agents also use the knowledge representation to personalize different aspects of the user interface. In this contribution the current state of GSSW, its software architecture and the personalization features of the user interface layer are presented.

1 Introduction

Intelligence, surveillance, and reconnaissance (ISR) are important military processes. The aim is to gain battlefield advantage through information superiority. The availability of secure information and communication networks results in the request for the collection, merging, and timely dissemination of sensor data and derived intelligence.

Different technological and organizational approaches are possible to achieve such an integration of information sources. GSSW is an attempt to gain empirical evidence for comparing the feasibility and effectiveness of these implementation strategies. GSSW consists of an experimental secure network and a middleware for integrating different kinds of information sources in the area of ISR.

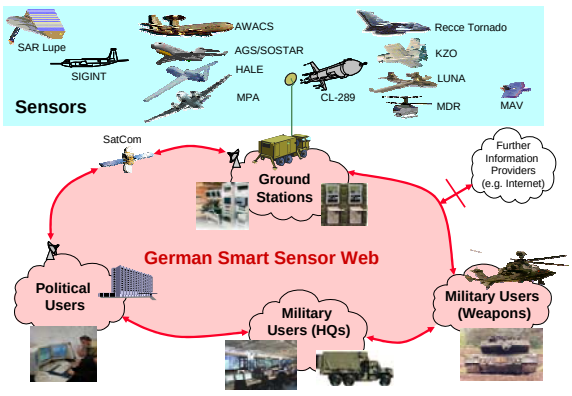


Fig 1: Concept of Operations

The project goal for GSSW is to enable each user to access all reconnaissance information and services relevant for him or her rapidly with the help of support systems. The necessary steps are to identify, access, and retrieve the needed information and to categorize, summarize, filter, and evaluate available information according to the users' individual tasks.

The rough concept of operations (see Fig 1) is that sensors like unmanned aerial vehicles or satellites provide data through their ground stations. Ground stations have databases that store the ISR relevant information to be integrated into the information space of GSSW. Users include political and military users. Users and information source nodes are connected via a secured network. Some nodes in the network do not only provide information sources but also ISR specific algorithmic services either provided by computer systems or human operators.

2 Network

Several locations are connected through the GSSW network. It is implemented as a virtual private network based on the secure internet architecture (SINA [BSI, n.y.]) via the public internet. Currently there are four locations connected in the GSSW. Every location provides information sources, services by users and computer systems. Two locations provide additionally a middleware with user interfaces and functionality for access to ISR services and information. The user interfaces are implemented as web applications and can be accessed from every node with a web browser.

One of the two alternative systems has been implemented by Fraunhofer IITB. The software architecture and its means for personalization are presented in this contribution.

The current implementation of GSSW uses different information sources. They are structural equivalent to operational systems but do not store operational data or are deployed as part of operational systems. Examples for content provided by the information sources are aerial images, intelligence reports, and air reconnaissance tasks.

Examples for services are automated target recognition software for the automatic detection of runways, and services for the conversion of image formats.

3 Software Architecture

The software architecture of the Fraunhofer IITB node of GSSW is based on a semantic representation of ISR information. This representation uses a specific GSSW ontology on ISR concepts. The representation is used to

structure the communicative speech acts of software agents that encapsulate the middleware, provision of services and data access.

3.1 ISR Ontology

Ontologies are linguistic representations of concepts for the knowledge domain. They consist of a description of classes in the domain, relationships between classes in terms of axioms or rules, and a lexicon with syntactic representations of concepts. It is thus a terminology description of domain specific information. Technical it establishes a common vocabulary for agent interaction.

The ISR ontology provides a semantic representation and support infrastructure for high level elements such as ISR objects, ISR processes, and workflows.

3.2 Software Agents

Software agents are a design pattern for cooperative distributed information systems. The GSSW node of Fraunhofer IITB has adopted this design pattern as the basis of the software architecture. It is based on the Java Agent Development Framework (JADE [Bellifemine et al., 2003]). JADE is a development and runtime infrastructure for agent based software systems. It provides good interoperability features through the support for FIPA compliant communications protocols and cooperation language [FIPA, n.y.].

The agents in GSSW are not only deployed as a distributed software infrastructure for cooperative information systems but do also support users in performing demanding tasks in information management.

Features of software agents can be: reactivity, autonomy, cooperation, communication on problem domain level, continuity, deliberative capabilities, adaptivity, mobility, and personality. In a specific implementation only a set out of these features will apply. In GSSW all features except for mobility are met. GSSW-agents are additionally capable of protecting local system resources from unauthorized access.

In GSSW there are agents for accessing information sources and providing services. There are also instances of the interface agent that handle communication with the users and build up their individual user interfaces. The interface agents communicate with the broker agent to achieve the users' information requests. The broker agent knows all resource and service agents and tries to infer a plan to fulfil the interface agent's request.

Fig 2 shows the principal interaction between these agent types. User 1 needs information about airfields in country y. His or her interface agent (1) transforms this to a specific request for the broker agent. The broker agent gets this request via the FIPA ACL using concepts and representations from the ISR ontology. It uses the ontology to make a plan, which agents need to be created or asked to achieve the information request. It builds up an appropriate chain of information resource accesses and services to transform the information. The broker agent knows about all resource and service agents because they have advertised their capabilities according to predefined concepts from the ISR ontology. In the example of Fig 2 there are two agents accessing four different information sources to fulfil the information request of user 1. The broker agent translates the request of interface agent 1 to the appropriate concepts of image agent and report agent. Other users have different instances of the interface agent.

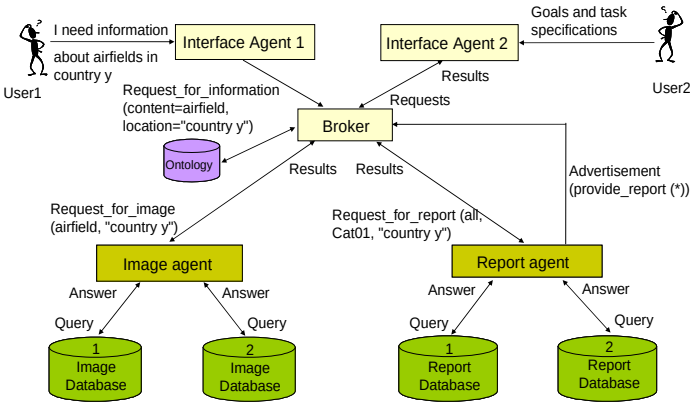


Fig 2: Agent based software architecture

4 User Interface

The user interface of GSSW is targeted at information access and information management in large collections of ISR data. Thus the specific demands of information management in ISR are to be met:

- access to different object types: experts, tasks, reports, maps, images,
- with different methods: push, pull, post before process, subscriptions to new information
- support through functions: zoom, pan, overlay, annotations, mail, chat
- retrieval of ISR information with spatial, temporal, spectral, object specific and free text characteristics
- filtering and selection of information on relevant ISR objects according to user, role, and task

Fig 3 shows a screenshot of the user interface of GSSW. In the centre is a dynamically rendered map. Relevant ISR objects with spatial features are presented as pin points in the map (e.g. reports). Aerial images are shown georeferenced with their foot prints. On the right bottom is an area with buttons for activating tools associated with the map. On the left side is a navigation menu for activating functions.

On the centre bottom is a time scale. Most ISR objects have a timestamp besides their spatial features (e.g. the time when an aerial image was taken or the date of the source for a report). The time scale shows a time-oriented view onto ISR objects. It can be used as a filter to select which information should be displayed spatially in the map.

GSSW features a set of smart support systems that are based on the semantic representation of ISR information. Examples are

- ontology based full text search engine,
- rule-based information fusion,
- automatic annotation in images according to reports.

5 Personalization

Personalization in GSSW does not only apply to the layout and features of the user interface but affects also the internal functioning of the software architecture. It is based on user profiles.

Fig 3: Screenshot of main GSSW user interface

5.1 Profiles

The user profile consists of information about role, tasks, environment, and preferences. Preferences about language, areas of interest/responsibility (spatial as well as temporal) for information access can be modified by the user. Role and tasks affect information access authorization and can thus be modified only administratively.

Users can specify their own preference of different features of services that affect the choice of equivalent available services. Three measures can be given for every available service group: quality, time needed and price.

The quality of individual services is rated by their providers and also by the users. The actual quality is computed as the mean value of the service profile resulting from all individual ratings.

5.2 Adaptive User Interface

Interface agents generate a web-based user interface individually for their respective user. Since the role and tasks of a user specify the needed functionality this information is taken to generate the appropriate navigational aids and tool palettes in the user interface. The preferences also define the area of interest presented in the user interface (map and time scale).

5.3 Adaptive Service brokering

The software architecture of GSSW is based on software agents. Each agent provides services to all other agents. On the basis of a user's interaction an interface agent informs the broker agent of a new goal that has to be achieved. The goal can be to resolve a specific problem or to fulfil a complex information request. The broker agent has information about all capabilities of all other agents. It uses the ontological representation of the capabilities and matches it with the new request. The result is a plan in which sequence agents have to combine their services.

There can be alternative agents (possibly at different locations in the GSSW network) that provide similar services. Since the cost-benefit ratio of different service providers can differ an algorithm suggests the optimal choice to the broker agent. The algorithm takes the users preferences on relevant service features for the appropriate service group into account. The broker agent thus generates a user specific sequence of service providing agents to achieve the goal of the interface agent.

5.4 Information management

Access to information in networked information systems is possible in different ways: Users can push information or meta-information to a public pool and users can pull relevant information from the pool through a search engine or browsing. GSSW allows for an additional personalized way to access the pool. Users can create subscriptions that use the ISR ontology to describe what information in the public pool is of interest to them. A new agent is generated for every subscription that acts as a watch dog and informs its user either asynchronous via SMS or synchronous through a system internal mail box when new matching intelligence products arrive in the pool.

6 Conclusions

GSSW is an experimental system to test information management approaches for handling military intelligence. It has user profile based personalization features that affect the presentation of the user interface and agent interaction.

The latter is a core capability for future service oriented architectures (SOA). Current implementations of SOA infrastructures rely on manually coded static service composition to implement complex business processes. A more flexible and user centred combination of services is needed. The demonstrator GSSW shows a new approach to resolve a complex service request dynamically based on user and service profiles as well as on cost-benefit models.

The use of ontologies and semantic web services is a prerequisite for such future adaptive service brokering and user model based combination of services.

Acknowledgments

The GSSW system has been developed in cooperation with EADS under a research contract of the German Federal Office of Defence Technology and Procurement (BWB). The authors would like to thank the German Federal Ministry of Defence and BWB for the support and Ernst Josef Blum, Stefan Buhl, Dr. Ralf-Peter Eule, Wilmuth Müller, Dr. Detlef Pade, Frank Reinert, and Gottfried Seemann for contributions to GSSW.

References

[Bellifemine *et al.*, 2003] F. Bellifemine, G. Caire, A. Poggi, and G. Rimassa. *JADE - A White Paper, Sept. 2003*, Online document accessible under http://jade.tilab.com/papers/2003/WhitePaperJADEEXP.pdf (last access: July 14, 2006).

[BSI, n.y.] Bundesamt für Sicherheit in der Informationstechnologie [German Federal Agency for Security in IT]. *Sichere Inter-Netzwerk Architektur (SINA) [secure internet architecture (SINA)]*. Online document accessible under http://www.bsi.de/fachthem/sina/index.htm (last access: July 14, 2006).

[FIPA, n.y.] The Foundation for Intelligent Physical Agents. *FIPA specifications*. Online document accessible under http://www.fipa.org/specifications/index.html (last access: July 14, 2006).

Prospector: An adaptive front-end to the Google search engine

Christian Schwendtner, Florian König, Alexandros Paramythis
Johannes Kepler University
Institute for Information Processing and Microprocessor Technology (FIM)
Altenbergerstraße 69, A-4040 Linz, Austria
{schwendtner, koenig, alpar}@fim.uni-linz.ac.at

Abstract

This paper presents Prospector a front-end to the Google search engine which, using individual and group models of users' interests, re-ranks search results to better suit the user's inferred needs. The paper outlines the motivation behind the development of the system, describes its adaptive components, and discusses the lessons learned thus far, as well as the work planned for the future.

1 Introduction

1.1 Background and motivation

The motivation underlying the Prospector's development has been to investigate the extent to which a simple yet effective adaptive meta-search layer can be applied as a front-end to a popular search engine, for personalizing search results. The idea of personalizing web search results is not new, but is increasingly relevant as the sheer number of pages / sites available online rises at immense paces, with synonimity and homonimity compounding the problems of identifying search results that are truly relevant to the user's search [Tanudjaja and Mui, 2002].

The feasibility of creating such a personalization layer has been largely dependent on the availability of public services exposing an API for accessing the search functionality of major search engines. The development of the Prospector commenced in the spring of 2005, at which time we chose to implement it as a meta-layer to the Google search engine, due to the maturity of the respective API[1], and the overall standing of Google as the most popular search engine in the world. The Prospector utilizes Open Directory Project (ODP)[2] metadata for effecting user- and group-oriented re-ranking "on top" of the original search results. This data was originally provided by the Google Search API, and, in the current version of the system, is derived directly from the ODP site.

The design goals we had on the outset can be summarized as follows:

- The adaptation algorithm should be as simple as possible, and should be based on the users' interests (both at the individual- and group- levels), as these are inferred by characterizing search results, and identifying their thematic classification(s) within an established taxonomy. Although simple, the algorithm should be capable of supporting, directly or as extensions, standard features of similar systems, such as "aging" of user interests (see, e.g., [Koychev and Schwab, 2000]).
- Along the same lines as the adaptation algorithm, the derivation of the user and group models should be as simple as possible. Input to the modeling process was to include: (a) implicit ratings of search results, inferred from specific types of user behaviour (such as marking a result link as unsuitable, even without following it); and (b) explicit ratings of individual results by users.
- The personalization features of the Prospector, at the individual level, were to be available to users as an optional feature that requires registration (and, consequently, logging into the system prior to issuing search queries). Nevertheless, it was desirable to utilize group models to provide a "generic" level of adaptivity to non-registered users as well, on the basis of the thematic categories mentioned earlier.
- Last but not least, the Prospector was to provide a scrutable user model [Kay, 2000], enabling registered users to both inspect their user model, and modify it with respect to their interests.

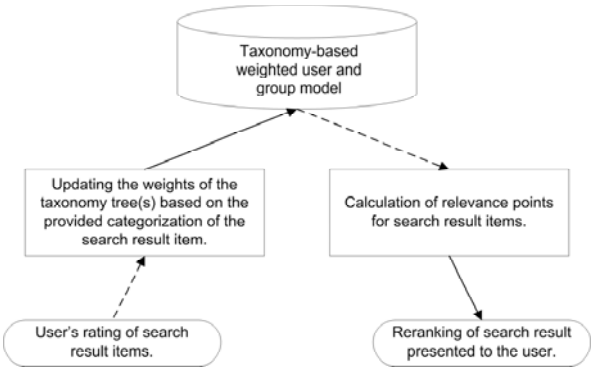


Figure 1: A synoptic view of adaptivity in the Prospector.

Using the categorization scheme introduced by Jameson [2003], the intended adaptive features of the

[1] For more information see: http://www.google.com/apis/
[2] Open Directory Project: http://dmoz.org/

Prospector could then be roughly summarized as depicted in Figure 1.

1.2 Related work

The Prospector can be broadly categorized as an adaptive information retrieval system. The literature on this type of systems is too extensive to cover here; interested readers may refer to [Micarelli and Sciarrone, 2004], and [Pierrakos *et al.*, 2003] for related work on adaptive information filtering and Web personalization.

A representative and widely acclaimed system that we will use as an example is I-SPY [Smyth *et al.*, 2003a; 2003b]. I-SPY implements an adaptive collaborative search technique that enables it to selectively re-rank search results according to the learned preferences of a community of users. Effectively I-SPY actively promotes results that have been previously favored by community members during related searches so that the most relevant results are top of the result list [Smyth *et al.*, 2003a]. I-SPY monitors user selections or "hits" for a query and builds a model of query-page relevance based on the probability that a given page will be selected by the user when returned as a result to a specific query.

Google has itself introduced two versions of personalized search functionality. The first version, unveiled in March 2004, allowed users to create a profile used to customize search results. By selecting categories, one could tell Google that they are interested in things like movies, radio and music. Then by using a slider, users could "personalize" their results to skew them toward their particular interest areas. This first version was based on classification of pages across the web into topics. The "personal" results were those skewed more toward the topics areas users were interested in, according to the profile they manually created.

The current incarnation of the service[3], follows an entirely different approach. Although details of the algorithms used have not been published to date, the general principles are as follows: (a) User profiles are built up by monitoring the user's search behaviour, as well as the links the user follows among the search results. (b) The only way in which users can modify their profiles is by removing items from their personal search history, which is where all information about past user behaviour is stored. (c) When this service is applied (although it is not clear under what conditions it is triggered), the search results are re-ranked to better suit the user's profile. When this happens, a link is also provided to allow the user to see the results in unmodified order.

The systems most relevant to the Prospector are the ones described in [Tanudjaja and Mui, 2002], and [Chirita *et al.*, 2005]. The first paper describes Persona, a system which utilizes ODP metadata for creating taxonomies of user interests and disinterests and tree coloring to represent user profiles. Taxonomy nodes visited are 'colored' by the number of times they have been visited, by user ratings if available, and by the URLs associated with the node [Tanudjaja and Mui, 2002]. The second paper describes a system with more similarities to the Prospector. Specifically, Chirita *et al.* [2005] have used ODP metadata to create user profiles, and then used various approaches to calculating the distance between a given search result and the user's profile, to decide that result's rank. Users pre-select ODP categories that they are interested in, for the creation of their profiles; the system does not have an adaptive component, so these profiles do not evolve over time. The distance calculation approaches range from what the authors term "naïve" –based primarily on graph node distances–, to a version of the PageRank [Brin and Page, 1998] algorithm modified to include a measure of the semantic similarity between nodes in an ODP taxonomy. User-based experiments have shown that these approaches to search personalization deliver superior results to their non-personalized counterparts [Chirita *et al.*, 2005].

The Prospector shares some of the characteristics, but also differs in several ways from the approaches outlined above. In synthesis, the Prospector: is intended as a meta-layer or front-end to a search engine; creates and utilizes user models built from explicit user ratings of items; represents user (dis-) interests using ODP-based thematic taxonomies; maintains (thematic) group models, and uses them in conjunction with the individual user model to adapt search results; and, supports user model scrutability.

2 The Prospector system

As already mentioned, the Prospector is a front-end to the Google search engine. A basic anonymous search returns exactly the same results as a search made directly on the Google site. Adaptivity comes into play in two guises: firstly, while still remaining anonymous, the user can have the results re-ranked by the system, according to thematically-based group profiles; and, secondly, users can register (and log in), progressively building up their personal interest profile, which is then used to automatically re-rank search results. The rest of this section outlines the overall functionality of the system, and then goes on to discuss the modeling and adaptivity aspects of the system in more detail.

2.1 Overall functionality

Anonymous searching

The main page of the Prospector is depicted in Figure 2. Apart from the search field itself, there is a set of links at the top of the page which allow users to register and log in, as well as to acquire additional information about the system and how it works. Search results appear under the search pane (gray area in Figure 2), as shown in Figure 3.

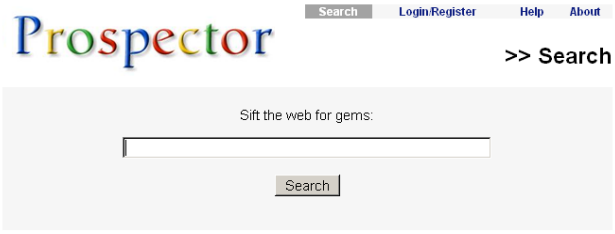


Figure 2: The main page of the Prospector.

Note that, above the search results, there is a form which lets one re-rank the search results according to the interest profile of a specific group (groups are organized according to thematic categories; more details are provided in the next section). When a re-ranking has been performed, the group whose profile was used for re-ranking is indicated under the form (see Figure 4). Also note that, whenever re-ranking takes place, the Prospector

[3] Available at: http://www.google.com/psearch

places a relevance score expressed as points next to each item. The derivation and significance of this score is explained in the next section.

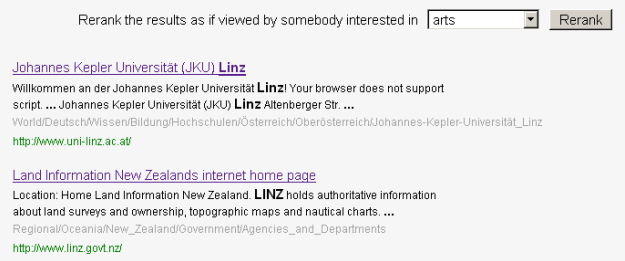


Figure 3: Unranked search results, as returned by Google.

Figure 4: Anonymous search results re-ranked according to the profile of the thematic group 'computers'.

User-aware searching

User-aware searching is available to users that are registered and logged in. This results in search results being automatically reordered according to the user's interest profile, and associated user model. Both of the later are accessible through respective entries in the link bar.

Typically, the first task of registered users that log in for the first time is to specify their interests. This is done through a form that contains a listing of selected top-level categories of Google's thematic taxonomy. Users are able to indicate that they have no interest, or use a 5-point scale to rate their interest, in a particular theme (see Figure 6). Users can return to this page at any time to adjust their entries.

From the available choices on this form, 'No interest' neither influences the ranking of results, nor alters the group profile when rating. The higher the extent of a user's interest in a group, the more the profile influences the reordering and the more a rating influences the group's profile.

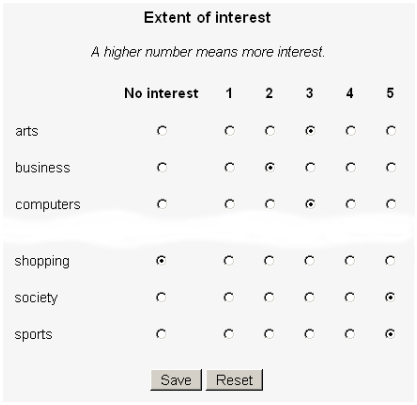


Figure 6: Specifying user interests (interest groups).

User-aware searching is done in much the same way as anonymous searching, as far as the interactive part of the system is concerned (see Figure 7). There are two noteworthy exceptions however: as a logged-in user one can disable a group-oriented reordering and rank the search result items according to the combined user and group profiles (using the link '[disable]' in Figure 7); and, the system adds a link after each result item which allows users to both remove the item from the list displayed and, at the same time, rate it negatively (this is the graphical link with the caption 'Unsuitable' in Figure 7).

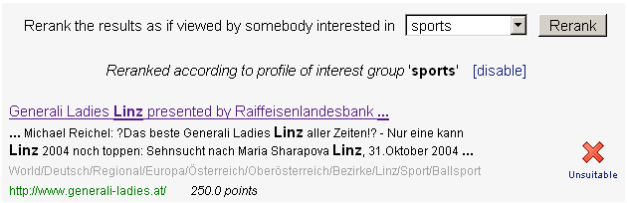


Figure 7: Search results for logged in users.

When a user follows a link among the search results, the respective page is shown with a rating frame above it (see Figure 7). Users can then rate a page positively ('Result OK') or negatively ('Result NOT OK') and can also choose whether to return to the search results display ('Take me back') or simply remove the rating frame ('Stop searching'). Rating a page negatively and returning to the

Figure 5: Rating frame above the actual result page.

search results automatically removes that page from the list of items.

The Prospector's interface also provides support for viewing and modifying one's own user model (following the "scrutability" principles of user modeling). This feature will be discussed in the next section, as an understanding of what is modeled and how is necessary for understanding that part of the system.

2.2 Modelling and adaptation

User and group modeling

Modelling of user and group interests is done using a thematic taxonomy. This taxonomy is derived from the hierarchical classification scheme provided by the Open Directory Project. In the first incarnation of Prospector, search results retrieved programmatically through the Google search API were annotated with a 'path' that identified the page's categorization within that hierarchy[4]. These paths formed the basis for user and groups models. The top-level categories in this scheme were the ones used to enable users to express their general thematic interests, as described in the previous section.

This approach had to be modified somewhat, when the ODP category stopped being included in the results returned by the Google search API[5]. There are, in general, two alternative ways in utilizing ODP metadata. The first is to use the data that ODP makes freely available from their site, and construct a layer / component that allows for querying that data directly. Although this approach requires custom development efforts, and makes for a quite heavy-weight system (the data alone is several hundred MBs *compressed*), one might argue that it is also the most cohesive and self-sufficient. An alternative approach, used as an intermediate solution in Prospector, is to utilize the ODP search functionality already available online. In short, this involves using a URL as the query string, and receiving as results the ODP categories (i.e., taxonomy nodes) under which the given URL has been classified. This solution can, of course, only serve as a temporary one, since it introduces a significant delay in the search, by requiring a request-response cycle to identify the ODP category (or categories) of each result returned by the underlying search engine.

Two types of user activity are taken into account for updating the user and group models: the user's removing an item from the search results as unsuitable; and, the user's providing an explicit, positive or negative, rating on a page after having visited it.

When either type of activity is encountered, the following take place, as far as the individual user model is concerned: if the 'path' corresponding to the rated item does not exist in the user model, it is added; for each node on the path the weight is changed to reflect the user's rating. Weight is subtracted or added to reflect a negative or, respectively, positive rating. The exact change in the weight in each node is affected by the following factors: (a) the "depth" of the node – more specific nodes are affected the most; and, (b) the user's interest ranking for the top-level category of the path – paths of higher interest to the user are affected more.

Group modeling in the Prospector functions along the same premises, and can be thought of as using predefined thematic clustering, with users "belonging" to different clusters with varying degrees of affinity (based on their self-expressed interest rankings). In other words, the Prospector maintains a group model for each of the top-level thematic categories. Apparently, this results in each group model representing a distinct portion of the overall taxonomy.

Updating of the group models occurs whenever there is an update in the model of an individual user that has any degree of affinity to the group. To start with, for each such change, paths are added as necessary. Subsequently, the changes in the individual user model are propagated to the group model. The impact of the weight propagations is itself weighted using the degree of affinity of the user to the group.

Adaptive reordering of search results

The primary adaptive function of Prospector is the reordering / re-ranking of search results. This is done using the current user's individual interest model, combined to varying degrees with the models of groups to which the user has some degree of affinity. Specifically, the models are used to calculate a relevance score for each item, which is in turn used to reorder items bringing the potentially most relevant ones to the top.

Let's take a closer look at the process of calculating the relevance score ("points") for a single search result. Assume that a single result item of a Google search belongs to the category "World / Sports / Basketball". To calculate the points, the categories of the search result item are retrieved, and the algorithm looks for the corresponding path through the portion of the taxonomy already represented in the model. So in our example we try to find a root element of the tree named "World". If it is found, the points assigned to the respective tree node are retrieved, and the algorithm continues with the next category entry ("Sports"), which it looks for among the children of the current tree node. If it is found, the points assigned to the tree node are added to the current sum. This process is repeated until the entire path is covered, or there is no node corresponding to a given path fragment.

As already mentioned the Prospector allows users to specify their interest in (or, degree of affinity to) a group, using a 5-point scale. This "degree" is used to repeat the above described process using the models of each of the groups to which the user belongs. The result of this process is that the ranking of each item is influenced by the ranking of other group members for the category under which the item is classified.

In practice, the ranking algorithm does not semantically distinguish between group models and the user's personal model. Instead, it receives a list of models and a list of weights that define the "importance" of each. The highest weight is assigned to the model corresponding to the user, which renders it the primary factor in determining an item's rating, but still leaves plenty of room for benefiting from groups ratings on categories that the user has not rated yet (thus addressing the "bootstrapping" problem for individual models).

Individual models can both be inspected and modified by users using forms like the one shown in Figure 8. Tree nodes with a positive weight represent categories the user

[4] To be precise, this was actually the path in the hierarchy maintained under Google's own directory service (http://directory.google.com), which, however, is practically identical to the ODP, as it is based on the same data.

[5] This was a decision on the part of Google, who announced that this change is to be considered a permanent one.

appears to be interested in according to the ratings performed, with negatively rated nodes represent uninteresting categories. Nodes with a weight of 0.0 don't contribute to the ranking of a result item. The weights can be changed to directly alter the user profile. When checking the box in the column 'Disable sub-tree' the corresponding node and all its sub-nodes are set to a weight of 0.0.

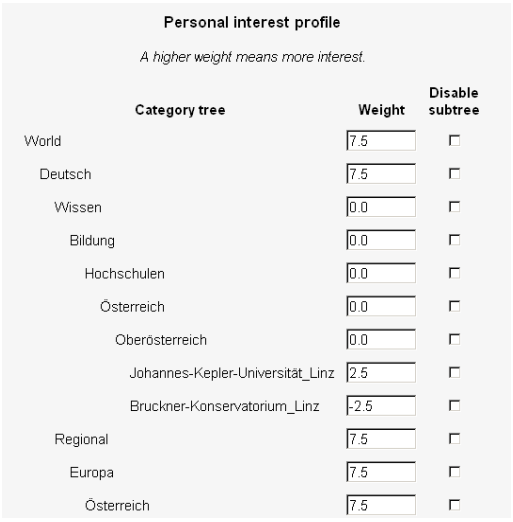


Figure 8: Viewing and modifying the user model.

3 Discussion

Preliminary evaluation

To evaluate the system, we have engaged in two types of preliminary evaluation activities. Firstly, we conducted a informal, usability-oriented heuristic evaluation with the assistance of affiliated usability experts. Secondly, we asked members of our institute to use the system for the period of one day, and asked them to provide their feedback on an individual basis, without prescribing the type or range of input we were looking for.

Very important technical limiting factors which constrained the types of assessment feasible at this stage, and may have also influenced the results, were that each search query returned only 50 results (in 5 pages of 10 results each), and took considerably longer than it would have done, had it been issued directly on the Google site. Both of these constraints were due to limitations in the Google API, which returns only 10 result items per query request. We decided to perform 5 such requests, to have a sufficient amount of links to work with. This however meant that a normal search, from the perspective of the user, took approximately 5 times as long as they might have expected. Experts and users participating in the studies had been advised regarding these limitations, and were asked to try and disregard them as much as possible.

The findings of this preliminary informal round can be summarized as follows:

- Users were in general positively disposed towards, and rather satisfied with the effects of reordering. It should be noted however, that there were no control settings, and users had no way of comparing the reordered results to the ones they would have gotten directly from Google, other than repeating the search on the Google site. Furthermore, we anticipate that the results would have been even better, if the system was used over a longer period of time, thus having more opportunity to build more comprehensive user and group models.
- The relation between the top-level categories (used to rank user interests in different themes) and the user model representation (where categories are associated with weights) was found to be confusing. Specifically, users found it hard to anticipate how changes in one part of the system affected the other. They were also unclear about the effects of weights on the search results.
- Another source of confusion was the use and exposure of arbitrary weights for items. The way the modeling and adaptation algorithms work at the moment results in weights that may vary significantly in range. Although this has beneficial effects on the ranking, it is apparently not easy to comprehend. It also presents an interactivity challenge, as users have to guess what weights might be appropriate for nodes in their model, without having any semantic interpretation of the available ranges.

Planned improvements

Based on these preliminary results, we have planned a number of improvements to be made to the system. These include:

- The unification of the top-level categories used to derive the initial user profile, with the contents of the user model. Specifically, we intend to eliminate the differentiation between the two, and simply use the initial user input to seed the user model. Along the same lines, we will also eliminate the “Interests” view and use the user model view as the only one from where the user’s model can be inspected and modified.
- The modelling and adaptation algorithms will be modified to use a probabilistic approach to weighting, instead of the current unconstrained one. Specifically, we intend to apply two correlated types of normalization in the derivation of weights for items, and in the modifications of node weights on the basis of user ratings. This may have a somewhat negative impact on the ranking results, but will provide for a concise and easy to comprehend range of values in the models.
- Following from the previous step, we intend to modify the representation of weights in the user model view, replacing the weight text field with a pseudo-continuous scale that can be adjusted by means of a slider control.
- Finally, an additional option will be added to the interface for logged in users, which will allow them to view the search results without any ranking applied to them. Although this feature is expected to be of limited utility in the general case, it is expected to have at least two important benefits: firstly, it may prove useful in building up user trust in the system’s adaptive behaviour; secondly, it may serve as a “backup” option for cases where wrongly inferred

(dis-) interests at the user- or group- levels skew results in an undesirable direction.

Further to the above, we intend to release the Prospector as open source software. We feel that the modularity of the system render it a potentially interesting "playground" for adaptive / personalized search work, or, alternatively, a viable tool for hands-on work in adaptive hypermedia courses. Fr instance, Prospector allows, among other things, for: easy migration to different search engines; the implementation of alternative modeling and adaptation algorithms; the modification of the user interface, independently of the adaptive core; etc.

Planned evaluation activities

Once the improvements described in the previous section are in place, Prospector will undergo two evaluations.

- The first will be performed in October 2006 by the Department of Technical and Professional Communication, Faculty of Behavioural Sciences, University of Twente (the Netherlands), and will have a strong focus on the user. In this study, Prospector will be evaluated as an adaptive system *per se*, but also used as a case study for determining the appropriateness of different evaluation methods in assessing how users experience adaptivity.
- In a second stage we intend to engage in a full-scale evaluation of the system, using the layered evaluation framework for adaptive systems, introduced by Paramythis and Weibelzahl [2005]. In fact, we intend to approach the formal assessment of the system from two different directions: On the hand, we will use the aforementioned evaluation framework to validate the different layers of the adaptive system. On the other hand, we will use the two "generations" of the system to validate the framework. The validity study will address both the granularity of the layers, and the proposed per-layer evaluation criteria, methods and instruments.

4 Summary and conclusions

This paper has presented the Prospector system, an adaptive front-end to the Google search engine, which re-ranks results according to user- an group- interests, identified and represented according to ODP-based thematic taxonomies. We have tried to follow an as simple as possible approach both in modeling and in effecting adaptivity. The primary aim in doing so has been to make Prospector as portable as possible across different search engines, or search frameworks in general, with the overarching goal being to develop a generalised adaptive front-end for personalized searching.

Preliminary informal evaluation activities have provided encouraging results and valuable input for continued work on the system. We plan to apply the improvements outlined herein, and engage in a full-scale empirical evaluation of the system, using the layered evaluation approach.

References

[Brin and Page, 1998] S. Brin, and L. Page. The Anatomy of a Large-Scale Hypertextual Web Search Engine. In *Proceedings of the 7th World Wide Web Conference*, Brisbane, Australia, April 1998. *Computer Networks and ISDN Systems*, 30(1-7): 107-117, 1998.

[Chirita *et al.*, 2005] P. A. Chirita, W. Nejdl, R. Paiu, and C. Kohlschütter. Using ODP Metadata to Personalize Search. In *Proceedings of the 28th ACM International SIGIR Conference on Research and Development in Information Retrieval*, pages Salvador, Brazil, August 2005. ACM.

[Jameson, 2003] A. Jameson. Adaptive Interfaces and Agents. In *Human-computer interaction handbook*, pages 305–330. Erlbaum, Mahwah, NJ, 2003.

[Kay, 2000] J. Kay. Stereotypes, Student Models and Scrutability. In *Proceedings of the 5th international Conference on intelligent Tutoring Systems*, pages 19-30, Montréal, Canada, June 2000 (Lecture Notes In Computer Science, vol. 1839). Springer-Verlag, Berlin.

[Koychev and Schwab, 2000] I. Koychev, and I. Schwab. Adaptation to Drifting User's Intersects. In *Proceedings of ECML2000/MLnet workshop "Machine Learning in the New Information Age"*, pages 39-45, Barcelona, Spain, May-June 2000.

[Micarelli and Sciarrone, 2004] A. Micarelli, and F. Sciarrone. Anatomy and Empirical Evaluation of an Adaptive Web-Based Information Filtering System. *User Modeling and User-Adapted Interaction*, 14(2-3): 159-200, 2004.

[Paramythis and Weibelzahl, 2005] A. Paramythis, and S. Weibelzahl. A Decomposition Model for the Layered Evaluation of Interactive Adaptive Systems. In *Proceedings of the 10th International Conference on User Modeling* (UM2005), pages 438-442, Edinburgh, Scotland, UK, July 2005 (Lecture Notes in Computer Science LNAI 3538, Springer Verlag). Springer-Verlag, Berlin.

[Pierrakos *et al.*, 2003] D. Pierrakos, G. Paliouras, C. Papatheodorou, and C.D. Spyropoulos. Web Usage Mining as a Tool for Personalization: A Survey. *User Modeling and User-Adapted Interaction*, 13(4): 311-372, 2003.

[Smyth *et al.*, 2003a] B. Smyth, E. Balfe, P. Briggs, M. Coyle, and J. Freyne. Collaborative Web Search. In: *Proceedings of the 18th International Joint Conference on Artificial Intelligence* (IJCAI-03), pages 1417-1419, Acapulco, Mexico, August 2003. Morgan Kaufmann.

[Smyth *et al.*, 2003b] B. Smyth, J. Freyne, M. Coyle, P. Briggs, and E. Balfe. I-SPY: Anonymous, Community-Based Personalization by Collaborative Web Search. In *Proceedings of the 23rd SGAI International Conference on Innovative Techniques and Applications of Artificial Intelligence*, pages 367-380, Cambridge, UK, December 2003. Springer.

[Tanudjaja and Mui, 2002] F. Tanudjaja, and L. Mui. Persona: A Contextualized and Personalized Web Search. In *Proceedings of the 35th Annual Hawaii International Conference on System Sciences (HICSS'02)-Volume 3*, pages 67 (9), Hilton Waikoloa Village, Island of Hawaii, January 2002. IEEE Computer Society.

Workshop Information Retrieval 2006 of the Special Interest Group Information Retrieval (FGIR)

9.-13. October 2006, Universität Hildesheim

Norbert Fuhr, Sebastian Goeser, Thomas Mandl

Information Retrieval has become a key technology in the knowledge society. The use of search engines has risen dramatically. Search engines are part of everyday life of most internet users and they demonstrate the advantages and limitations of information retrieval methods. The ubiquity of search systems has led to the application of information retrieval technology in many new contexts (e.g. mobile and international) and for new object types (products, patents, music). In order to develop appropriate products, basic knowledge on information retrieval needs to be revisited and innovative approaches need to be taken. The quality of information retrieval needs to be evaluated for each context. Large evaluation initiatives respond to these challenges and develop new benchmarks.

The workshop Information Retrieval 2006 provides a forum for scientific discussion and the exchange of ideas. The workshop of the Special Interest Group for Information Retrieval within the German Gesellschaft für Informatik (GI) takes place in the week of workshops LWA "Learning, Knowledge and Adaptivity“ (LWA, 9.-13. Oct. 2006 at the University of Hildesheim, Germany). This workshop continues a successful series of conferences and workshops of the special interest group on information retrieval (http://www.fg-ir.de).

The workshop received 25 submissions from six countries. Seven submisssions were accepted as full papers.

We received submissions on the following topics:

- Development and optimization of retrieval systems
- Retrieval with structured and und multimedia documents
- Evaluation and evaluation research
- Text mining and information extraction
- Multilingual systems
- Digital libraries
- User interfaces and user behavior
- Machine learning in information retrieval
- Information retrieval and knowledge management

Keynote:

We are grateful to Holtzbrinck eLab GmbH for their financial support which made one keynote possible:

- Mark Sanderson (University of Sheffield):
  Retrieval from Cultural Heritage Collections

Program Chairs:

- Prof. Dr. Norbert Fuhr, Universität Duisburg-Essen
- Dr. Sebastian Goeser, IBM Germany Development
- PD Dr. Thomas Mandl, Universität Hildesheim

Program Committee:

- Prof. Dr. Martin Braschler, University of Applied Science and Technology, Zürich
- Prof. Dr. Norbert Fuhr, University of Duisburg-Essen
- Dr. Sebastian Goeser, IBM Germany Development
- Prof. Dr. Andreas Henrich, University of Bamberg
- Prof. Dr. Gerhard Knorz, University of Applied Science and Technology at Darmstadt
- Dr. Johannes Leveling, University of Hagen
- PD Dr. Thomas Mandl, University of Hildesheim
- Prof. Dr. Marc Rittberger, DIPF, Frankfurt
- Dr. Ralf Schenkel, Max-Planck-Institute for Computer Science, Saarbrücken
- Dr. Peter Schäuble, Eurospider AG, Zürich, Switzerland
- Dr. Ulrich Thiel, FhG-IPSI, Darmstadt
- Dr. Gregor Thurmair, Linguatec, Munich
- Prof. Dr. Gerhard Weikum, Max-Planck-Institute for Computer Science, Saarbrücken
- Prof. Dr. Christian Wolff, University of Regensburg
- Prof. Dr. Christa Womser-Hacker, University of Hildesheim

Initial Observations on Query Based Sampling in Distributed CLIR

Xiao Mang Shou, Mark Sanderson
Department of Information Studies
University of Sheffield
Western Bank, Sheffield S10 2TN, UK
[x.m.shou, m.sanderson]@shef.ac.uk

Abstract

Cross Language Information Retrieval (CLIR) enables people to search information written in different languages from their query languages. Information can be retrieved either from a single cross lingual collection or from a variety of distributed cross lingual sources. This paper presents initial results exploring the effectiveness of distributed CLIR using query-based sampling techniques, which to the best of our knowledge has not been investigated before. In distributed retrieval with multiple databases, query-based sampling provides a simple and effective way for acquiring accurate resource descriptions which helps to select which databases to search. Observations from our initial experiments show that the negative impact of query-based sampling on cross language search may not be as great as it is on monolingual retrieval.

1 Introduction

Cross Language Information Retrieval (CLIR) is the process of retrieving documents written in a language(s) different from the language of the query. In recent years much CLIR research was undertaken in the academic communities via the academic evaluation forums like CLEF, NTCIR and TREC and a number of application areas have been developed over the years. CLIR is basically a combination of machine translation and traditional monolingual IR and four approaches commonly used for translation include [Gollins, 2000]: (1) a controlled vocabulary, (2) machine translation, (3) bilingual parallel corpora, (4) bilingual dictionaries, or more recently a combination of all approaches. When doing translation, one can either translate the query into the target language (query translation, QT), translate search documents into the query language (document translation, DT), or translate both queries and documents into a common language [Oard, 1997]. So far, query translation, which transforms a user's query into the language of the documents, is the dominant approach because this can be made to work successfully with simple translation methods and does not require the overhead of translating collection documents which is often computationally expensive. With the right approach, CLIR systems are able to achieve retrieval effectiveness that is only marginally degraded from the effectiveness achieved had the query been manually translated [Ballesteros & Croft, 1998].

Current CLIR research focuses on improving retrieval effectiveness under monolingual, bilingual or multilingual conditions. However, how to process multilingual information in a distributed environment has not yet been sufficiently explored. In distributed retrieval with multiple multilingual resources (referred to here as databases), the common approach is to translate queries into the resource language for retrieval and then results from individual collections are merged into a single list. Using this method, similar to monolingual distributed retrieval, when there are a large number of databases, it can be difficult to choose which databases to search. This situation exacerbates in a multilingual environment. Obtaining cross language resource descriptions of each database automatically and efficiently becomes necessary.

Query-based sampling (QBS) [Callan and Connell, 2001] is a technique used for acquiring resource descriptions of databases by running queries on the databases examining text of the documents returned, seeking new queries from the text and using the text to build a uni-gram language model of the database content. Empirically, results demonstrated that sampling 300-500 documents from each database appears to be effective for resource description across different rage of database sizes. This method is particularly useful in distributed retrieval to decide which databases to search according to a given query. To the best of our knowledge, there has been no investigation of QBS and CLIR in the past. Therefore, in our experiments, we tested QBS with query translation, document translation and both query and document translation together for distributed retrieval using the CLEF2003 Italian collection. QBS Results were compared with original distributed monolingual retrieval results. What we report here are a series of observations based on our initial experiments.

This paper divides into the following sections: section 2 describes our experimental set up, section 3 presents our results and compares these results with original monolingual baseline, section 4 list some directions for future work and section 5 summaries our findings.

2 Experimental Setup

2.1 The CLEF2003 Test Collection

The test data set we use is CLEF (http://clef.isti.cnr.it/) 2003's Italian collection (157,558 documents, average

document length 214, overall size 370MB) and the 60 query topics (141- 200) with title, description and narrator fields. The same 60 query topics are available in eight languages including Italian and English by human translation.

For additional comparison, CLEF2003's English collection was tested for query translation using both original English topics and machine translated Italian to English topics.

2.2 Distributed Retrieval and Query-Based Sampling

In our experiments, we tested cross language retrieval using distributed retrieval methods where the original Italian collection was divided into 30 sub databases by document order in collection with each of them containing around 5,252 documents. Distributed retrieval was performed based on the 30 databases using both complete and sampled resource descriptions. Resource descriptions store information about what each database contains. A complete resource description is generated using full collection index whereas sampled resource description (sometimes called learned resource description) is generated by QBS. In our experiments, for each of the 30 sub databases, 300 documents were selected to obtain the sampled resource description.

In parallel, the same set of retrievals were run for the translated Italian to English collection and CLEF2003 English collection as well for further comparison.

2.3 Translation Resources and Retrieval System

The machine translation tool used in our experiments was Systran Professional Premium 5.0's MultiTranslate Utility (http://www.translation.net/systran_professional.html) which can translate multiple files in batches. The whole Italian collection was translated into English and this process took about one month running on a "standard desktop PC". Using the same tool, original Italian query topics were translated into English and English topics were translated into Italian.

The retrieval tool we used for Italian and English retrieval was Lemur3.1 (http://www.lemurproject.org/). Lemur supports distributed retrieval providing functions to rank databases by their resource descriptions and merge their distributed search results using the CORI algorithm [Callan, 2000]. The default setting for Lemur in our experiments was to retrieve the top 30 ranked documents from the top 10 ranked databases.

Since CLEF data format was not compatible with Lemur and Systran formats, conversion of the query and document format was necessary.

2.4 Cross-Language Retrieval

Given the Italian collection and queries and their translated English versions, we compare query versus document translation alone as well as applying both query and document translation. Before the experiment, the data collection and queries were processed. We first translated the Italian text and queries into English and English queries into Italian using the MT system. Next, stopwords were removed using stopword lists provided by the Snowball stemmer (http://snowball.tartarus.org). We then applied stemming using Snowball and removed diacritics in Italian using the UNIX recode tool. To perform this, we recoded the character set from latin1 to HTML and then replaced the HTML characters by their original ASCII characters. Finally, all characters were converted to lower case. After the process, all collection texts were split into sub databases to be indexed and retrieved by Lemur.

The following experiments were performed with each of them applied to distributed retrieval using both complete and sampled resource descriptions:

1. Retrieval using the original Italian collection and topics. This will be used as a baseline in comparison (monolingual).
2. Query translation with original Italian collection and English topics translated to Italian (QT).
3. Document translation with the Italian collection translated to English and original English topics (DT).
4. Both query and document transition with the Italian collection translated to English and the Italian topics translated to English (QT+DT).

The English collection was tested under monolingual and query translation configurations. Precision at rank 5, 10, 15, 20 and 30 (P5, P10, P15, P20, P30) was used to measure retrieval effectiveness. In addition, recall and the number of queries with any relevant documents retrieved (rel_q) was also computed.

3 Results

Results are listed in Tables 1, 2 and 3: the tables show results covering a number of configurations of the collection and topics. Each table collates the results for a particular form of collection: in Table 1, the native Italian collection; table 2, native English; and in table 3, Italian collection translated to English. In each table two forms of query are shown and within each query the search on the full (columns 1&3) and sampled resource descriptions (columns 2&4) are compared. Since we did not have time to translate the English collection to Italian, results for that configuration are not shown.

	Italian collection, Italian topics (monolingual)		Italian collection, English -> Italian topics (QT)	
	1	2	3	4
	Complete resource description	Sampled resource description	Complete resource description	Sampled resource description
P5	0.3133	0.3067 (-2.1%)	0.2667	0.2533 (-5.0%)
P10	0.2483	0.2333 (-6.0%)	0.2033	0.2067 (+1.7%)
P15	0.2111	0.2033 (-3.7%)	0.1711	0.1744 (+2.0%)
P20	0.1867	0.1808 (-3.2%)	0.1467	0.1467 (0.0%)
P30	0.1511	0.1406 (-7.0%)	0.1128	0.1183 (+4.9%)
rel_q	43/51	41/51 (-4.7%)	39/51	40/51 (+2.5%)
recall	272/809	253/809 (-7.0%)	203/809	213/809 (+4.9%)

Table 1. Italian monolingual and QT using complete and sampled resource descriptions

	English collection, English topics (monolingual)		English collection, Italian -> English topics (QT)	
	1	2	3	4
	Complete resource description	Sampled resource description	Complete resource description	Sampled resource description
P5	0.4100	0.3367 (-10.6%)	0.2933	0.2500 (-14.8%)
P10	0.3100	0.2817 (-9.1%)	0.2250	0.2050 (-8.9%)
P15	0.2633	0.2456 (-6.7%)	0.1989	0.1878 (-5.6%)
P20	0.2400	0.2200 (-8.3%)	0.1817	0.1675 (-7.8%)
P30	0.2028	0.1806 (-11.0%)	0.1572	0.1411 (-10.2%)
rel_q	47/54	44/54 (-6.4%)	42/54	39/54 (-7.1%)
recall	365/1006	325/1006 (-11.0%)	283/1006	254/1006 (-10.2%)

Table 2. English monolingual and QT using complete and sampled resource description

	Italian -> English collection, English topics (DT)		Italian -> English collection, Italian -> English topics (QT+DT)	
	1	2	3	4
	Complete resource description	Sampled resource description	Complete resource description	Sampled resource description
P5	0.2133	0.1867 (-12.5%)	0.3133	0.2467 (-21.3%)
P10	0.2017	0.1800 (-10.8%)	0.2683	0.2133 (-20.1%)
P15	0.1833	0.1489 (-18.8%)	0.2322	0.1911 (-17.7%)
P20	0.1625	0.1342 (-17.4%)	0.2008	0.1667 (-17.0%)
P30	0.1283	0.1111 (-13.4%)	0.1611	0.1322 (-17.9%)
rel_q	39/51	37/51 (-5.1%)	46/51	41/51 (-10.7%)
recall	231/809	200/809 (-9.7%)	290/809	238/809 (-17.9%)

Table 3. Italian DT and QT+DT using complete and sampled resource description

As can be seen across all the tables, with one exception (in Table 1), query-based sampling reduces effectiveness compared to retrieval based on the full resource description. The one exception to this is under the QT condition where use of the sampled resource description resulted in improved effectiveness (average of the % difference in column 4 of Table 1 is +0.72%). In addition, more relevant documents were retrieved and more queries with at least some relevant documents retrieved were found than with the full resource description. However, the improvement was not significant. In general, as would be expected, QBS reduced retrieval effectiveness. The reductions observed in Tables 1 & 2 are in-line with reductions reported from the original QBS paper [Callan and Connell, 2001].

The reductions in Table 3 are larger. Here collection translation is being used, with the Italian collection being translated into English. Errors will be made in the translation process and whether those errors are somehow causing problems in the resource description process of QBS, is an area of investigation to be examined in the future.

3.1 Using translation to enhance monolingual search?

Separate from QBS, we report one other result. Comparing column 3 of Table 3 with column 1 of Table 1, both configurations are the same taking Italian queries retrieving on the same CLEF Italian collection. The results in the column of Table 3 however shows monolingual retrieval after both queries and documents are translated into another language, in this case English. Performing QT and DT together resulted in a 6.4% precision improvement on average over original monolingual retrieval with no translation. This gain is consistent after rank 10, but not significant. Even under QBS condition, comparing results in column 2 in Table 1 with column 4 in Table 3, performing QT and DT together resulted in 9.6% precision drop on average over monolingual retrieval which is still comparably effective.

Our observation that translating both collection and queries from Italian to English together outperforms its originally monolingual baseline is to the best of our knowledge new. Franz and McCarley's tried improving monolingual retrieval by query and document translation (as done here) working on an English test collection, who's documents and queries were translated into French [Franz and McCarley, 1999]. With their experiments, however, they reported a 10-20% drop in effectiveness compared to monolingual baseline.

Whether our observation is an outlier case or an indication of a general trend will be the subject of further work.

4 Future Work

In our experiments, we only managed to translate the CLEF2003 Italian collection and topics into English and English topics to Italian. Due to the restriction of collection, the query-based sampling size is approximately 5% of each database and may be impractical for larger collections. In order to better understand the importance, significance and durability of our observations, we will extend our work to larger collection size and other language translations. Further future work includes translating other languages such as German into English and observes whether the specialty of compound words in German will result in different performance in CLIR. Since document translation are time consuming, other translation resources such as dictionary look up, parallel texts based translation [Chen and Gey, 2003] or statistical translation models built by GIZA++ (http://www.fjoch.com/GIZA++.html) can be applied for future experiments as well. We also will translate English collection to Italian and run the same set of experiments on it to test the bilingual distributed CLIR and to better understand the drops in effectiveness when using query-based sampling and document translation.

5 Conclusion

In this paper we have presented experiments incorporating machine translation of both the queries and documents into an IR engine for distributed CLIR using Systran Professional and CLEF2003 Italian collection. Distributed CLIR was run using a complete resource description of all collections or using sampled resource collection by query-based sampling technique. The obtained results were compared with the results from original monolingual retrieval and results from applying query and document translation. In Italian distributed CLIR, we observed that QT using QBS sampled resource description performed better than DT at the same condition, and its performance

was comparably effective as using complete resource description. Furthermore, we also found that when translating both queries and documents together, distributed cross language retrieval is almost as good as monolingual retrieval either using complete resource description or QBS sampled resource description.

Acknowledgments

The work described in this paper was funded as part of the EU 6th Framework project, BRICKS - Building Resources for Integrated Cultural Knowledge Services. Further information on the project can be found at http://www.brickscommunity.org/.

References

[Ballesteros and Croft, 1998] Lisa Ballesteros and Bruce Croft. Resolving Ambiguity for Cross Language Retrieval. In *Proceedings of the 21st International Conference on Research and Devlelopment in Information Retrieval,* p64-71.

[Franz and McCarley, 1999] Martin Franz and J. Scott McCarley. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, p295-296.

[Gollins, 2000] Timothy John Collins. Dictionary Based Transitive Cross-Language Information Retrieval using Lexical Triangulation. *Masters Dissertation.* Department of Information Studies, University of Sheffield.

[Oard, 1997] Doug Oard. Serving Users in Many Languages. D-Lib.

[Callan, 2000] Jamie Callan. Distributed Information Retrieval. W. B. Croft Editor, Kluwer Acedemic Publishers 2000, *Advances in Information Retrieval*, p127-150

[Callan and Connell, 2001] Jamie Callan and Margaret Connell. Query-Based sampling of Text Databases. In *ACM Transactions on Information Systems (TOIS),* Volume 19 , Issue 2 (April 2001), p97-130

[Chen and Gey, 2003] Aitao Chen, Fredric C. Gey. Combining Query Translation and Document Translation. in Cross-Language Retrieval. In *CLEF 2003*, p108-121

Ansätze zur Bestimmung von Locality für deutsche Webseiten

Raiko Eckstein, Andreas Henrich, Volker Lüdecke
Otto-Friedrich-Universität Bamberg
D-96045, Bamberg, Deutschland
mail@raikoeckstein.de, {andreas.henrich|volker.luedecke}@wiai.uni-bamberg.de

Abstract

Das geographische Information Retrieval (GeoIR) berücksichtigt bei Suchanfragen – insb. nach Webseiten – neben dem Inhalt von Dokumenten auch eine räumliche Komponente, um gezielt nach Seiten suchen zu können, die für eine spezifische Region bedeutsam sind. Dazu müssen GeoIR-Systeme den geographischen Kontext einer Webseite erkennen können und in der Lage sein zu entscheiden, ob eine Seite überhaupt regional-spezifisch („lokal“) ist oder einen rein informativen Charakter besitzt, der keinen geographischen Bezug besitzt.

Im Folgenden werden Ansätze vorgestellt, Merkmale lokaler Seiten zu ermitteln und diese für eine Einteilung von Webseiten in globale und lokale Seiten zu verwenden. Dabei sollen insbesondere die sprachlichen und geographischen Eigenschaften deutscher Webseiten berücksichtigt werden.

1 Einführung

Nach einer Studie von [Sanderson and Kohler, 2004] weisen etwa 20% aller Suchanfragen im Web einen geographischen Kontext auf. Das bedeutet, dass in diesen Fällen insbesondere Seiten gesucht werden, die einen bestimmten lokalen Bezug haben. Herkömmliche Suchdienste sind bei der Anfragebearbeitung auf die in der Anfrage verwendeten Begriffe beschränkt, die über boolesche Operatoren miteinander verknüpft werden können. Eine Suche nach „Hotel“, „bei“ und „Bamberg“ würde dementsprechend nur solche Webseiten liefern können, die alle Begriffe enthalten, unabhängig von den tatsächlichen geographischen Kontexten der jeweiligen Seiten. Das Ergebnis wäre, dass viele der Seiten im Suchergebnis keinen direkten Bezug zu Bamberg hätten, sondern beispielsweise zu Reisebüros oder Hotelsuchmaschinen gehörten. Auch würden Seiten von Hotels aus dem Umland von Bamberg dadurch in der Regel nicht gefunden, wenn dort der Begriff „Bamberg“ nicht enthalten ist. Die geographische Nähe würde also ebenfalls nicht berücksichtigt.

Spezialisierte geographische Suchmaschinen sollten den geographischen Aspekt des Informationswunsches eines Benutzers erkennen und in der Anfragebearbeitung explizit berücksichtigen können. Im Idealfall würden dem Benutzer die geographischen Kontexte der Ergebnisse seiner Suche z. B. mit Hilfe einer Karte veranschaulicht werden.

Das geographische Information Retrieval beschäftigt sich mit der Nutzbarmachung des geographischen Kontexts von Dokumenten. Ein solcher Kontext besteht aus einem Ort oder einer Menge von Orten („Location“), auf die in einer Seite Bezug genommen wird, was auch als der geographische Fokus einer Seite bezeichnet wird. Daneben kann die *Locality* den Grad der Lokalität einer Seite angeben. Mit Hilfe von *Locality* soll entschieden werden können, ob der Inhalt eines Dokuments für eine lokal oder regional eingegrenzte Nutzergruppe relevant ist – wie z. B. bei einer Tourismus- oder Handwerkerseite – oder ob dieser als ortsunabhängig von Interesse und somit eher „global“ einzuordnen ist, wie es beispielsweise bei Bedienungsanleitungen der Fall ist.

Ein geographischer Suchdienst muss dementsprechend in der Lage sein, den geographischen Kontext einer Seite automatisiert zu erfassen und sowohl Location als auch *Locality* bei der Anfragebearbeitung zu berücksichtigen. In diesem Beitrag sollen Ansätze zur automatischen Bestimmung von *Locality* für deutsche Webseiten beschrieben werden. Dabei werden zunächst „white box“ Ansätze betrachtet, die mit dem Ziel verfolgt wurden, nachvollziehbare Anhaltspunkte für *Locality* zu erhalten. Anschließend werden auch lernende Verfahren wie Support Vector Machines (SVM) eingesetzt.

Im folgenden Abschnitt werden verwandte Arbeiten vorgestellt. In Abschnitt 3 werden unsere Untersuchungen zur Ermittlung spezifischer Merkmale von *Locality* beschrieben. Eine vergleichende Betrachtung und Bewertung der Ergebnisse erfolgt in Abschnitt 4. Schließlich wird eine Zusammenfassung und ein Ausblick auf weitere Arbeiten gegeben.

2 Verwandte Ansätze

In der Literatur wird beim geographischen Kontext nicht immer eindeutig zwischen Location und *Locality* unterschieden. Deswegen sollen in diesem Abschnitt verwandte Ansätze vorgestellt werden, die sich mit der Bestimmung des geographischen Kontexts bzw. Fokus befassen.

[Buyukkokten *et al.*, 1999] adressieren das Problem des *Geographical Scope*. Mit Hilfe der Registrierungsdaten einer Domain sowie anhand von identifizierten Telefonvorwahlen und Postleitzahlen auf einer Webseite wird versucht, eine geographische Ausdehnung zu ermitteln, auf die sich diese Webseite bezieht.

[McCurley, 2001] beschäftigt sich mit dem geographischen Indexieren von Webseiten und einer geographischen Navigation zwischen ihnen. Eine geographische Einordnung erfolgt bei diesem Ansatz anhand von Indikatoren, die durch Auswertung des Seiteninhalts gewonnen werden. Neben der Sprache, in der ein Dokument geschrieben ist, und die ein grober Anhaltspunkt für einen geographischen

Kontext sein kann, analysiert McCurley Adressen und Telefonnummern und beschäftigt sich mit den Problemen, die aus einer Vielzahl möglicher Formate entstehen. Schließlich wird versucht, weitere Informationen aus der Verlinkung von Webseiten zu gewinnen. Es wird hier versucht, jeder Seite einen geographischen Kontext zuzuordnen und dabei nicht zwischen lokalen und globalen Seiten unterschieden.

Einen anderen Weg beschreiten Amitay, Har'El, Sivan und Soffer in [Amitay *et al.*, 2004]. Das dort vorgestellte System *Web-A-Where* versucht, Webseiten einer bestimmten geographischen Region zuzuordnen und befasst sich in erster Linie mit der Auflösung von Mehrdeutigkeiten. Diese treten im GeoIR auf, wenn derselbe Ausdruck für unterschiedliche geographische Orte steht oder ein Ausdruck sowohl eine geographische als auch eine nicht-geographische Bedeutung tragen kann. Der in diesem System verwendete *Geotagger* extrahiert potentielle Toponyme aus Webseiten und ordnet diese einem Knoten in einer Taxonomie für die USA zu, die aus den Ebenen Stadt, Bundesstaat und Land besteht. Diese Einordnung dient anschließend zur Auflösung der genannten Mehrdeutigkeiten und zur Bestimmung eines geographischen Fokus.

Ding, Gravano und Shivakumar versuchen in [Ding *et al.*, 2000] auf ähnliche Weise, den geographischen Fokus von Webseiten zu bestimmen. Auch sie teilen das Gebiet der USA in eine dreistufige Hierarchie auf und betrachten die Linkstruktur sowie Toponyme im Seiteninhalt.

Zhang et al. versuchen in [Zhang *et al.*, 2006] Suchergebnisse unter Berücksichtigung von geographischer Ähnlichkeit zu ranken und nutzen dafür die Linkstruktur zwischen Seiten und eine lokal vorhandene Datenbasis mit Beispieleinträgen zum gesuchten Thema.

Markowetz, Brinkhoff und Seeger unterscheiden in [Markowetz *et al.*, 2004] zwischen Seiten von lokalem und globalem Interesse. Sie skizzieren in dieser Arbeit ihre Idee, diese *Locality* anhand ein- und ausgehender Links zu berechnen, ohne sie jedoch weiter zu verfolgen.

Den angeführten Ansätzen ist gemein, dass zwar eine geographische Zuordnung von Dokumenten bzw. Webseiten erfolgt, aber nicht betrachtet wird, ob diese Seiten überhaupt von lokaler Bedeutung sind.

Unserer Arbeit am ähnlichsten ist [Gravano *et al.*, 2003]. Dort werden Anfragen an Internetsuchmaschinen mit dem Ziel untersucht, sie lokalen oder globalen Informationsbedürfnissen zuzuordnen. Dazu kommen verschiedene Klassifizierungs-Mechanismen zum Einsatz, die auf Features angewendet werden, die aus Suchergebnissen von Google zu einer Anfrage ermittelt werden. Die verwendeten Features werden aus Häufigkeiten und Verteilungen von Toponymkandidaten im Text gebildet. Zudem wird ein C4.5 Klassifikator eingesetzt, um Webseiten in lokal oder global zu unterscheiden. Die Trainingsdaten bestehen aus 140 manuell klassifizierten Websites des Yahoo! directory, wobei Seiten, die unter Regional eingeordnet sind, als lokal bezeichnet werden und solche in allgemeinen Kategorien als global. Eine genauere Betrachtung oder Hinterfragung der *Locality* oder der Gründe für die Lokalität von Seiten findet jedoch nicht statt.

3 Merkmale von lokalen Webseiten

Im Folgenden werden Ansätze vorgestellt, mit denen die *Locality* von Webseiten automatisch bestimmt werden soll. Dabei werden die Besonderheiten der deutschen Sprache und der hierarchischen Gliederung Deutschlands explizit berücksichtigt. Unsere These ist, dass für eine Bestimmung des Lokalitätsgrades einer Seite verschiedene Merkmale, die sich aus einer Analyse des Seiteninhalts ergeben, als Indizien herangezogen und nicht ein einzelnes Kriterium verwendet werden sollte.

	lokal	global	Stadt	BL	Land
Seiten	1756	1040	893	539	324
Anteil Toponyme	2,38%	0,75%	2,74%	2,34%	1,46%
mind. 1 Toponym	85,1%	70,7%	92,7%	76,6%	78,1%
Ø TelNr.	1,86	0,50	2,18	1,93	0,86
mind. 1 TelNr.	39,9%	24,0%	47,4%	32,8%	31,2%

Tabelle 1: Merkmale lokaler Webseiten

3.1 Locality

Eine nicht selten anzutreffende Sichtweise von *Locality* ist, dass eine Seite umso „lokaler" ist, je enger die geographische Ausdehnung oder der geographische Fokus ist. Wird *Locality* als Maß für den Grad der Lokalität eines Dokumentes verstanden, wird deutlich, dass die Bestimmung von *Locality* kontextabhängig ist und damit auch nicht direkt mit der Größe eines geographischen Gebietes verbunden ist, sondern dass dieses im jeweiligen Kontext enthalten sein kann. Während globale Information keinen geographischen Bezugspunkt aufweist – oder dieser höchstens implizit durch die Sprache eines Dokumentes gegeben sein kann –, ist eine lokale Information stets für einen bestimmten geographischen Bereich von Interesse. So ist eine Seite, die das deutsche Autobahnnetz beschreibt, von genauso lokaler Bedeutung für das gesamte Bundesgebiet, wie ein Gasthaus von lokaler Bedeutung für ein sehr eingegrenztes Gebiet ist. Der Schluss, dass eine weite geographische Ausdehnung der *Locality* widerspricht, ist also bei diesem Verständnis von Lokalität nicht richtig.

Abschließend sei darauf hingewiesen, dass die Vielzahl möglicher Kontexte, in denen ein Dokument betrachtet werden kann, eine eindeutige Entscheidung über den Grad der Lokalität unmöglich macht. Als Beispiel sei eine Webseite einer Pizzeria genannt, die neben der Kontaktadresse mit Öffnungszeiten auch das Rezept der Hauspizza beinhaltet; aus Sicht eines Restaurantbesuchers ist dies eine lokale Seite, wohingegen ein Hobbykoch das Pizzarezept als globale Information gesucht haben könnte. In der Praxis treten darüber hinaus weitere, schwieriger zu bewertende Fälle auf.

3.2 Testkollektion

Den Untersuchungen lag eine selbst erstellte Testkollektion lokaler und nicht-lokaler Webseiten zu Grunde. Dafür wurden verschiedene Teiläste mit den Schwerpunkten Bamberg und Bayern des deutschen Regional Verzeichnisses des Open Directory[1] gecrawled und diese Webseiten danach manuell klassifiziert. Um der Problematik der geographischen Ausdehnung und der damit verbundenen unterschiedlichen Kontexte Rechnung zu tragen, wurde jedes Dokument als lokal für *Stadt*, *Bundesland* oder *Deutschland* oder als nicht-lokales Dokument bewertet. Seiten, bei denen keine plausible Einordnung möglich war, wurden verworfen. Für manche Untersuchungen wurden die 3 lokalen Klassen zu einer einzigen zusammengefasst. Insgesamt umfasst die Testkollektion ca. 3000

[1] http://dmoz.org/World/Deutsch/Regional/Europa/Deutschland/

	$\alpha = 0$			$\alpha = 0,25$			$\alpha = 0,5$			$\alpha = 0,75$			$\alpha = 1$		
	Rec.	Prec.	F	Rec.	Prec.	F	Rec.	Prec.	F	Rec.	Prec.	F	Rec.	Prec.	F
Locality	46,6	70,9	56,3	60,7	78,4	68,4	70,8	**84,1**	76,9	**74,5**	83,2	**78,6**	**74,5**	82,5	78,3
No Locality	65,0	40,0	49,5	69,3	49,1	57,5	**75,5**	58,6	66,0	72,4	**60,8**	**66,1**	71,2	60,4	65,4
Accuracy	53,2			63,8			72,5			**73,8**			73,3		

Tabelle 2: Klassifizierung anhand absoluter Termhäufigkeiten

Webseiten. Für die Untersuchungen thematischer Zusammenhänge wurden nochmals ca. 3200 Webseiten manuell den Themen *Tourismus-*, *Behörden-*, *Finanz- und Wirtschaftsförderungs-* und *Gewerbeseiten*, sowie Webauftritte *lokaler Tageszeitungen* zugeordnet, wobei hier die geographische Ausdehnung nicht gesondert berücksichtigt wurde.

3.3 Termanalyse

Zunächst wurden die Termhäufigkeiten aller Dokumente mit dem Ziel untersucht, charakteristische Ausdrücke als Entscheidungshilfe für die Bestimmung von *Locality* zu gewinnen. Die Ergebnisse der Betrachtung zeigen, in welchen Domänen lokale Seiten gehäuft auftreten und lassen erkennen, dass eine Differenzierung nach geographischer Ausdehnung und die Berücksichtigung einer thematischen Klassifikation durchaus lohnenswert sein können. Als Terme wurden stoppwortbereinigte Wortstämme betrachtet.

Die Auswertung hat gezeigt, dass es notwendig ist, sowohl relative als auch absolute Unterschiede zwischen den Vorkommenshäufigkeiten lokaler und nicht-lokaler Seiten zu berücksichtigen. Beispielsweise kommt der Begriff *Gemeinde* in 21,5% der Stadt-lokalen vor und in 2,5% der nicht-lokalen Seiten, der Begriff *Heiligenstadt* hingegen in 6,18% der Stadt-lokalen und 0,1% der nicht-lokalen. Im Folgenden werden deshalb getrennt voneinander die absoluten Unterschiede (19% im ersten Fall) und die relativen Abweichungen der lokalen von den nicht-lokalen Seiten betrachtet (6180% im zweiten Fall, also ein gut 60mal häufigeres Vorkommen des Terms in lokalen Seiten), was im Weiteren als relativer Unterschied bezeichnet wird.

Auf lokalen Stadtseiten finden sich neben Orts- und Gemeindenamen auch häufig Regionennamen sowie Regierungsbezirke. Viele Begriffe dieser Klasse können darüber hinaus dem Tourismus oder der Verwaltungsebene zugeordnet werden, wie *Sehenswürdigkeit*, *Hotel*, *Restaurant* oder *Rathaus*, *Bürgermeister* und *Gemeinderat*. Auf Bundesland-Ebene sind diese beiden Themenbereiche ebenfalls vertreten, allerdings durch andere Begriffe wie *Bayern*, *Veranstaltung*, *Mittelfranken*, *Oberfranken* etc.

Die oberste betrachtete Ebene *Deutschland* zeichnet sich unter anderem durch Terme verschiedener deutschlandweit operierender Organisationen aus, wie dem Deutschen Wetterdienst (*DWD*), dem Fahrradclub *ADFC* oder dem deutschen Sportbund. Die gefundenen Toponyme umfassen neben Stadtnamen und Bundesländern auch Gewässer und Flüsse wie die *Nordsee* und die *Donau*. Hoch gewichtet finden sich viele meteorologische Termini, die auf Wetterportale in der Testkollektion zurückzuführen sind. Dabei wurden hauptsächlich Haupteinstiegsseiten betrachtet, die in diese Klasse eingeordnet wurden. Erst auf den Unterseiten wird die geographische Ausdehnung auf Regionen und Städte eingeschränkt.

Die Termanalysen ergaben, dass bei der Berücksichtigung von geographischen Konzepten neben Orts- und Gemeindenamen auch Bezeichnungen berücksichtigt werden sollten, die sowohl die administrative Ebene als auch unpräzise Regionen, wie beispielsweise die *Fränkische Schweiz* und Gewässernamen, umfassen.

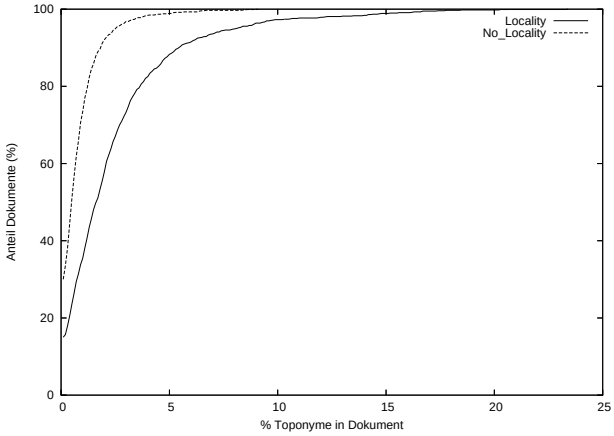


Abbildung 1: Approximative Verteilungsfunktion von Toponymen auf Seiten mit Locality

3.4 Inhaltsanalyse

Für die Gewinnung weiterer Merkmale wurden aus den Webseiten Telefonnummern und Toponyme, zu denen Orts- und Gemeindenamen, Bundesländer sowie Postleitzahlen gehören, extrahiert. Gefundene Toponymkandidaten wurden dabei als Toponyme betrachtet und mögliche Mehrdeutigkeiten mit gleichlautenden nichtgeographischen Begriffen vernachlässigt. Nur besonders häufig auftretende Homonyme wurden mit Hilfe einer manuell erstellen Liste entfernt.

Als erstes Merkmal wurde der Anteil der Toponyme am Seiteninhalt betrachtet. Aus Tabelle 1 wird deutlich, dass ein Dokument nicht zwangsläufig eine lokale Bedeutung hat, wenn in dessen Inhalt geographische Begriffe auftreten: auch 70% der nicht-lokalen Seiten enthalten mindestens ein Toponym. Dennoch treten erwartungsgemäß Toponyme auf lokalen Seiten häufiger auf. Durchschnittlich sind 2,38% der Terme eines lokalen Dokuments Toponyme, gegenüber 0,75% bei nicht-lokalen Seiten. Abbildung 1 zeigt die Verteilungsfunktion der Toponymanteile beider Klassen. Der Toponymanteil ist demnach ein Indiz für eine lokale Webseite.

Bei Betrachtung der lokalen Klassen unterschieden nach deren geographischer Ausdehnung ist in Abbildung 2 erneut zu erkennen, dass jede der Klassen einen höheren Toponymanteil als nicht-lokale Dokumente aufweist. Allerdings wird die Abgrenzung der Klassen beispielsweise durch den sehr ähnlichen Verlauf von lokalen Stadt- und Bundeslandsseiten erschwert.

Als weiteres Merkmal wurde das Vorkommen von Telefonnummern betrachtet. Während etwa nur 24% der nichtlokalen Seiten eine solche enthalten, sind dies bei lokalen etwa 40%, bei Stadt-lokalen Seiten sogar ca. 47%. Mehr als 90% der nicht-lokalen Seiten enthalten nicht mehr als eine Telefonnummer, gegenüber 73% bei lokalen Seiten. Es erscheint plausibel, diese Merkmale als Indiz für Lokalität zu

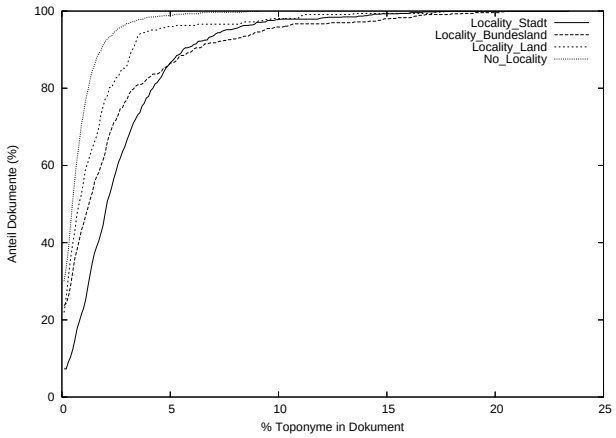


Abbildung 2: Approximative Verteilungsfunktion von Toponymen auf Seiten mit Locality (mit geographischer Ausdehnung)

verwenden, wenn sie auch nicht sehr stark differenzieren.

Daneben wurden lokale Seiten auch thematisch klassifiziert und anhand derselben Kriterien untersucht. Dabei werden signifikante Unterschiede zwischen den Themenbereichen deutlich. So weisen Tourismusseiten mit 2,99% erwartungsgemäß den größten Toponymanteil auf, wohingegen Finanz- und Wirtschaftsförderungsseiten mit durchschnittlich 0,61% Toponymen sogar unter den Werten nicht-lokaler Seiten (0,75%) liegen.

Es erscheint daher lohnenswert, die Merkmale einer Webseite in Abhängigkeit von einer thematischen Klassifikation zu bewerten. So ist beispielsweise auch zu erkennen, dass Telefonnummern auf Finanz- und Wirtschaftsförderungsseiten ein wichtiges Indiz für *Locality* darstellen, da auf 86,5% dieser Seiten mindestens eine auftrat. Dagegen beinhalten nur 13,2% der betrachteten Seiten lokaler Tageszeitungen eine Telefonnummer.

Die Zahlen zeigen, dass keines dieser Merkmale allein dazu geeignet ist, *Locality* festzustellen. Vielmehr stützen sie die These, dass dafür eine Vielzahl von Indizien herangezogen werden muss.

3.5 Klassifikation nach Wortvorkommen

Es soll nun versucht werden, die Ergebnisse der Termanalyse aus Abschnitt 3.3 dazu zu verwenden, Webseiten in lokal und nicht-lokal einzuordnen. Dabei sollen zwei Merkmale betrachtet werden: zum einen der reine Anteil der lokalen Terme in Relation zu der Gesamtwortanzahl eines Dokuments, zum anderen der gewichtete Anteil lokaler Terme, bei dem solche Terme höher gewichtet werden, die nach der Analyse aus Abschnitt 3.3 häufiger in lokalen Seiten vorkommen. Um dabei die Dominanz der häufigsten Terme abzuschwächen, wurden die Gewichte rangerhaltend mittels des Logarithmus zur Basis 10 angepasst. Mit Hilfe des Parameters α werden diese beiden Merkmale relativ zueinander gewichtet.

Dafür werden folgende Definitionen verwendet:

n_D	Anzahl der Wörter in Dokument D
n_{lD}	Anzahl der lokalen Terme in Dokument D
w_{ck}	Gewicht des Terms k in Klasse c
tf_{dk}	Vorkommenshäufigkeit des Terms k in Dokument d
α	Gewichtungsfaktor [0;1]

Die w_{ck} werden den Termauswertungen entnommen. Der *score*, der den Grad der *Locality* für eine der betrachteten Klasse angibt, wird mit Hilfe der untenstehenden Formel berechnet.

$$score = \alpha \cdot \frac{n_{lD}}{n_D} + (1-\alpha) \cdot \frac{\sum_{k \in N_{lD}} \log_{10}(w_{ck}) \cdot tf_{dk}}{top(k)}$$

$$top(k) = \sum \text{k-größten Gewichte } w_{ck}$$

Nachfolgend werden die Ergebnisse der Klassifizierung anhand von Wortvorkommen vorgestellt, die durch die Berechnung von *score*-Werten erreicht wurden. Die Auswertungen wurden sowohl mit den relativen als auch den absoluten Unterschieden der Wortvorkommen zwischen lokalen und nicht-lokalen Seiten durchgeführt. Zur Messung des Einflusses der beiden Merkmale wurde der Parameter α variiert, um die Gewichtung der zwei Merkmale auf der Testkollektion zu untersuchen. Dabei wurden für die geographische Dimension weitere 437 und für die thematischen Klassen 758 Dokumente als Testmenge klassifiziert.

Da eine isolierte Betrachtung von *Precision* bzw. *Recall* nicht aussagekräftig genug erscheint, wird in [van Rijsbergen, 1979] das F_1*-Measure* vorgestellt. Das F_1*-Measure* beschreibt das harmonische Mittel zwischen *Recall* und *Precision*.

$$F - Measure = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

Daneben gibt die *Accuracy* den Anteil korrekter Klassifizierungsvorgänge an.

Zunächst werden die absoluten Wortvorkommenshäufigkeiten bei variierendem α betrachtet, und es wird nur zwischen lokalen und nicht-lokalen Seiten unterschieden. Die Ergebnisse sind in Tabelle 2 dargestellt. Es lässt sich erkennen, dass die Ergebnisse mit größerem α besser werden.

Das beste Ergebnis nach dem F-Maß lässt sich bei einem α von 0,75 erreichen. Dabei werden 74,5% der relevanten lokalen Dokumente aus der Testmenge erkannt, und nur 17,0% der dieser Klasse zugeordneten Dokumente falsch eingeordnet.

Dieselbe Betrachtung wurde noch einmal anhand der relativen Wortvorkommen durchgeführt. Die Ergebnisse davon sind in Tabelle 3 zu sehen.

Schließlich wurde zusätzlich die Einteilung in Stadt-, Bundesland- und Deutschland-lokal berücksichtigt. In Tabelle 4 sind die entsprechenden Ergebnisse zu finden. Während bei Stadt-lokalen Seiten eine Einbeziehung charakteristischer Wörter eine leichte Verbesserung bewirkt, trifft dies auf die anderen *Locality*-Klassen nicht zu.

Aus diesen Betrachtungen scheint ersichtlich, dass der Anteil Toponyme in einem Dokument ein wichtigeres Kriterium als das Vorkommen charakteristischer Terme ist. Gleichwohl bringt die Berücksichtigung dieser Terme einen leichten Anstieg der Klassifikationsgenauigkeit. Weitere Verbesserungen sind zu erwarten, wenn diese Liste (manuell) optimiert wird. Beispielsweise sind dort noch alle Toponyme enthalten, die zum Teil sehr ortsspezifisch sind. Eine Unterscheidung von *Locality* in Klassen unterschiedlicher geographischer Ausdehnung scheint hier zunächst weniger erfolgversprechend. Untersuchungen, für jede Klasse eigene Wortlisten zu erstellen, fanden noch nicht statt und erscheinen ohne eine Verbesserung des Verfahrens hinsichtlich einer Verbesserung der Ergebnisqualität auch fraglich.

	$\alpha = 0$			$\alpha = 0,25$			$\alpha = 0,5$			$\alpha = 0,75$			$\alpha = 1$		
	Rec.	Prec.	F	Rec.	Prec.	F	Rec.	Prec.	F	Rec.	Prec.	F	Rec.	Prec.	F
Locality	55,7	84,7	67,2	64,8	**85,0**	73,5	66,8	84,7	74,7	**68,5**	**85,0**	**75,8**	66,8	84,7	74,7
No Locality	**81,6**	50,2	62,2	79,1	55,1	65,0	77,9	56,2	65,3	77,9	**57,5**	**66,2**	77,9	56,2	65,3
Accuracy	64,9			69,9			70,7			**71,8**			70,7		

Tabelle 3: Klassifizierung anhand relativer Unterschiede der Termhäufigkeiten

	$\alpha = 0$			$\alpha = 0,25$			$\alpha = 0,5$			$\alpha = 0,75$			$\alpha = 1$		
	Rec.	Prec.	F	Rec.	Prec.	F	Rec.	Prec.	F	Rec.	Prec.	F	Rec.	Prec.	F
Stadt	38,8	61,0	47,4	58,1	83,3	68,5	72,1	**92,1**	**80,9**	72,1	**92,1**	**80,9**	**73,6**	89,6	**80,9**
Bundesland	77,7	42,4	54,9	87,2	50,6	64,1	88,3	58,0	70,0	**89,4**	60,9	72,4	87,2	**63,1**	**73,2**
Land	38,7	38,2	38,4	44,0	54,1	48,5	44,0	55,9	49,3	46,5	58,3	51,9	**46,7**	**63,6**	**53,9**
No Locality	43,6	54,2	48,4	55,8	61,5	58,5	63,2	65,2	64,2	66,3	**66,7**	66,5	**69,3**	66,5	**67,9**
Accuracy	48,4			61,0			67,7			69,4			**70,5**		

Tabelle 4: Klassifizierung anhand absoluter Unterschiede der Termhäufigkeiten, nach geographischer Ausdehnung

Für eine Berücksichtigung unterschiedlicher thematischer Kontexte wurden die Seiten der Testkollektion fünf verschiedenen thematischen Klassen zugeordnet. Anschließend wurden dieselben Untersuchungen mit relativen und absoluten Wortvorkommen durchgeführt.

Die Ergebnisse zeigten, dass zwischen den einzelnen thematischen Klassen große Unterschiede in der Anwendbarkeit dieses Verfahrens bestehen. Insbesondere Behördenseiten weichen hier von den Charakteristika anderer Themenklassen ab. Für die weitere Optimierung der Erkennungsrate lokaler Seiten kann es demnach durchaus hilfreich sein, zunächst den thematischen Kontext eines Dokumentes zu erfassen, um anschließend dessen Besonderheiten bei den Ausprägungen der betrachteten Merkmale zu berücksichten.

3.6 Berücksichtigung weiterer Merkmale

Man kann nun versuchen, die gefundenen Indizien für *Locality* aus Abschnitt 3.4 zusätzlich zu den Gewichtungen der Wortvorkommen für die lokalen Klassen Abschnitt 3.5 zu verwenden.

Die Untersuchung beschränkt sich momentan auf die binäre Entscheidung, ob ein Dokument in die Klasse der lokalen oder nicht-lokalen Dokumente gehört. Dabei wird die geographische Testkollektion und deren Vokabular verwendet.

Es werden Auswertungen durchgeführt, die bei der *score*-Berechnung den Toponymanteil und die Anzahl der Erwähnungen von Telefonnumern berücksichtigen. Dabei wird zuerst jeweils eins der drei Indizien verwendet und mit der Wortvorkommenmethode kombiniert, wobei der Einfluss eines Indizes über einen Parameter β variiert.

$$score_{combined} \quad = \quad \beta \cdot score_{\alpha=0,75} + (1-\beta) w_{Indiz}$$

Aus Abschnitt 3.4 stehen für die verschiedenen untersuchten Indizien Verteilungsfunktionen zur Verfügung. Es soll möglich sein, die Gewichtungen für eine Ausprägung eines Indizes dafür je nach Stärke des Indizes zu variieren. Ist bekannt, dass 30% der lokalen Webseiten drei Telefonnummern aufweisen und dieser Anteil bei nicht-lokalen nur bei 5% liegt, soll diese Verteilung mit in die Gewichtung eingehen.

Für die Bestimmung dieser Gewichte für die Indizien wurden verschiedene Ansätze evaluiert. Anfangs wurde als Gewichtung die relative Dokumentenhäufigkeit einer Klasse verwendet, bei der das Indiz bei einer bestimmten Ausprägung vorkommt. Die Ergebnisse für die Klasse der lokalen Seiten waren nach dem F-Maß schlechter als bei der reinen Auswertung nach den Wortvorkommen. Einzig bei Gewichtung der Wortvorkommen zu 90% und zu 10% des Toponymanteils lassen sich die Basisergebnisse einstellen. Bei den nicht-lokalen Dokumenten lässt sich eine leichte Steigerung des F-Maßes erreichen. Aufgrund dieser Ergebnisse insbesondere für die Klasse der lokalen Dokumente wurde nach anderen Gewichtungsmöglichkeiten für Indizien gesucht, die auf *Locality* hinweisen.

Im nächsten Schritt wurde die approximative Verteilungsfunktion der verschiedenen Indizien in den betrachteten Klassen verwendet. Als Gewichtung kam folgende Berechnung zum Einsatz:

$$w(X = x) = 1 - \hat{F}^*_X(x)$$

Die Gewichte eines Indizes in einem Dokument mit der Ausprägung x ergeben sich aus der Differenz von 1 mit dem Wert der approximativen Verteilungsfunktion an der Stelle $X = x$. Die Gewichtung eines Indizes stellt also den Anteil der Dokumente dar, die das Indiz x-mal oder mehr aufweisen.

Als Basisdaten kommen die Ergebnisse der Klassifizierung der Seiten der geographischen Taxonomie zum Einsatz, die bei Betrachtung der absoluten Differenzen der Vorkommenshäufigkeiten in Abschnitt 3.3 errechnet wurden. Als Referenzwerte werden die besten Ergebnisse, die bei Betrachtung des *score* erzielt worden sind, verwendet. Die Auswertungen sind in den Tabellen 5 und 6 dargestellt. Zuerst wird der Einfluss des Toponymanteils auf das Ergebnis der *Locality*-Berechnung untersucht.

Die nächste Auswertung befasst sich mit dem Einfluss von Telefonnummern auf einer Seite und kombiniert diesen mit den Ergebnissen der Wortvorkommenmethode. Auch hier ist eine leichte Verbesserung der Erkennungsraten festzustellen.

In einer letzten Untersuchung wird eine Kombination der drei Indizien mit der Wortvorkommenmethode durchgeführt. Dabei wurde die Gewichtung zwischen 0,5 und 0,1 variiert. Der Anteil der Indizien wurde auf jedes Einzelindiz zu gleichen Teilen verteilt. Die hierbei erzielten Ergebnisse sind in Tabelle 7 wiedergegeben. Es ist zu erkennen, dass die F-Maßwerte der lokalen Klasse bei jeder betrachteten Gewichtung über denen der Basismethode liegen.

3.7 Klassifikation mit lernenden Verfahren

Alternativ zur oben vorgestellten Methode fand zusätzlich eine Klassifikation nach lokalen Webseiten mit lernenden

	Toponyme 0,5			Toponyme 0,25			Toponyme 0,1		
	$\alpha = 0,75$			$\alpha = 0,75$			$\alpha = 0,75$		
	Rec.	Prec.	F-Maß	Rec.	Prec.	F-Maß	Rec.	Prec.	F-Maß
Locality	**85,9**	73,8	79,4	80,2	80,5	**80,3**	76,2	**82,9**	79,4
No Locality	44,2	63,2	52,0	64,4	**64,0**	64,2	**71,2**	62,0	**66,3**
Accuracy	71,2			**74,4**			**74,4**		

Tabelle 5: Kombination von Toponym- und Wortvorkommen

	Tel.Nr. 0,5			Tel.Nr. 0,25			Tel.Nr. 0,1		
	$\alpha = 0,75$			$\alpha = 0,75$			$\alpha = 0,75$		
	Rec.	Prec.	F-Maß	Rec.	Prec.	F-Maß	Rec.	Prec.	F-Maß
Locality	**82,6**	79,4	**80,9**	78,5	82,4	80,5	75,8	**83,1**	79,3
No Locality	60,7	**65,6**	63,1	69,3	63,8	**66,5**	**71,8**	61,9	**66,5**
Accuracy	74,8			**75,3**			74,4		

Tabelle 6: Kombination von Telefonnummern- und Wortvorkommen

Verfahren statt. Dabei wurden ein Naive Bayes Klassifikator (vgl. [Rish, 2001]) und Support Vector Machines (vgl. [Boser *et al.*, 1992]) verwendet. Beide Verfahren eignen sich für die Klassifikation von Texten (vgl. [Lewis and Ringuette, 1994], [Joachims, 1998], [Yang, 1999]). Für die Klassifizierung nach dem Bayes Theorem wird die Java - Bibliothek Classifier4J[2] verwendet. Beim Training mit dem Bayes Klassifikator ist anzumerken, dass die verwendete Bibliothek keine Mehrklassenklassifizierung unterstützt, d. h. beim Lernvorgang ist darauf zu achten, zu der positiven Trainingsdatenmenge auch negative Trainingsbeispiele anzugeben. Es muss explizit beschrieben werden, was das Charakteristische an Dokumenten mit *Locality* ist und wie die Komplementärmenge der nicht-lokalen Webseiten aussieht. Als Support Vector Machine kam die Bibliothek LIBSVM[3] zum Einsatz (vgl. [Chang and Lin, 2001]). Zur Klassifizierung wurden ein *linearer* und ein *radial basis function (rbf)* Kernel verwendet. Die Mehrklassenklassifizierung greift auf die *One-Against-One* Methode zurück (vgl. [Hsu and Lin, 2002] und [Chin, 1999]). Dazu werden bei der Klassifizierung von k Klassen $k(k-1)/2$ binäre Probleme gelöst. Die Entscheidung, in welche Klasse ein Dokument gehört, wird über die *Max Wins* Strategie bestimmt. Als Ergebnisklasse wird diejenige gewählt, für die bei den binären Entscheidungen am häufigsten eine Klassenzugehörigkeit bestimmt wurde.

Zunächst wird die Klassifizierung nach *lokalen* und *nicht-lokalen* Webseiten betrachtet. Die Ergebnisse sind in Tabelle 8 dargestellt. Die Support Vector Machine mit *rbf* Kernel erreicht bei Betrachtung des F-Maßes beider Klassen die besten Werte.

Tabelle 9 stellt die Ergebnisse der Klassifizierung nach den drei untersuchten geographischen Ausdehnungen dar. Auch hier ist die Support Vector Machine am erfolgreichsten. Die Klassifizierung von Dokumenten funktioniert bei der Klasse der lokalen Dokumente auf Stadtebene mit beiden Kerneln sehr gut.

Bei der Klasse der lokalen Seiten auf Bundeslandebene sinken die Ergebnisse bei allen drei Verfahren, auf Landes-Ebene verschlechtern sich die Klassifizierungsergebnisse sogar signifikant.

Betrachtet man nur die Rate der richtig klassifizierten Dokumente bezogen auf die Gesamtzahl der relevanten Dokumente dieser Klasse ergibt sich der höchste Recall von 87,8% bei dem Naive Bayes Klassifikator, der allerdings mit einer Precision von 55,0% am schlechtesten abschneidet, d. h. es werden zuviele Dokumente anderer Klassen dieser Klasse zugeordnet. Begründet werden kann dies durch die Bestimmung der Klasse beim verwendeten Naive Bayes Klassifikator. Ein Dokument wird einer (lokalen) Positivklasse, erst dann zugeordnet, wenn die errechnete Wahrscheinlichkeit der Klassenzugehörigkeit größer gleich 80% beträgt. Deshalb fallen Dokumente, die eigentlich zu einer Klasse gehören mitunter in die Klasse der nicht-lokalen Dokumente (Negativklasse), da die Einzelwahrscheinlichkeit für eine Zuordnung nicht ausreicht. Die Support Vector Machine betrachtet alle Klassen und errechnet die Wahrscheinlichkeiten so, dass diese addiert 100% ergeben. Als Ergebnisklasse wird die mit der höchsten Wahrscheinlichkeit gewählt.

3.8 Berücksichtigung thematischer Kontexte

Darüber hinaus wurden die lernenden Verfahren auch verwendet, um Seiten direkt thematisch lokalen Seiten zuzuordnen. Dabei wurde jede Seite entweder als lokal und einem bestimmten thematischen Gebiet zugeordnet oder als nicht-lokal eingestuft. Die Klasse der nicht-lokalen Dokumente enthält demnach Webseiten aus allen Themenbereichen, die nicht als von lokalem Interesse eingestuft wurden. Die Ergebnisse der Untersuchung zeigt Tabelle 10. Auffällig ist das schlechte Abschneiden der Gewerbe-Klasse, für die offenbar andere Merkmale zum Tragen kommen als in den anderen Klassen.

4 Evaluierung

Abschließend soll eine zusammenfassende und kritische Betrachtung der erzielten Ergebnisse sowohl bei den Textanalysen als auch den Termanalysen stattfinden. Außerdem werden die Ergebnisse der automatischen Klassifizierung nach den beiden lernenden Verfahren – Naive Bayes Klassifikator und Support Vector Machine – dem Ansatz gegenübergestellt, *Locality* anhand von einzelnen Merkmalen zu bestimmen.

4.1 Indizien für Locality

Die Textanalysen aus Abschnitt 3.4 haben ergeben, dass sich lokale Webseiten in ihrem Seiteninhalt von nicht-lokalen unterscheiden. Dies trifft sowohl auf verschiedene untersuchte Indizien zu als auch für das verwendete Vokabular. Je nach betrachtetem Indiz – Telefonnummern oder Toponymanteil – sind die Unterschiede bei den Hinweisen auf *Locality* unterschiedlich stark ausgeprägt. Bei Betrachtung von lokalen und nicht-lokalen Dokumenten kann

[2] http://classifier4j.sourceforge.net/

[3] http://www.csie.ntu.edu.tw/ cjlin/libsvm/

	Kombination 0,5			Kombination 0,25			Kombination 0,1		
	$\alpha = 0,75$			$\alpha = 0,75$			$\alpha = 0,75$		
	Rec.	Prec.	F-Maß	Rec.	Prec.	F-Maß	Rec.	Prec.	F-Maß
Locality	**83,2**	78,5	**80,8**	77,9	82,6	80,1	75,8	**83,1**	79,3
No Locality	58,3	**65,5**	61,7	69,9	63,3	**66,5**	**71,8**	61,9	**66,5**
Accuracy	74,4			**75,1**			74,4		

Tabelle 7: Klassifizierungsergebnisse der Kombination lokaler Indizien

	Naive Bayes			SVM (linear kernel)			SVM (rbf kernel)		
	Recall	Precision	F-Maß	Recall	Precision	F-Maß	Recall	Precision	F-Maß
Locality	77,9	**82,6**	80,2	84,7	81,5	83,1	**86,9**	81,5	**84,1**
No Locality	**70,5**	64,0	67,1	65,4	70,3	67,8	63,8	**72,7**	**68,0**
Accuracy	75,3			77,8			**78,7**		

Tabelle 8: Klassifizierungsergebnisse durch lernende Verfahren

vor allem der Toponymanteil ein guter Hinweis auf lokale Webseiten sein. Der Schluss auf die Tatsache, dass lokale Webseiten durchschnittlich mehr Toponyme aufweisen als nicht-lokale Seiten, ist somit möglich.

Das Kriterium der Telefonnummernanzahl auf einer Webseite ist nicht so trennscharf wie der Toponymanteil. Ersichtlich wird das durch den hohen Anteil der Webseiten beider Klassen, die keine Telefonnummer aufweisen.

Unterteilt man die lokale Klasse nach der geographischen Ausdehnung, sind auch Unterschiede bei der Verteilung der verschiedenen Indizien festzustellen. Allerdings fällt hier die Abgrenzung der einzelnen Klassen mitunter schwer, da die Verteilung der verschiedenen Klassen nicht immer trennscharf ist.

Es erscheint schwer möglich, für alle untersuchten Klassen und Taxonomien Heuristiken zu formulieren, anhand derer man eine Klassifizierung nach *Locality* vornehmen bzw. eine auf anderen Kriterien beruhende verbessern kann. Bei Beschränkung auf zwei Klassen erscheint die Formulierung einfacher Heuristiken nach dem Schema noch relativ einfach realisierbar zu sein.

4.2 Klassifizierung durch lernende Verfahren

Bei den lernenden Verfahren erzielte die Support Vector Machine mit rbf Kernel insgesamt die besten Ergebnisse, die auch besser waren als diejenigen, die das Verfahren anhand von Einzelmerkmalen erzielen konnte.

Aus diesem Grund wurde zuletzt noch versucht, diese Merkmale auch durch die SVM berücksichtigen zu lassen. Dazu wurden diese Merkmale den Features für die SVM hinzugefügt. In den ersten Versuchen ließen sich hier jedoch nur bei Mitberücksichtigung des Toponymanteils geringe Verbesserungen erzielen, wodurch die Accuracy um 0,3 erhöht werden konnte. Weitere Optimierungen könnten hier in der Zukunft die Klassifikationsgenauigkeit weiter erhöhen.

4.3 Geschwindigkeit

Bei der reinen Bewertung der Klassifkationsqualität scheint die SVM das geeignetste Verfahren zur Bestimmung von *Locality* zu sein. Dass eine Betrachtung von Merkmalen von Lokalität dennoch eine Daseinsberechtigung hat, zeigt sich spätestens bei einer Berücksichtigung der Verarbeitungsgeschwindigkeit. Die Bestimmung von *Locality* allein anhand von Wortvorkommen war bei den Untersuchungen um den Faktor 4-5 schneller als durch den Einsatz von SVMs, und etwa 11mal schneller als mit Hilfe des Naive Bayes Klassifikators.

5 Zusammenfassung und Ausblick

In dieser Arbeit wurde gezeigt, dass lokale Webseiten bestimmte Merkmale aufweisen, die auch dazu verwendet werden können, über die *Locality* einer Webseite zu entscheiden.

Als wesentliches Merkmal wurde dabei das Vorkommen von Toponymen im Seiteninhalt identifiziert und die Annahme bestätigt, dass viele Toponyme auf eine Seite von lokaler Bedeutung hinweisen. Gleichwohl konnte auch gezeigt werden, dass das Vorkommen von Toponymen allein nicht ausreicht, um eine Webseite als lokal ansehen zu können.

Daneben scheint auch das Vokabular ein geeignetes Kriterium für Lokalität zu sein. Lernende Verfahren konnten hier erwartungsgemäß die besten Ergebnisse aufweisen, wobei ein einfacherer Ansatz ebenfalls zu ordentlichen Ergebnissen kommt und dabei wesentlich schneller arbeiten kann.

Die Ergebnisse zeigen, dass es grundsätzlich möglich ist, Webseiten einerseits nach dem Kriterium der *Locality* und andererseits nach der geographischen Ausdehnung einer Seite – im Sinne des *Geographical Scopes* – zu klassifizieren. Auch unterscheiden sich Webseiten aus unterschiedlichen Themenbereichen in ihren Merkmalsausprägungen, was sich in mehrstufigen Klassifikationsvorgängen ausnutzen ließe.

Weitere Verbesserungen der Klassifikationsgenauigkeit bei Beibehaltung einer hohen Verarbeitungsgeschwindigkeit, die für einen Indexierungsprozess einer Web-Suchmaschine unabdingbar ist, sind das Ziel weiterer Untersuchungen. Dabei sollen weitere Merkmale erkannt und betrachtet werden, wie z. B. die Linkstruktur zwischen lokalen Webseiten oder erkannte Adressangaben, sowie insbesondere die gewonnen Erkenntnisse aus dieser Arbeit genutzt werden, um Merkmale geeignet mit Klassifikationen zu kombinieren, um die aussagekräftigsten Indizien für bestimmte Kontexte heranziehen zu können.

Letztlich soll als Ergebnis der Bestimmung von Lokalität ein nicht-binärer Wert stehen, der den Grad der Lokalität oder die Wahrscheinlichkeit für Lokalität angibt. Ein solcher Wert kann in weiteren Prozessschritten der Anfragebearbeitung genutzt werden: einerseits könnten von vornherein nur solche Dokumente überhaupt für eine ortsbezogene Suche betrachtet werden, die ein Mindestmaß an Lokalität aufweisen, andererseits ist auch eine Integration in ein Rankingverfahren ohne weiteres Möglich. Kombinierte thematische und geographische Rankings, wie sie beispielsweise in [Vaid *et al.*, 2005] und [Martins *et al.*, 2005] betrachtet

	Naive Bayes			SVM (linear kernel)			SVM (rbf kernel)		
	Recall	Precision	F-Maß	Recall	Precision	F-Maß	Recall	Precision	F-Maß
Stadt	78,7	76,8	77,7	**87,7**	87,7	**87,7**	86,9	**88,3**	87,6
Bundesland	46,1	77,4	57,7	71,9	76,2	74,0	**75,3**	**77,9**	**76,6**
Land	8,57	60,0	15,0	47,1	71,7	56,9	**50,0**	**76,1**	**60,3**
No Locality	**87,8**	55,0	67,7	76,3	64,3	69,8	77,6	**65,4**	**71,0**
Accuracy	64,1			73,9			**75,3**		

Tabelle 9: Klassifizierungsergebnisse durch lernende Verfahren nach geographischen Ausdehnungen

	Naive Bayes			SVM (linear kernel)			SVM (rbf kernel)		
	Recall	Precision	F-Maß	Recall	Precision	F-Maß	Recall	Precision	F-Maß
Tourismus	72,7	62,4	67,2	71,6	78,0	74,6	**75,4**	**80,2**	**77,7**
Behörden	50,0	80,5	61,7	**65,6**	88,0	75,1	65,2	**89,8**	**75,6**
Tageszeitungen	40,2	97,1	56,9	**82,9**	**100,0**	**90,7**	81,7	**100,0**	89,9
Finanzen	44,4	**92,3**	60,0	81,5	81,5	81,5	**88,9**	82,7	**85,7**
Gewerbe	13,0	75,0	22,2	**37,0**	**77,3**	**50,0**	**37,0**	58,6	45,3
No Locality	**92,7**	45,9	61,4	87,8	51,4	64,9	86,0	**52,0**	**64,8**
Accuracy	61,2			72,6			**73,1**		

Tabelle 10: Vergleich der Klassifizierungsergebnisse bei verschiedenen thematischen Klassen

werden, lassen sich somit um einen Lokalitätsfaktor erweitern und wären damit nicht länger auf die inhaltliche Ähnlichkeit und eine geographische Nähe beschränkt.

Literatur

[Amitay *et al.*, 2004] Einat Amitay, Nadav Har'El, Ron Sivan, and Aya Soffer. Web-a-where: geotagging web content. In *SIGIR '04*, pages 273–280, New York, NY, USA, 2004. ACM Press.

[Boser *et al.*, 1992] Bernhard E. Boser, Isabelle Guyon, and Vladimir Vapnik. A training algorithm for optimal margin classifiers. In *Computational Learing Theory*, pages 144–152, 1992.

[Buyukkokten *et al.*, 1999] Orkut Buyukkokten, Junghoo Cho, Hector Garcia-Molina, Luis Gravano, and Narayanan Shivakumar. Exploiting geographical location information of web pages. In *WebDB (Informal Proceedings)*, pages 91–96, 1999.

[Chang and Lin, 2001] Chih-Chung Chang and Chih-Jen Lin. *LIBSVM: a library for support vector machines*, 2001.

[Chin, 1999] K. Chin. Support vector machines applied to speech pattern classification. Master's thesis, University of Cambridge, 1999.

[Ding *et al.*, 2000] Junyan Ding, Luis Gravano, and Narayanan Shivakumar. Computing geographical scopes of web resources. In *26th Intl. Conf. on Very Large Databases, VLDB 2000*, Cairo, Egypt, September 10–14 2000.

[Gravano *et al.*, 2003] Luis Gravano, Vasileios Hatzivassiloglou, and Richard Lichtenstein. Categorizing web queries according to geographical locality. *CIKM'03*, 2003.

[Hsu and Lin, 2002] C.-W. Hsu and C.-J. Lin. A comparison of methods for multi-class support vector machines. In *IEEE Transactions on Neural Networks*, number 13, pages 415–425, 2002.

[Joachims, 1998] Thorsten Joachims. Text categorization with support vector machines: learning with many relevant features. In Claire Nédellec and Céline Rouveirol, editors, *Proceedings of ECML-98, 10th European Conf. on Machine Learning*, number 1398, pages 137–142, Chemnitz, DE, 1998. Springer, Heidelberg.

[Lewis and Ringuette, 1994] David D. Lewis and Marc Ringuette. A comparison of two learning algorithms for text categorization. In *Proceedings of SDAIR-94, 3rd Annual Symposium on Document Analysis and Information Retrieval*, pages 81–93, SIGIR 2004, 1994.

[Markowetz *et al.*, 2004] Alexander Markowetz, Thomas Brinkhoff, and Bernhard Seeger. Geographic information retrieval. *Proceedings of the 3rd Intl. Workshop on Web Dynamics, WWW2004, New York, NY, USA*, 2004.

[Martins *et al.*, 2005] Bruno Martins, Mario J. Silva, and Leonardo Andrade. Indexing and ranking in geo-ir systems. In *GIR '05: Proceedings of the 2005 workshop on Geographic information retrieval*, pages 31–34, New York, NY, USA, 2005. ACM Press.

[McCurley, 2001] Kevin S. McCurley. Geospatial mapping and navigation of the web. In *WWW '01: Proceedings of the 10th intl. conf. on World Wide Web*, pages 221–229, New York, NY, USA, 2001. ACM Press.

[Rish, 2001] Irina Rish. An empirical study of the naive bayes classifier. In *IJCAI-01 workshop on Empirical Methods in AI*, 2001.

[Sanderson and Kohler, 2004] Mark Sanderson and Janet Kohler. Analyzing geographic queries. In *Proceedings of the ACM SIGIR Workshop on Geographic Information Retrieval*, Sheffield, UK, 2004.

[Vaid *et al.*, 2005] Subodh Vaid, Christopher B. Jones, Hideo Joho, and Mark Sanderson. Spatio-textual indexing for geographical search on the web. In *SSTD*, pages 218–235, 2005.

[van Rijsbergen, 1979] C.J. van Rijsbergen. *Information Retrieval*. Butterworths, London, 2nd edition, 1979.

[Yang, 1999] Yiming Yang. An evaluation of statistical approaches to text categorization. *Information Retrieval*, 1(1/2):69–90, 1999.

[Zhang *et al.*, 2006] Jianwei Zhang, Yoshiharu Ishikawa, Sayumi Kurokawa, and Hiroyuki Kitagawa. Localrank: A prototype for ranking web pages with database considering geographical locality. *Lecture Notes in Computer Science*, Volume 3841:1209–1213, 2006.

Service-orientierte Architekturen für Information Retrieval

Sven Meyer zu Eissen and **Benno Stein**

Fakultät Medien, Mediensysteme
Bauhaus-Universität Weimar, 99421 Weimar

sven.meyer-zu-eissen@medien.uni-weimar.de
benno.stein@medien.uni-weimar.de

Abstract

Dieses Papier gibt eine Einführung in TIRA[1], einer Software-Architektur für die Erstellung maßgeschneiderter Information-Retrieval-Werkzeuge. TIRA ermöglicht Anwendern, den Verarbeitungsprozess eines gewünschten IR-Werkzeugs interaktiv als Graph zu spezifizieren: die Knoten des Graphen bezeichnen so genannte „IR-Basisdienste“, Kanten modellieren Kontroll- und Datenflüsse. TIRA bietet die Funktionalität eines Laufzeit-Containers, um die spezifizierten Verarbeitungsprozesse in einer verteilten Umgebung auszuführen.

Motivation für unsere Forschung ist u. a. die Herausforderung der Personalisierung: Es gibt eine Diskrepanz zwischen der IR-Theorie und ihren Algorithmen und der – an persönlichen Wünschen angepassten – Implementierung, Verteilung und Ausführung entsprechender Programme. Diese Kluft kann mit adäquater Softwaretechnik verkleinert werden.

1 Einleitung

Information-Retrieval (IR) ist eine Schlüsseltechnologie im Umgang mit der Informationsüberflutung, die wiederum aus ubiquitärer Verfügbarkeit von Informationen und einer schnell wachsenden Zahl an Informationsquellen und -erzeugern entsteht. Dabei steht IR nicht als universelle Lösung für ein generisches Problem – vielmehr ist IR ein Sammelbegriff für unzählige Lösungen individueller Informationsbedürfnisse. Um wirksam und nützlich zu sein, muss IR-Technologie an persönliche Fragestellungen, an persönliche Vorlieben, an persönliche Fähigkeiten und an persönliche Daten angepasst werden. Diese Forderung wird von existierender IR-Technologie erst ansatzweise erfüllt: beispielsweise werden generische Suchmaschinen bei der Suche im World Wide Web benutzt, die nicht über problemspezifisches Wissen verfügen und den Erfolg einer Suche der Kreativität des Anwenders und seiner Erfahrung und Zeit überlassen.

Aus der Sicht der Softwaretechnik steht hinter jedem IR-Werkzeug ein bestimmter IR-Prozess. Die Implementierung eines IR-Prozesses sollte nicht monolithisch geschehen, sondern dem Paradigma der Service-Komposition folgen: Ein maßgeschneidertes IR-Werkzeug für einen individuellen IR-Prozess könnte durch die Kombination von IR-Basisdiensten spezifiziert und operationalisiert werden. Für eine Architektur, die so etwas ermöglicht, wünscht man sich folgende Eigenschaften:

- *Flexibilität.* Die Spezifikation von IR-Prozessen, ihre Anpassung an geänderte Informationsbedürfnisse und ihre Evaluation kann ad-hoc geschehen.
- *Offenheit.* Die Entwicklung und Integration neuer IR-Dienste ist unterstützt.
- *Modularität.* Das Speichern und die Wiederbenutzung von Prozessen als eigene Basisdienste ist möglich.
- *Skalierbarkeit.* Mehr Rechenleistung führt zu schnellerer Ausführung.

Das vorliegende Papier beschäftigt sich mit diesen Herausforderungen und stellt entsprechende Lösungen vor. Kapitel 2 beschreibt den Zusammenhang zwischen Retrieval-Theorie und IR-Software und motiviert einen Service-orientierten Lösungsansatz zur Implementierung von IR-Prozessen. Kapitel 3 diskutiert Formalismen, mit denen IR-Prozesse modelliert werden können, und Kapitel 4 erläutert die Konzepte von TIRA.

2 Von IR-Theorie zu IR-Software

Abhängig von einer gegebenen Retrieval-Aufgabe können verschiedene Aspekte eines Dokuments d wichtig sein, z. B. sein Layout, sein struktureller oder logischer Aufbau, oder seine Semantik. Eine Computerrepräsentation $\mathbf{d}$ von d muss die für die Retrieval-Aufgabe relevanten Aspekte widerspiegeln; bei der Konzipierung einer Repräsentation spielen linguistische Theorien, Algorithmen zur Textanalyse, Datenstrukturen zur Verwaltung großer Datenmengen und statistische Erkenntnisse eine Rolle. Eine optimale Repräsentation $\mathbf{d}$ ist sowohl auf die formalisierte Anfrage $\mathbf{q}$ gemäß der Retrieval-Aufgabe als auch auf das Retrieval-Modell $\mathcal{R}$ abgestimmt. Dabei umfasst $\mathcal{R}$ die linguistische Theorie, auf der die Abbildung $d \mapsto \mathbf{d}$ basiert, sowie die Funktion $\rho(\mathbf{q}, \mathbf{d})$, welche die Relevanz einer Abfrage $\mathbf{q}$ zu der Computerrepräsentation $\mathbf{d}$ eines Dokuments quantifiziert.

Abbildung 1 (unterhalb der gestrichelten Linie) illustriert diese Zusammenhänge; darüber ist die abstrakte Softwaretechniksicht dargestellt: der individuelle Informationsbedarf eines Anwenders wird durch einen IR-Prozess erfüllt.

Tatsächlich ist die gegenwärtige Praxis bei der Implementierung von IR-Prozessen *bibliotheksbasiert*: Funktionen, die mehr oder weniger komplexe Aufgaben lösen, werden mit generischen Schnittstellen versehen und in anderen Projekten wiederverwendet. Diese Praxis hat sich

[1] Akronym für „Text-based Information Retrieval Architecture“.

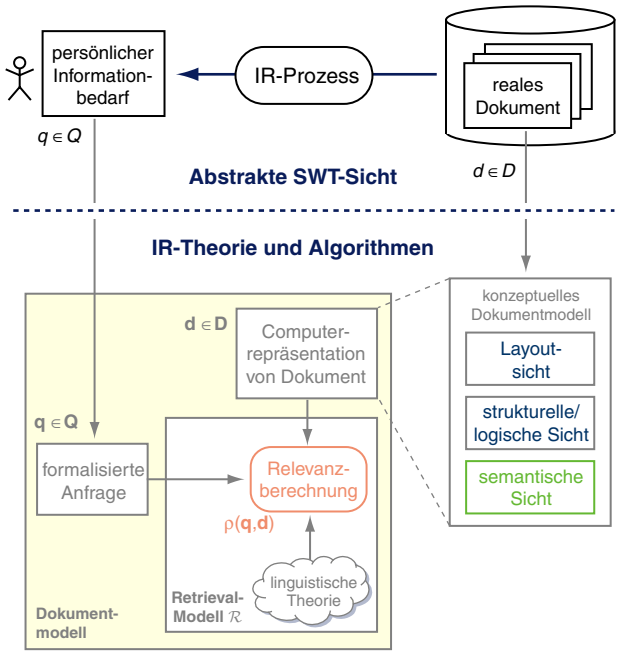


Abbildung 1: Als Ergebnis eines IR-Prozesses wird zu einem Informationsbedarf q ein passendes Dokument d geliefert (oberhalb der gestrichelten Linie). Die Realisierung dieses Prozesses bedingt die Abstraktion von q und d zu Computerrepräsentationen $\mathbf{q}$ bzw. $\mathbf{d}$ (unterhalb der gestrichelten Linie). Dieser Abstraktion liegt eine linguistische Theorie zugrunde, die in einem Retrieval-Modell $\mathcal{R}$ operationalisiert ist.

teilweise bewährt, sie berücksichtigt jedoch kaum die IR-spezifische Entwurfssituation:

- IR-Prozesse bestehen aus autonomen Software-Bausteinen, im folgenden als Module bezeichnet. Grundsätzlich stellt jedes Modul einen Dienst zur Verfügung, der eine Eingabedatenstruktur in eine Ausgabedatenstruktur überführt. Beispiele für solche Module sind Importfilter, Cluster-Algorithmen, Validitätsmaße, Ranking-Funktionen, Klassifizierer, POS-Tagger oder Visualisierungsalgorithmen.
- Typisch im Information Retrieval ist die Existenz alternativer Lösungen sowohl für ein und dieselbe Aufgabe als auch für verwandte Probleme.[2] Beispielsweise gibt es statistische und regelbasierte Stemming-Algorithmen [Porter 1980; Stein und Potthast 2006] oder interne, externe und korpusbasierte Schlüsselwortextraktionsverfahren.
- Unterschiedliche Aufgaben innerhalb eines IR-Prozesses können mit unterschiedlichen Parametrisierungen eines generischen Algorithmus gelöst werden.[3] Beispiele hierfür sind Stemming-Algorithmen oder Stopwort-Filter [Porter 2001], die abhängig von der gewünschten Sprache Regeln oder Wortlisten als Eingabeparameter erhalten.
- IR-Prozesse werden häufig modifiziert: sie werden optimiert, mit neuen Ideen erweitert und an sich ändernde Informationsbedürfnisse angepasst.
- Häufig lassen sich Teile von IR-Prozessen parallel ausführen, insbesondere wenn Dokumente bezüglich unterschiedlicher Fragestellung analysiert werden. Ein Beispiel hierfür ist die intrinsische Ähnlichkeitsanalyse einer Dokumentkollektion bzgl. Thema, Genre, oder Schreibstil [Ifrim *et al.* 2005; Stamatatos *et al.* 2000; Meyer zu Eissen und Stein 2006].
- Es gibt Standardmodule, die für fast jeden IR-Prozess von Nutzen sind. Hierzu zählen Module für das Stemming, Module für die Stopwortentfernung oder Konverter für Binärformate wie Adobe Acrobat (PDF) oder Microsoft Word.

Die dargestellten Punkte zeigen die modulare Natur von IR-Prozessen; dieser sollte bei einer Operationalisierung Rechnung getragen werden. In diesem Zusammenhang schlagen wir ein zweistufiges Konzept vor: Spezifikation eines IR-Prozesses als Diagramm (Schritt 1), das automatisch instanziiert und als verteiltes Softwaresystem implementiert wird (Schritt 2).

3 Spezifikation von IR-Prozessen

Als Beispiel für einen IR-Prozess betrachten wir die Aufgabe, ein Dokument sowohl nach Thema als auch nach Genre in eine Themen-Taxonomie bzw. Genre-Taxonomie einzuordnen [Meyer zu Eißen und Stein 2004]. Abbildung 2 zeigt eine Spezifikation des zugrunde liegenden IR-Prozesses in Pseudo-Code: aus einem Dokument mit der URL u werden Computerrepräsentationen für eine Themen-Kategorisierung und eine Genre-Kategorisierung erstellt, die als Eingabe für bereits konstruierte Klassifizierer dienen. Man beachte, dass mehrere Textrepräsentationen (HTML, Rohtext, gefilterter Text) notwendig sind, um die Repräsentationen zu generieren.

Diese Art der Spezifikation folgt dem eingangs beschriebenen bibliotheksbasierten Paradigma, und sie besitzt mehrere Schwächen: (*i*) der Austausch eines Moduls zieht fehleranfällige Datenstruktur- und Code-Ersetzungen nach sich, (*ii*) Expertenwissen bezüglich der zugrunde liegenden Softwarebibliothek ist notwendig, (*iii*) das Ausnutzen der Parallelität zwischen Teilaufgaben führt zu einem unflexiblen Design, da die Parallelität im Programm fest verdrahtet werden muss, u. a. in der Gestalt von Threads oder Remote-Function-Calls, (*iv*) auch die Verteilungsstrategie muss fest verdrahtet werden.

Einen Ausweg stellt die Spezifikation von IR-Prozessen auf einer konzeptuellen Ebene dar, beispielsweise mittels einer grafischen Modellierungssprache. In der Vergangenheit wurden verschiedene Modellierungstechniken für ähnliche Fragestellungen eingesetzt; sie lassen sich nach dem folgenden Schema einteilen [Teich 1997]:

1. kontrollflussdominant oder zustandsorientiert: endliche Automaten, UML Zustandsdiagramme
2. datenflussdominant oder aktivitätsorientiert: Datenflussgraphen, Petri-Netze, markierte Graphen, UML Aktivitätsdiagramme
3. strukturorientiert: Komponentenzusammenhangsdiagramme, UML Klassendiagramme, UML Verteilungsdiagramme
4. zeitorientiert: UML Zeitdiagramme
5. datenorientiert: ER-Diagramme
6. hybrid: Kombinationen der oben genannten Prinzipien, z. B. Kontroll/Datenflussgraphen

[2] Man beachte die Anwendbarkeit des in [Gamma *et al.* 1998] beschriebenen Strategy-Design-Patterns.

[3] Man beachte den Zusammenhang zum Factory-Design-Pattern und zum Decorator-Pattern aus [Gamma *et al.* 1998].

```
Input:   URL u, dictionary dict, stopword list stl.
Output:  genre and topic class for the document at URL u.

Text ht=download(u);
Text plainText=removeHTMLTags(ht);
Text filteredText=removeStopwords(plainText, stl);
FeatureVector topicModel=buildTopicModel(filteredText, dict);

Language lang=detectLanguage(plainText);
FeatureVector presentationFeatures=buildPresentationFeatures(ht);
FeatureVector posFeatures=buildPOSFeatures(plainText, language);
FeatureVector genreModel=union(presentationFeatures, posFeatures);

int topicClass=classifyTopic(topicModel);
int genreClass=classifyGenre(genreModel);

return(topicClass, genreClass);
```

Abbildung 2: IR-Prozess für eine Kategorisierungsaufgabe, spezifiziert in Pseudo-Code

Ein Großteil der IR-Prozesse kann als datenflussdominant angesehen werden, da sie von Anwendern mittels einer Anfrage gestartet werden und kein involviertes Modul ohne die Ausgabedaten seiner Vorgängermodule ausführbar ist.

Neben der Möglichkeit, Datenabhängigkeiten zu spezifizieren, muss ein Modellierungsansatz für IR-Prozesse es auch ermöglichen, Parallelität (Verzweigungen und Synchronisation) zu definieren. Weiterhin sollte ein Modellierungsansatz die Typisierung von Daten unterstützen, um leistungsfähige Constraints für die Menge möglicher Modulverbindungen definieren zu können. Abhängig von der Modellierungsgranularität kann es sinnvoll sein, Iterationen auf Teilprozessen mit den damit verbundenen Bedingungen zu formulieren.

Der folgende Abschnitt diskutiert gängige Modellierungswerkzeuge in Hinblick auf ihre Eignung zur Spezifikation von IR-Prozessen.

3.1 Petri-Netze

Ein Petri-Netz [Petri 1962; Teich 1997] ist ein Tupel $N = \langle S, T, F, c, w, m_0 \rangle$; dabei ist

- $S = \{s_1, \ldots, s_m\}$ eine Menge von Stellen,
- $T = \{t_1, \ldots, t_n\}$ eine Menge von Transitionen,
- $S \cap T = \emptyset$,
- $F \subseteq (S \times T) \cup (T \times S)$ eine Flussrelation; die Elemente in F werden auch als Kanten bezeichnet,
- $k : S \rightarrow \mathbf{N} \cup \{\infty\}$ eine Funktion, die eine Kapazitätsbeschränkung für jede Stelle definiert,
- $w : F \rightarrow \mathbf{N}$ eine Funktion, die Kantengewichte definiert,
- $m_0 : S \rightarrow \mathbf{N}_0$ eine Anfangsmarkierung mit der Eigenschaft $\forall s \in S : m_0(s) \leq k(s)$.

Ein Petri-Netz ist ein bipartiter Graph, dessen Knotenmenge aus einer Stellenmenge S und einer Transitionsmenge T besteht, wobei jede Stelle $s \in S$ bis zu $k(s)$ Marken aufnehmen kann. Stellen werden als Kreise notiert, Marken stellen Punkte in den entsprechenden Kreisen dar, das Symbol für eine Transition ist ein Balken, und gerichtete Kanten aus F werden als Pfeile zwischen Transitionen und Stellen notiert. Die Markierung m_0 definiert eine initiale Verteilung der Marken auf die Stellen, und eine beliebige Markierung ist mit $m : S \rightarrow \mathbf{N}_0$ bezeichnet. Abhängig davon, welche Transition feuert, ändert sich auch die Markierung. Abbildung 3 zeigt ein Petri-Netz, das die Kategorisierungsaufgabe modelliert.

Semantik von Petri-Netzen Für $x \in S \cup T$ bezeichne $\bullet x = \{y \mid (y, x) \in F\}$ den Vorbereich von x und $x\bullet = \{y \mid (x, y) \in F\}$ den Nachbereich von x. Eine Transition $t \in T$ heißt aktiviert unter einer gegebenen Markierung m genau dann, wenn gilt:

1. $\forall s \in \bullet t \setminus t\bullet : m(s) \geq w(s,t)$
2. $\forall s \in t\bullet \setminus \bullet t : m(p) \leq k(s) - w(t,s)$
3. $\forall s \in t\bullet \cap \bullet t : m(s) \leq k(s) - w(t,s) + w(s,t)$

Eine aktivierte Transition t kann feuern. Dadurch werden an jeder Stelle $s \in \bullet t$ genau $w(s,t)$ Marken konsumiert und für jedes $s \in t\bullet$ genau $w(t,s)$ Marken produziert.

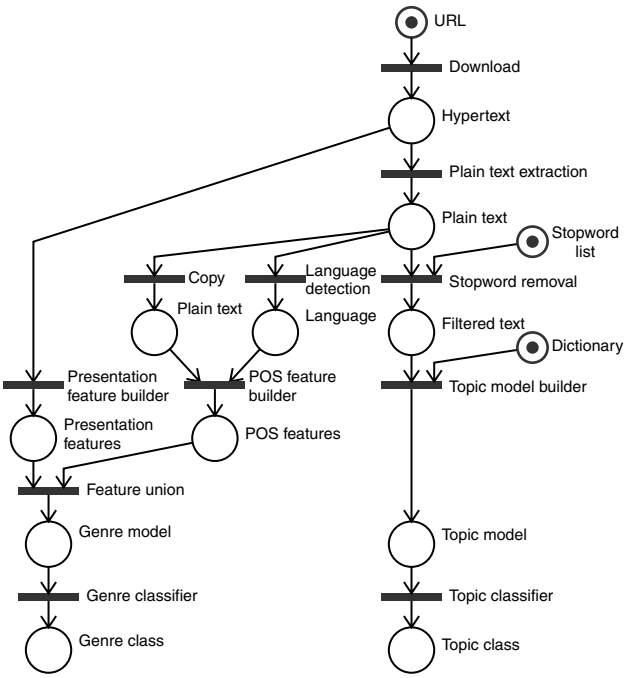


Abbildung 3: IR-Prozess für die Kategorisierungsaufgabe, spezifiziert als Petri-Netz.

Diskussion In unserem Szenario werden Module durch Transitionen modelliert, und die Marken entsprechen den Daten, die in Modulen verarbeitet und zwischen Modulen ausgetauscht werden. Petri-Netze können sowohl sequentielle als auch nebenläufige Prozesse modellieren: Abbildung 3 zeigt ein Petri-Netz für die Kategorisierungsaufgabe. Die beiden Klassifikationen werden parallel ausgeführt und lassen sich synchronisieren, um das Ergebnis einem Anwender anzuzeigen.

Petri-Netze sind gut erforscht; in den vergangenen vierzig Jahren wurden zahlreiche Werkzeuge zu ihrer Analyse und Simulation entwickelt, wie Algorithmen zur Ermittlung der Erreichbarkeit oder zur Feststellung von Deadlocks. Allerdings sind die Marken in Petri-Netzen nicht unterscheidbar und daher nicht zur Modellierung unterschiedlicher Datentypen geeignet. Des weiteren sehen Petri-Netze nicht vor, eine Verarbeitungsreihenfolge der Marken innerhalb der Stellen zu definieren, und es ist fraglich, ob eine Standardstrategie (z. B. FIFO) immer hinreichend für IR-Prozesse ist.

Die Beschränkung bezüglich der Datentypen lässt sich mit gefärbten Petri-Netzen aufheben [Jensen 1997]. Aber auch mit dieser Erweiterung bleibt das Modellieren von Kontrollflüssen stark eingeschränkt: Da Petri-Netze nicht in eine Marke hinein sehen können, lassen sich Kontrollflüsse, die von den Werten der Daten abhängen, nur umständlich modellieren.

3.2 Datenflussgraphen und Kontroll/Datenflussgraphen

Ein Datenflussgraph $G = \langle V, E \rangle$ ist ein gerichteter Graph, in dem jeder Knoten eine Aufgabe repräsentiert und jede gerichtete Kante einen Datenfluss zwischen ihren inzidenten Knoten darstellt. Die Semantik eines solchen Graphen ist, dass eine Aufgabe $v \in V$ nur dann ausgeführt werden kann, falls alle Aufgaben $u \in V$ mit $(u, v) \in E$ schon ausgeführt worden sind. Abbildung 4 zeigt einen Datenflussgraphen für die Kategorisierungsaufgabe.

Wenn man in dem abgebildeten Datenflussgraphen Knoten und Kanten durch Petri-Netz-Transitionen und -Stellen ersetzt, erhält man ein Petri-Netz, das isomorph zu Abbildung 3 ist. Folglich treffen die oben diskutierten Schwächen auch auf Datenflussgraphen zu. Die Schwäche, dass Kontrollstrukturen wie Iterationen schlecht modellierbar sind, trifft nicht mehr auf die ausdrucksstärkeren Kontroll/Datenflussgraphen (CDFG) zu. Diese nämlich ergänzen Datenflussgraphen um Kontrollflusskanten, die zur Modellierung von Kontrollflussalternativen sowie zur Modellierung von Iterationen benutzt werden können. Üblicherweise definieren Kontrollflusskanten alternative Pfade, von denen genau einer gemäß einer Bedingung begehbar ist.

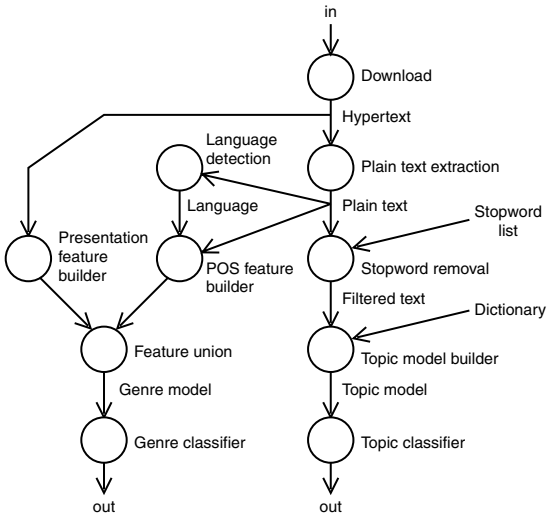


Abbildung 4: IR-Prozess für die Kategorisierungsaufgabe, spezifiziert als Datenflussgraph.

Diskussion CDFGs sind ausdrucksstark genug, um komplexe IR-Prozesse mit Verzweigungen und Iterationen zu modellieren. Allerdings zeigt die Datenflusskomponente in Abbildung 4, dass sich Datentypen und Synchronisation nur implizit durch Kantenbeschriftungen modellieren lassen. Dieses Defizit wird von dem intuitiven UML Modellierungsansatz behoben.

3.3 UML Aktivitätsdiagramme

UML Aktivitätsdiagramme vereinen neue Ideen, die Flusssprachen zur Spezifikation von Web-Service-Kompositionen (z. B. BPEL [Andrews *et al.* 2003]) zugrunde liegen, mit traditionellen Konzepten wie dem Markenkonzept von Petri-Netzen, um Kontroll- und Datenflüsse zwischen so genannten Aktionen zu modellieren. Insbesondere werden Aktionsknoten, Objektknoten und Kontrollknoten mit gerichteten Kanten verbunden, die sowohl Datenflüsse als auch Kontrollflüsse modellieren können [Hitz *et al.* 2005]. Abbildung 5 zeigt ein Aktivitätsdiagramm, das die Kategorisierungsaufgabe modelliert.

Ähnlich wie bei CDFGs werden IR-Basisdienste mit Knoten beschrieben, hier als Aktionsknoten bezeichnet. Objektknoten können zwischen Aktionsknoten platziert werden und stellen Datenobjekte dar, die zwischen den Aktionsknoten übertragen werden. Alternativ können auch Konnektoren, so genannte Pins, direkt mit Aktionsknoten

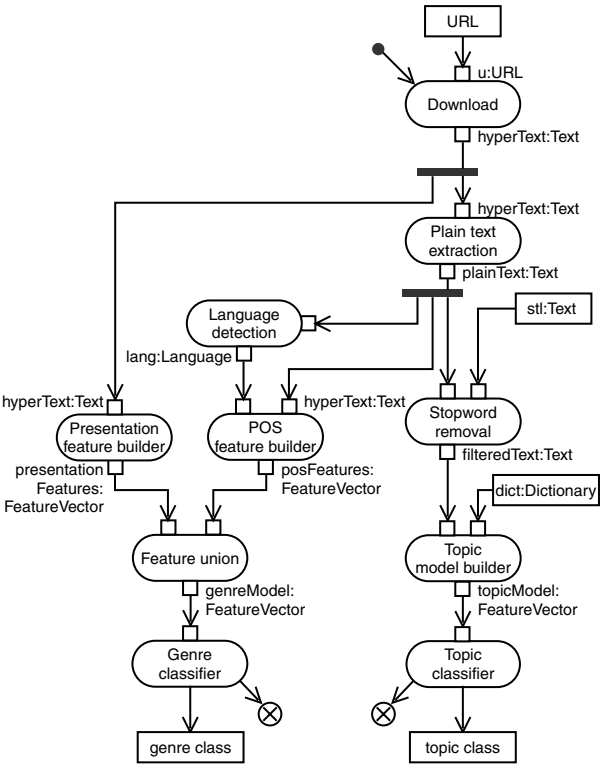


Abbildung 5: IR-Prozess für die Kategorisierungsaufgabe, spezifiziert als UML Aktivitätsdiagramm.

verknüpft werden, um Eingabe- und Ausgabedatentypen zu spezifizieren.

Kontrollknoten unterteilen sich weiter in Verzweigungsknoten (decision nodes), Verschmelzungsknoten (merge nodes), Parallelisierungsknoten (fork nodes) und Synchronisierungsknoten (join nodes). Verzweigungsknoten leiten den Kontrollfluss exklusiv über einen von mehreren möglichen Zweigen, abhängig von der Bedingung, die an einen Verzweigungsknoten gebunden sind; ihr Gegenstück sind die Verschmelzungsknoten. Der Beginn einer nebenläufigen Verarbeitung wird mittels Parallelisierungsknoten modelliert; Synchronisierungsknoten synchronisieren sowohl Daten- als auch Kontrollflüsse.

Ein Aktivitätsdiagramm kann in „Schwimmbahnen“ (swim lanes) unterteilt werden, wobei eine Schwimmbahn dazu dient, Knoten und Kanten hinsichtlich gemeinsamer Eigenschaften zu gruppieren. Solche logischen Einheiten (zum Beispiel in sich geschlossene Teile einer Retrieval-Aufgabe) werden vom Anwender definiert und erlauben es, einen komplexen IR-Prozess zu strukturieren.

Diskussion Nicht nur wegen ihrer Intuitivität sind UML Aktivitätsdiagramme weitläufig akzeptiert. Weiterführende Konzepte umfassen die Modellierung von Streams, Parametermengen, Stereotypen, aktions- und zeitgesteuerten Ereignissen sowie Ausnahmebehandlung. Diese Konzepte sind bereits in der aktuellen Version von UML standardisiert und machen die Diagramme zu einem idealen Modellierungswerkzeug für IR-Prozesse. Für die kommende UML 2.1-Spezifikation sind an Blockdiagramme erinnernde Bedingungs- und Iterationsknoten geplant; sie sollen helfen, die Modellierung von Kontrollflüssen noch weiter zu vereinfachen.

4 Operationalisierung von IR-Prozessen mit TIRA

Unter dem MDA-Paradigma[4] stellt eine UML-Spezifikation eines IR-Prozesses ein plattformunabhängiges Modell (PIM) dar [Object Management Group (OMG) 2003a], da Aktivitätsdiagramme nicht an Programmiersprachen, Betriebssysteme, Middleware oder Systemarchitekturen gebunden sind. Um einen als Aktivitätsdiagramm spezifizierten IR-Prozess zu operationalisieren, ist eine Zielplattform zu wählen und das PIM in ein ausführbares, plattformabhängiges Modell (PSM) zu übersetzen.

Der Begriff Plattform bezeichnet hier die nächste (= tiefere) Abstraktionsebene, auf der ein bestimmtes Modell konkreter beschrieben wird. Beispielsweise sind J2EE und CORBA mögliche Plattformen für die Implementierung von Geschäftsprozessen, und die Java-Entwicklungsumgebung ist eine mögliche Plattform für die CORBA-Implementierung. Das Beispiel verdeutlicht, wie durch eine Folge von Transformationen auf jeweils eine tiefere Ebene ein PIM ausführbar gemacht wird. Die OMG bezeichnet in diesem Zusammenhang alle Plattformen, die sich zwischen PIM und ausführbarem Code befinden, als Middleware-Plattform [Object Management Group (OMG) 2003a].

Wie in anderen MDA-basierten Anwendungsszenarien ist es unser Ziel, ausgehend vom PIM Transformationen auf tiefere Plattform-Ebenen hinsichtlich ihrer Semantik zu definieren und zu implementieren. Im Unterschied zu typischen MDA-basierten Anwendungsszenarien sind wir allerdings nicht daran interessiert, das PIM auf eine große Anzahl verschiedener Middleware-Plattformen abzubilden, sondern konzentrieren uns auf *eine* Zielplattform, die besonders geeignet ist, personalisierte IR-Prozesse auszuführen. Kurz gefasst: unser Fokus liegt auf schnellem Entwurf, kurzen Entwicklungszyklen und einem minimiertem Aufwand für Implementierung und Test.

Abbildung 7 zeigt unseren Vorschlag einer Schichtenarchitektur für TIRA: Eingabe ist ein PIM, das einen IR-Prozess in Form eines UML Aktivitätsdiagramms spezifiziert. Das Diagramm wird, wie im nächsten Abschnitt erläutert, in ein PSM überführt und in einer verteilten Umgebung ausgeführt. Die Module für die IR-Basisdienste stammen aus einer Modulbibliothek; sie sind als Web-Services gekapselt und über Rechnergrenzen hinweg transparent aufrufbar. Sowohl Eingabe als auch Ausgabe der Module sind Datenobjekte, serialisiert in der Form von XML-Datenströmen.

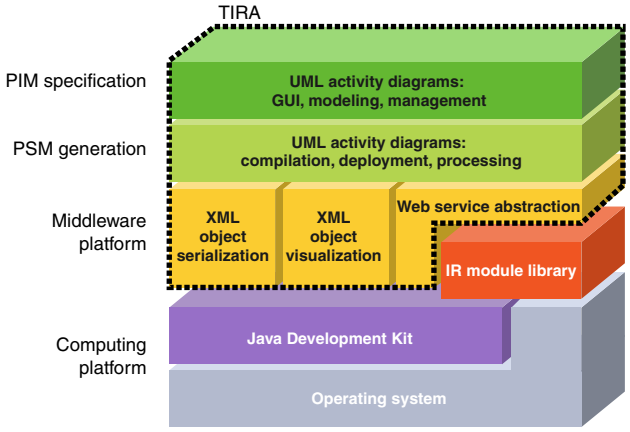


Abbildung 7: Die Schichtenarchitektur von TIRA setzt auf einer Rechnerplattform auf. Die IR-Modul-Bibliothek ist nicht Teil von TIRA, sondern steht für eine offene und erweiterbare Menge von IR-bezogenen Algorithmen und Datenstrukturen.

4.1 Von PIM zu PSM

Ein Aktivitätsdiagramm – sei es aus einer Datei geladen oder interaktiv mit der TIRA-GUI modelliert – wird speicherintern als Objektstruktur dargestellt. Diese Struktur ist am UML Metamodell der Object Management Group (OMG) [2003b] orientiert: Die Objekte der Struktur sind Instanzen von Aktionsknoten, Parallelisierungsknoten usw. und mittels Datenknoteninstanzen verbunden. Diese Struktur wird zu einem Kontroll-/Datenflussgraphen G übersetzt und mit einer Petri-Netz-artigen Markensemantik ausgeführt. Die Aktionsknoten in G werden an die entsprechenden Web-Services gebunden, die Datenknoten in G auf XML-Objekte abgebildet. Zur Abarbeitung von G werden zunächst alle ausführbaren Knoten bestimmt und die assoziierten Web-Services asynchron aufgerufen. Sobald ein Web-Service sein Ergebnis liefert, wird die entsprechende Marke in G propagiert, und die Menge der ausführbaren Aktionsknoten aktualisiert.

4.2 Die TIRA Middleware-Plattform

Die Funktionen in der Modulbibliothek erhalten Objekte als Eingabe und liefern neue Objekte zurück. Anstatt die

[4] MDA steht für Model Driven Architecture.

Web-Services mit Serialisierungen dieser Objekte aufzurufen, ist die Parameterübergabe mit dem Call-By-Name-Paradigma realisiert: Ein Parameter ist eine URL, die auf den Ort einer serialisierten XML-Repräsentation des entsprechenden Objekts zeigt. Dieser Ansatz hat folgende Vorteile:

1. Während der Ausführung eines IR-Prozesses muss ein Client nicht die Daten zwischen zwei Web-Service-Aufrufen übertragen. Stattdessen greift ein Web-Service unmittelbar auf die angegebenen URLs zu. So werden Datentransportkosten reduziert – insbesondere, wenn die Web-Services zweier aufeinander folgender Module auf demselben Server gehostet sind.
2. Der Transfer von URL-Referenzen anstelle von Datenobjekten ermöglicht es, auch Rechner mit geringer Bandbreite zu voll funktionstüchtigen Clients zu machen.
3. Die Verwendung von URLs schafft die Voraussetzung, das World Wide Web zur Datenspeicherung und -verteilung zu nutzen.

XML ist als Standard zum Datenaustausch und zur Serialisierung von Datenobjekten weit verbreitet. In TIRA werden zum Lesen und Schreiben von XML-Daten-Streams moderne Parser-Generatoren der Java-XML-Sprachbindung (JAXB) eingesetzt [Sun Microsystems 2003].

Die zwischen den Modulen ausgetauschten Daten lassen sich für eine visuelle Analyse oder zu Debugging-Zwecken angezeigen. Hierbei kommt die Technik der XSL-Transformationen zum Einsatz, mittels der serialisierte XML-Objekte auf Basis von XSL-Stylesheets in Formate wie XHTML oder PostScript on-the-fly umgewandelt werden können.

4.3 Arbeiten mit TIRA

Abbildung 6 (links) zeigt einen Snapshot unserer Meta-Suchmaschine AIsearch als TIRA-Anwendung. AIsearch sucht zu eingegebenen Schlüsselworten passende Web-Dokumente mit Hilfe kommerzieller Suchmaschinen und ordnet die gefundenen Dokumente nach inhaltlicher Ähnlichkeit [Meyer zu Eißen und Stein 2002]. Der zugrunde liegende IR-Prozess extrahiert die von den Suchmaschinen gelieferten Dokumentausschnitte, entfernt Stopworte, führt eine Wortstammreduktion durch und erzeugt eine komprimierte Term-Vektor-Darstellung. Auf Grundlage der Term-Vektoren wird eine Cluster-Analyse mit dem MajorClust-Algorithmus durchgeführt [Stein und Niggemann 1999] und für die gefundenen Themenkategorien aussagekräftige Bezeichnungen mittels statistischer Textüberdeckungsalgorithmen generiert.

Abbildung 6 (rechts) zeigt einen Snapshot des TIRA Aktivitätsdiagramm-Editors, der in einem Java Applet ausgeführt wird. Die linke Seite zeigt eine Auswahl der instanziierbaren Module und Datenknotentypen. Die Instanzen sind auf der rechten Seite des Applets zu sehen und können mit Pfeilen, die die Richtung des Datenflusses definieren, verbunden werden. Ein Klick auf einen Datenknoten startet die assoziierte XSL-Transformation, die die Daten in ein anzeigbares Format umwandelt, das dann im Browser dargestellt wird.

5 Zusammenfassung

IR-Prozesse haben eine ubiquitäre Präsenz erreicht, sei es in Form von Suchmaschinen im privaten oder professionellen Kontext, auf mobilen Endgeräten oder auf stationären Rechnern, oder als Retrieval-Komponenten in Dateisystemen, Dokumentkollektionen, Datenbanken oder Wissensmanagement-Werkzeugen. Der Grund dieser Durchdringung ist ein wachsender Informationsbedarf, die Vielfältigkeit der IR-Aufgaben, und ein gewünschter Grad an Personalisierung. Obgleich viele spezialisierte Retrieval-Algorithmen in der Vergangenheit entwickelt wurden, sind wenig Anstrengungen unternommen worden, IR-Prozesse aus der Sicht der Softwaretechnik zu modellieren und zu operationalisieren.

Das vorliegende Papier soll an dieser Stelle einen Beitrag leisten: Ausgehend von einer Diskussion einschlägiger Modellierungstechniken bezüglich ihrer Eignung für die Abstraktion von IR-Prozessen haben wir TIRA als eine flexible MDA-Lösung für den schnellen Entwurf maßgeschneiderter IR-Werkzeuge vorgestellt. Mit TIRA ist es möglich, IR-Prozesse als UML Aktivitätsdiagramme zu modellieren, die per Knopfdruck in plattformspezifische Modelle transformiert und in einer verteilten Umgebung ausgeführt werden können.

TIRA hat zurzeit den Status eines Forschungsprototyps und wird in unserer Arbeitsgruppe weiterentwickelt. Der

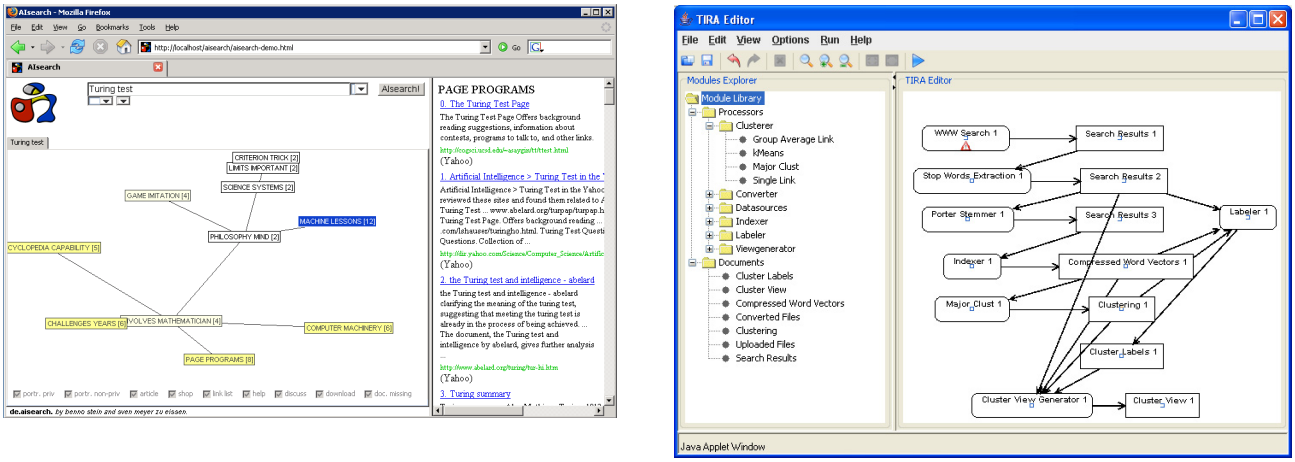


Abbildung 6: Der linke Snapshot zeigt unsere Meta-Suchmaschine AIsearch, deren IR-Prozess mit TIRA modelliert ist. Der rechte Snapshot zeigt den TIRA-Editor für die Spezifikation von IR-Prozessen mittels Aktivitätsdiagrammen.

TIRA-Ansatz ist unabhängig von IR-Algorithmen und Datenstrukturen; unser Ansatz sieht die Einbindung existierender IR-Bibliotheken (und das hierin codierte Know-How) explizit vor.

Literatur

Tony Andrews, Francisco Curbera, Hitesh Dholakia, Yaron Goland, Johannes Klein, Frank Leymann, Kevin Liu, Dieter Roller, Doug Smith, Satish Thatte, Ivana Trickovic und Sanjiva Weerawarana. Business process execution language for web services (bpel4ws) version 1.1. `http://www-128.ibm.com/developerworks/library/specification/ws-bpel/`, Mai 2003.

Erich Gamma, Richard Helm, Ralph Johnson und John Vlissides. *Design Patterns: Elements of Reusable Object-oriented Software*. Addison-Wesley Longman Publishing Co., Inc., 1998.

Martin Hitz, Gerti Kappel, Elisabeth Kapsammer und Werner Retschitzegger. *UML @ Work*. dpunkt.verlag, 2005.

Georgiana Ifrim, Martin Theobald und Gerhard Weikum. Learning Word-to-Concept Mappings for Automatic Text Classification. In *Proceedings ICML, Learning in Web Search Workshop*, 2005.

Kurt Jensen. *Coloured Petri Nets. Basic Concepts, Analysis Methods and Practical Use.*, Volume 1 der *Monographs in Theoretical Computer Science*. Springer, 1997.

Sven Meyer zu Eißen und Benno Stein. The AIsearch Meta Search Engine Prototype. In Amit Basu und Soumitra Dutta, Eds., *Proceedings 12th Workshop on Information Technology and Systems (WITS 02), Barcelona Spanien*. Technische Universität Barcelona, Dezember 2002.

Sven Meyer zu Eißen und Benno Stein. Genre Classification of Web Pages: User Study and Feasibility Analysis. In Susanne Biundo, Thom Frühwirth und Günther Palm, Eds., *KI 2004: Advances in Artificial Intelligence*, Volume 3228 *Lecture Notes in Artificial Intelligence*, S. 256-269, Berlin Heidelberg New York, September 2004. Springer.

Sven Meyer zu Eissen und Benno Stein. Intrinsic plagiarism detection. In Mounia Lalmas, Andy MacFarlane, Stefan M. Rüger, Anastasios Tombros, Theodora Tsikrika und Alexei Yavlinsky, Eds., *Proceedings European Conference on Information Retrieval (ECIR 2006)*, Volume 3936 *Lecture Notes in Computer Science*, S. 565-569. Springer, 2006.

Object Management Group (OMG). Model driven architecture (mda) guide. `http://www.omg.org/docs/omg/03-06-01.pdf`, 2003.

Object Management Group (OMG). The UML Metamodel. `http://www.omg.org/cgi-bin/doc?ptc/2004-10-05`, 2003.

Carl Adam Petri. *Kommunikation mit Automaten*. Dissertation, Universität Bonn, 1962.

M. F. Porter. An Algorithm for Suffix Stripping. *Program*, 14(3):130-137, 1980.

Martin Porter. Snowball. `http://snowball.tartarus.org/`, 2001.

E. Stamatatos, N. Fakotakis und G. Kokkinakis. Text genre detection using common word frequencies. In *Proceedings 18th Int. Conference on Computational Linguistics*, Saarbrücken, 2000.

Benno Stein und Oliver Niggemann. On the Nature of Structure and its Identification. In Peter Widmayer, Gabriele Neyer und Stefan Eidenbenz, Eds., *Graph-Theoretic Concepts in Computer Science*, Volume 1665 *Lecture Notes in Computer Science*, S. 122-134. Springer, Juni 1999.

Benno Stein und Martin Potthast. Putting Successor Variety Stemming to Work. In *30th Annual Conference of the German Classification Society (GfKl) 2006 (erscheint in Kürze)*, 2006.

Sun Microsystems. Java Architecture for XML Binding (JAXB Specification). `http://java.sun.com/xml/downloads/jaxb.html`, 2003.

Jürgen Teich. *Digitale Hardware/Software-Systeme*. Springer, 1997.

Users' Effectiveness and Satisfaction for Image Retrieval

Azzah Al-Maskari
Dept. of Information Studies
University of Sheffield
Sheffield, S1 4DP, UK
Lip05aaa@shef.ac.uk

Paul Clough
Dept. of Information Studies
University of Sheffield
Sheffield, S1 4DP, UK
p.d.clough@shef.ac.uk

Mark Sanderson
Dept. of Information Studies
University of Sheffield
Sheffield, S1 4DP, UK
m.sandersonh@shef.ac.uk

Abstract

This paper presents results from an initial user study exploring the relationship between system effectiveness as quantified by traditional measures such as precision and recall, and users' effectiveness and satisfaction of the results. The tasks involve finding images for recall-based tasks. It was concluded that no direct relationship between system effectiveness and users' performance could be proven (as shown by previous research). People learn to adapt to a system regardless of its effectiveness. This study recommends that a combination of attributes (e.g. system effectiveness, user performance and satisfaction) is a more effective way to evaluate interactive retrieval systems. Results of this study also reveal that users are more concerned with accuracy than coverage of the search results.

1 Introduction

The performance of Information Retrieval (IR) systems is typically quantified using metrics derived from the number of relevant items found. Commonly used measures include Mean Average Precision (MAP), Precision at 10 documents retrieved (P@10), and bpref. Much of IR research has focused on improving these metrics, assuming that higher system effectiveness will help users to find more useful information. Some recent studies have shown that system performance. For example, Allan et al. (2005) have reported that a high increase in system effectiveness did not have detectable gains for the user. . Järvelin and Ingwersen (2004) assert that the real issue in IR systems design is not whether recall/precision goes up by a statistically significant percentage, but whether it helps the user solve the search task more effectively. Knowledge about what satisfies users is therefore crucial to improving retrieval systems. Factors such as prior search experience, search strategies and knowledge about the topic are also expected to influence the effectiveness of retrieval.

This paper examines the information seeking behaviour of users querying an interactive Arabic image retrieval system. The aim of this study is to compare users' results with system performance and investigate the influence of factors such as users' perception of the task/topic and their judgements of the search results. Users' preference of coverage and accuracy of the results is also analysed.

We review past literature related to this topic in section 2; describe the system used in these experiments, the methodology and search tasks in section 3; present results of system and user evaluation in section 4 and provide discussion and conclusions in sections 5 and 6.

2 Related Research in User-Based Retrieval Evaluation

Recent studies have demonstrated that improvements in IR system effectiveness metrics do not translate into a direct benefit for end-users. A recent study by Turpin and Scholer (2006) attempted to address the relationship between the effectiveness of an IR system and how it matched up with user performance in a simple web search. Systems at various levels of MAP were used and assessment based on users performing a precision-search task measured by the length of time needed to find a single relevant document. Users also performed a recall-based task, measured by the total number of relevant documents users could identify in five minutes. There was no correlation between system performance measured with MAP and user performance on the precision task, and only a negligible improvement in performance on the recall task when MAP was increased.

A study by Hersh et al. (2000) showed that instance recall - where users try to identify different aspects of a question within a limited timeframe – did not improve with small increases in MAP of the underlying search system on the scale that is commonly reported in IR results. Allan et al. (2005) confirmed this result (using bpref), but also showed that for larger, specific increases in bpref, users did benefit on an instance recall task. Turpin and Hersh (2001) demonstrated a lack of improvement when users were engaged in a question answering task for a small number of questions.

The experiments of (Hersh et al., 2000), (Allan et al., 2005) and (Turpin and Hersh, 2001) have focused on recall-based tasks, whereas MAP is a precision-oriented measure. So, previous search tasks are different from what the employed effectiveness metrics are aiming to capture. The latest experiments of Turpin and Scholer (2006) were based on both recall and precision tasks and system effectiveness measured using MAP. In sum, from all these four studies, one can conclude that improvements in

system effectiveness as measured using MAP, P@10, and bpref does not translate into a direct benefit to users. Therefore in this study, it was decided to use both recall-oriented and precision-oriented measures and compare them with the users' searching behaviour. In addition, in all the four previous experiments are based on text retrieval systems, whereas this experiment is based on image retrieval which we assumed could give different results but it did not. Furthermore, this study combines the results of both qualitative and quantitative analysis by taking into account system performance, users' performance, and users' perception of the tasks they performed (i.e., task difficulty, interestingness) and users' satisfaction of the search results.

3 Experiment Methodology

This study was conducted in conjunction with a submission to iCLEF 2006[1], and therefore restricted to the guidelines of iCLEF (e.g. methodology and number of topics). Previous studies had shown that a high increase in system effectiveness did not have a significant impact on the users' performance; only marginal gains had been reported. Therefore, for this experiment it was decided to test users' performance using just one system with acceptable retrieval effectiveness.

3.1 System Description

The system used on this experiment is based upon FLICKR[2]. Users query the system in Arabic which is translated into English, French, Spanish, German, Italian, and Dutch (with English used as an interlingua between Arabic and the other languages). Users are presented with results in which images are annotated in different languages, thereby increasing recall (different images are annotated with different languages). The user is able to edit the English translation of the Arabic query prior to search. More details of this system can be found in (Clough et al., 2006). The motivation for this study comes from wanting to experiment with Arabic users, and the availability of local resources to run the experiment. According to ABC news[3], there has been a rapid increase of Arabic users online and therefore we believe that many Arabic users would like to access FLICKR but don't have the necessary language skills to formulate multilingual queries.

3.2 Data Collection

Data collected for this experiment consisted of both qualitative (IR effectiveness metrics) and quantitative measures (pre-search questionnaire, task questionnaire, and exit questionnaire). Each user retrieved images for two types of tasks: 1) Classical ad-hoc task: "Find as many European parliament buildings as possible, pictures from the assembly hall as well as from the outside" and 2) Find five illustrations to the text "The story of saffron", the goal being to find five distinct instances of information described in a given narrative (saffron flower, saffron thread, picking the thread/flower, powder, dishes with saffron). The time allotted was 20 minutes per task

3.3 Users

Eleven Arabic students (postgraduate and undergraduate) with a median age 28 were recruited via email for this experiment. The work was conducted under the guidelines of Human Ethics Committee of Sheffield University. Most users reported having a great deal of experience with on-line searching (82%) and searching for images (45%).

4 Results

This section presents the results of system effectiveness, users' effectiveness, their perception and satisfaction of the results followed by a comparison between users' performance and system effectiveness.

4.1 System Effectiveness

A combination of binary[4] relevance and graded[5] relevance measures were used to evaluate system effectiveness. The system was assessed based on its retrieval results for the "European parliament" and "saffron" queries. The system was measured without query reformulation since more than half of the users did not reformulate the query during the search process. Table 1 illustrates that the system performs at similar levels of effectiveness for both tasks. Following is a brief description about each measure:

- Normalized P@100: precision over the first 100 images - normalised by the minimum of 100 or the number of retrieved images(Buckley and Voorhees, 2000).
- Q-measure: based on graded relevance and cumulative gain, designed for the task of finding many relevant items (Sakai, 2005). In this experiment Q-measure is computed until rank 10.
- bpref-10: Binary preference is the number of times nonrelevant images are retrieved/judged before relevant images (Buckley and Voorhees, 2004). In this experiment bpref-10 is computed until rank 50.
- R-precision is the precision after *R* images are retrieved where *R* is the number of relevant images for a given topic (Buckley and Voorhees, 2000).

Task	P @50 norm	P@10 0norm	Q-measure	bpref-10	10-Precision
Parliament	0.48	0.46	0.27	0.48	0.58
Saffron	0.45	0.48	0.42	0.39	0.54

Table 1- System effectiveness (average)

4.2 Measuring Users' effectiveness

Tables 2 and 3 illustrate users' performance and satisfaction of both tasks. The evaluator assessed the images retrieved by users. Recall captures how well the subjects find different aspects of the topic. For the saffron

[1] http://nlp.uned.es/iCLEF/

[2] http://www.flickr.com/
FLICKR is a large-scale web-based image retrieval database. It is used to manage and share personal and commercial photographs and currently contains over five million freely accessible images.

[3] http://www.abc.net.au/news/newsitems/200604/s1624108.htm

[4] image is relevant or not relevant

[5] image is highly relevant, partially relevant, or not relevant

task: saffron flower, saffron thread, picking the thread/ flower, powder, dishes; the European parliament task: images of inside and out of different buildings from different European countries. Users are given one point for retrieving each true instance (unique images) and no credit for repeated instances (i.e. no credit for retrieving two saffron flowers. For the parliament task, one point was given for retrieving an inside image and another point for an outside image of the same building (unique images). Thus, users' recall for the parliament task was calculated as (number of unique images/ correct images retrieved) and for the saffron task as (number of unique images/ total required images), which is five in this case. Precision captures the proportion of correct relevant images to the total number of images retrieved. Precision for the parliament task was computed as (correct images/total images retrieved) and for the saffron task as (number of unique images/ total retrieved). Users' precision in the parliament task is better than the saffron task due to their perception of the topic according to the information obtained from the task questionnaire: familiarity with the topic, topic easiness and their interest in the topic. There is a moderate degree of correlation[6] between *familiarity* and user' precision (*p=0.7)* in the parliament task. Users' ability to achieve full recall and precision is fluctuates in both tasks (shown by Tables 2 and 3).

	Precision	U. Sat. Accuracy*	Recall	U. Sat. coverage*	Use ful*
user1	0.40	1.00	0.4	0.50	1
user2	0.80	0.50	0.8	0.50	0.5
user3	1.00	0.50	0.8	1.00	1
user4	0.60	1.00	0.6	1.00	1
user5	0.33	1.00	0.2	1.00	1
user6	0.80	1.00	0.8	1.00	0.5
user7	1.00	0.50	1	0.50	1
user8	1.00	0.50	1	0.50	0.5
user9	1.00	1.00	1	1.00	1
user10	0.40	0.50	0.4	0.50	0.5
user11	0.60	0.50	0.6	0.50	1
Average	**0.72**	**0.73**	**0.69**	**0.73**	**0.82**

Table 2 -Users' performance versus satisfaction- Parliament task

4.3 User' Prediction of the Search Results

Users' opinions and expectations of search tasks were extracted from the task questionnaire. Users rated the results in terms of their relevancy[7], satisfaction with usefulness of the results, satisfaction with the accuracy (efficiency) and coverage (completeness) of the results. According to the Tables 2 and 3, there is no significant correlation between users' estimation of results and their actual performance, except between usefulness of the results and users' recall (*p=0.02*). This suggests that users are not able to easily assess the success of their search. On average, users believed they had completed the task when they had actually achieved 69% recall. In general users were satisfied with the system despite the fact this did not reflect on their performance.

	Precision	U. Sat Accuracy*	Recall	U. Sat. Coverage*	Use ful *
user1	0.50	1.00	0.50	1.00	1
user2	0.55	1.00	0.27	1.00	1
user3	0.73	1.00	0.88	0.50	1
user4	0.85	1.00	0.65	1.00	1
user5	0.92	1.00	0.67	1.00	1
user6	0.83	1.00	0.90	0.00	0.5
user7	1.00	0.50	0.67	0.50	0.5
user8	0.88	0.00	0.86	0.00	1
user9	1.00	1.00	0.75	1.00	1
user10	0.77	1.00	0.70	1.00	1
user11	0.90	1.00	0.78	1.00	1
Average	**0.81**	**0.86**	**0.69**	**0.73**	**0.91**

Table 3-Users' performance versus satisfaction saffron task

4.4 Comparison between System and User Effectiveness

The differences between system and user effectiveness were measured using an analysis of variance (ANOVA). According to the results in Table 4, there is a statistically significant correlation between users' performance, gauged by users' recall in addition to their satisfaction of the coverage of the results, and system effectiveness when measured by Q-measure. In general, there is no correlation between users' effectiveness and system as quantified by P@100 and 10-Precision, except for users' precision and the system P@100 in the parliament task. Although there is a strong correlation between users coincident of the usefulness of the results and system bpref-10 on both tasks it did not reflect on their performance or satisfaction of the accuracy and coverage of the results. The lack of correlation between users and the system as determined by precision-oriented metrics indicate that these metrics are not compatible with user satisfaction and performance.

Therefore, users' effectiveness is by and large inconsistent with the system effectiveness as measured solely by traditional IR metrics. This conclusion gives further credence to the findings of (Hersh et al., 2000), (Turpin and Hersh, 2001),(Allan et al., 2005), and (Turpin and Scholer, 2006) in that improvements in the

[6] Measured using the Pearson correlation coefficient.

[7] If images directly address the core issue of the topic then *highly-relevant,* if they contain helpful information then *partially relevant*, otherwise *not relevant*.

* U=user; sat= satisfaction
1=very satisfied; 0.5=partially satisfied, 0=not satisfied

metrics of systems (P@10, MAP, bpref-10) do not translate into a direct benefit for the users. The aforementioned experiment indicate that user satisfaction with the results provide a better picture of system' accuracy than classical measures.

Effectiveness measures	Parliament Task	Saffron Task
Users' recall vs. Q-measure	**p=5.97E-05**	**p=0.039**
Users' satisfaction with coverage vs. Q-measure	**p=0.011**	**p=0.018**
Users' precision vs. P@100	**p=0.005**	p=0.185
Users' precision vs. 10-Precision	p=0.09	p=0.262
Users' satisfaction with accuracy vs. 10-Precision	p=0.11	p=0.200
Users' precision vs. bpref-10	P=0.059	P=0.076
Users' recall vs. bpref-10	P=0.166	P=0.123
Users' satisfaction with accuracy vs. bpref-10	P=0.064	P=0.065
Users' satisfaction with coverage vs. bpref-10	P=0.210	P=0.065
usefulness of results vs. bpref-10	**P=0.028**	**P=0.013**

Table 4- System versus users' effectiveness

4.5 Accuracy vs. Coverage of Search Results

In the search task questionnaire, users identified their preference of accuracy or coverage according to the tasks. Accuracy was defined to the users as "relatedness of results to the search topic" and coverage as "coverage of results to all aspects of topic". Most users (68%) opted for accuracy over recall (32%). This implies that whether users are looking for few or many images, they are concerned with quality than quantity. Users seem to prefer having fewer highly relevant images than a larger proportion of relevant images. Hence systems with adequate precision may contain what the users are looking for.

5 Discussion

This study reports the relationship between IR system effectiveness and user effectiveness by using 11 subjects to search using an image retrieval system for recall-based tasks. Users are required to find as many relevant images of the European parliament and find for five different instances of saffron where users' recall was measured by the number of instances saved.

Results demonstrate that users were highly satisfied with the system's performance despite the system not being of high quality as measured using P@100, R-precision and the Q-measure. Results revealed a significant relationship between users' recall and the system' Q-measure in both tasks. Therefore, Q-measure, a recall-oriented measure, can be more useful when comparing system versus users' performance. Precision measures do not seem to correlate well with user performance as there is no significant relationship between users' precision when compared with the system P@100 for the saffron task. One possible explanation for the lack of correlation between the system and users in the saffron task is the users' familiarity with the search topic, affecting the quality of the search results. For both tasks, observations indicated that some users are just better than others at searching.

6 Conclusions

The conclusion of this experiment begins to answer the doubt expressed by Turpin and Scholer (2006) over whether a direct relationship between IR effectiveness measures and users satisfaction with search results exists. This experiment reinforces the findings of previous studies in that there does not appear to be a strong relationship between the performance of a system and the user. . It was found that users can find what they are looking for despite a fairly low level of system effectiveness. This indicates that results for experiments based on system measures are not comparable with experiments based on real users. The fact that system languages are query languages differ in this experiment, we are not generalizing our conclusion to all IR systems.

It is believed that different types of topics and tasks lead to different levels of quality in the search results. While this experiment is limited to two topics, assessment based on system performance only does not interpret system quality and additional analysis of a users' satisfaction with the results presents a more holistic view of search performance. We are planning to conduct further work to determine what really satisfies the user of an IR system. This includes a further study that look into measures of system performance such as speed, accuracy, coverage, presentation of the results, and language related aspects together with a larger number of topics and more diverse tasks. Additionally, more study to investigate what other measures correlate with users' performance besides Q-measure for both recall-based and precision-based tasks.

References

Allan, J., Carterette, B. & Lewis, J. (2005). "*When Will Information Retrieval Be "Good Enough"? User Effectiveness As a Function of Retrieval Accuracy. 2005. " In: Proc ACM SIGIR 433-440, Salvador, Brazil*

Buckley, C. & Voorhees, E. M. (2000). "*Evaluating Evaluation Measure Stability" In: Proc ACM SIGIR, 33 - 40 Athens, Greece*

Buckley, C. & Voorhees, E. M. (2004). "*Retrieval evaluation with incomplete information" In: Proc SIGIR, 25-32, Sheffield, United Kingdom*

Clough, P., Al-Maskari, A. & Darwish, K. (2006). (in Press). "*FLICKRArabic: multilingual access to photos" In: Proc, iCELF.*

Flickr. http://www.flickr.com/

Hersh, W., Turpin, A., Price, S. & Chan, B. (2000). "*Do Batch and User Evaluations Give the Same Results?" In: Proc SIGIR 17-24, Athens, Greece*

iCLEF. 2006. Interactive track for the cross-Language Evaluation Forum. http://nlp.uned.es/iCLEF/

Reuters and ABC Science Online. 2006. Search engine to target Arabic speakers. *ABC NEWSONLINE* http://www.abc.net.au/news/newsitems/200604/s1624108.htm

Sakai, T. (2005). "*The Reliability of Metrics Based on Graded Relevance " In: Proc Information Retrieval Technology: Second Asia Information Retrieval Symposium, AIRS 2005, Jeju Island, Korea, October 13-15, 2005. Proceedings Korea*

Järvelin, K. & Ingwersen, P. (2004). Information seeking research needs extension towards tasks and *technology*. *Information Research*, 10(1)

Turpin, A. & Scholer, F. (2006). "*User Performance versus Precision Measures for Simple Search Tasks" In: Proc SIGIR, Seatle, Washington, USA*

Turpin, A. H. & Hersh, W. (2001). "*Why batch and user evaluations do not give the same results" In: Proc ACM SIGIR, 225 - 231 New Orleans, Louisiana, United States*

GeoCLEF 2006: Cross-linguales geographisches Information Retrieval

Kerstin Bischoff, Thomas Mandl, Christa Womser-Hacker
Informationswissenschaft, Universität Hildesheim
Marienburger Platz 22
D-31141 Hildesheim, Deutschland
{mandl, womser}@uni-hildesheim.de

Abstract

Der speziellen Behandlung geographischer Suchanfragen wird im Information Retrieval zunehmend mehr Beachtung geschenkt. So gibt der vorliegende Artikel einen Überblick über aktuelle Forschungsaktivitäten und zentrale Problemstellungen im Bereich des geographischen Information Retrieval, wobei speziell auf das Projekt GeoCLEF im Rahmen der cross-lingualen Evaluierungsinitiative CLEF eingegangen wird. Die Informationswissenschaft der Universität Hildesheim hat in diesem Projekt sowohl organisatorische Aufgaben wahrgenommen als auch eigene Experimente durchgeführt. Dabei wurden die Aspekte der Verknüpfung von Gewichtungsansätzen mit Booleschem Retrieval sowie die Gewichtung von geographischen Eigennamen fokussiert. Anhand erster Interpretationen der Ergebnisse und Erfahrungen werden weiterer Forschungsbedarf und zukünftige, eigene Vorhaben wie die Überprüfung von Heuristiken zur Query-Expansion aufgezeigt.

1 Einleitung

Häufig werden Informationen nicht nur zu einem speziellen Thema, sondern auch mit Bezug auf eine bestimmte geographische Region gesucht. Die (Weiter-) Entwicklung von Geographischen Informationssystemen (GIS), in denen raumbezogene Daten in strukturierter Form gespeichert werden, speziell räumlich abfragt und als Karten visualisiert werden können, weist daher bereits eine lange Tradition auf. Prominente Anwendungen in diesem Bereich sind Fachinformationssysteme bspw. für Umweltdaten, Verkehrs- und Routenplanung oder die digitalen Gelben Seiten. Weniger alt ist die Forschung im Bereich des Geographischen Information Retrieval (GIR), dem Zugänglichmachen und Auffinden von geographisch referenzierten Informationen aus unstrukturierten Daten wie Webdokumenten.

Erste Forschungsarbeiten untersuchen in diesem Kontext die Angemessenheit von erprobten textbasierten IR-Techniken, deren Erweiterungsmöglichkeiten durch externe Wissensressourcen sowie die Potentiale des Einsatzes räumlicher Indexierungs- oder Retrieval-Methoden (z.B. basierend auf Koordinaten zu Längen- und Breitengrad). Problemstellungen sind dabei u.a. die Erkennung und Disambiguierung von geographischen Eigennamen (z.B. *Washington*), die Ähnlichkeitsbestimmung bei vagen Anfragen (z.B. "*in der Nähe von*", *Norddeutschland*) und die Visualisierung der Ergebnisse.

Im Rahmen des Cross Language Evaluation Forum (CLEF) evaluiert das Projekt GeoCLEF GIR-Systeme darüber hinaus auch unter dem Gesichtspunkt der Mehrsprachigkeit, dem Umgang mit geographischen Informationen, die in unstrukturierten Dokumenten verschiedener Sprachen vorliegen. Denn gerade im Hinblick auf die mehrsprachigen Informationen im World Wide Web entsteht hier Mehrwert, wenn monolinguale Anfragen des Benutzers mithilfe maschineller Übersetzungstechniken auch relevante Dokumente anderer Sprachen liefern.

Vieles spricht dabei für die zentrale Rolle von Eigennamen und deren adäquaten Behandlung im Cross-Language Information Retrieval (CLIR). So ist auch im Cross-Language GIR anzunehmen, dass geographische Eigennamen eine hohe Diskriminierungsfähigkeit aufweisen und somit entscheidende Information tragen. Eine korrekte Erkennung dieser Eigennamen ist die Voraussetzung, um geeignete Methoden im Übersetzungsprozess anzuwenden, bspw. in Hinsicht auf die Kompositazerlegung (z.B. *Neuengland* => *New England* vs. *new narrow country*). [Womser-Hacker, 2006]

Nach einem kurzen Überblick über aktuelle Forschungaktivitäten zum GIR wird das Projekt GeoCLEF mit seinen Methoden und Ergebnissen vorgestellt. Es werden eigene Experimente in GeoCLEF beschrieben und potentielle Anschlussarbeiten skizziert.

2 Geographisches Information Retrieval

Um den geographischen Bezug einer Anfrage bzw. eines Dokumentes ermitteln zu können, müssen in einem ersten Schritt geographische Eigennamen korrekt erkannt werden (*Geo-Parsing*). Zur Named Entitiy Recognition (NER) existieren grundsätzlich drei Ansätze: listenbasiert, regelbasiert und mittels maschineller Lernverfahren.

Bei dem Abgleich mit vorgefertigten Listen – im Falle geographischer Eigennamen werden dazu geographische Thesauri, so genannte *Gazetteers*[1], genutzt – dürfte es sich um die zuverlässigste Art handeln, Eigennamen in unstrukturiertem Text zu erkennen. Jedoch sind derartige Ressourcen selten ausreichend umfassend, besonders im Hinblick auf Namensvarianten (*Los Angeles, Stadt der Engel*), und selten frei verfügbar. Regelbasierte Verfahren hingegen arbeiten nach intellektuell erstellten gram-

[1] z.B. Getty Thesaurus of Geographic Names: http://www.getty.edu/research/conducting_research/vocabularies/tgn/; GEOnet World Place Names Server: http://earth-info.nga.mil/gns/html/; World-Gazetteer: http://www.world-gazetteer.com

matikalischen Regeln für das Auftreten von Eigennamen in einer bestimmten Sprache. Dabei scheint die Aufgabe für die deutsche Sprache bspw. wesentlich schwieriger als für die englische, da hier alle Nomen groß geschrieben werden und die Wortstellung mehr Freiheiten zulässt [Womser-Hacker, 2006].

Die Problematik der Sprachabhängigkeit teilen maschinelle Lernverfahren, auch wenn der hohe Aufwand für die Modellierung bzw. Anpassung von Regeln an andere Sprachen durch deren automatische Erstellung anhand annotierter Trainingskorpora entfällt. Wegen der Unzulänglichkeiten der einzelnen Verfahren verwenden viele Systeme mehrere Ansätze sequentiell. In GATE, Teil des SPIRIT-Projektes, [Clough, 2005] folgt bspw. auf einen Thesaurusabgleich die Disambiguierung von Termen mit Treffern in mehreren Listen über den grammatikalischen Kontext. Eine Evaluation freizugänglicher NER-Systeme in Mandl et al. [2005] zeigte recht bescheidene Ergebnisse auf.

Hierarchische Gazetteers können im nächsten Schritt auch genutzt werden, um die Anfrage um alternative Namen und Übersetzungen sowie untergeordnete Länder, Städte, etc. zu expandieren. Speziell für die Problematik fehlender Gazetteer-Einträge für ungenaue Regionen (z.B. *Nordschottland*) stellen Clough et al. [2005a] einen Ansatz vor, durch eine Webanfrage mit Trigger-Phrasen (z.B. „*A is a town in x*“) diese über die Termhäufigkeiten der zurückgelieferten Städtenamen zu modellieren und ggfs. zu expandieren.

Eine wichtige Rolle kommt Gazetteers auch im Prozess der rein geographischen Disambiguierung zu. So kann bei der Mehrdeutigkeitsauflösung von Frankfurt der Kontext nach anderen geographischen Eigennamen durchsucht werden, die sich in ihren Hierarchien überlappen (*World → Europe → Hessen → Frankfurt*). Auf ähnliche Weise nutzen Amitay et al. [2004] einen Gazetteer, um den geographischen Schwerpunkt eines Dokuments durch Zuweisung von *Parent-Regions* zu ermitteln. Sind keine kontextuellen Hinweise vorhanden, kann als Standard eine Entscheidung stets für den Kandidaten mit der kürzesten Hierarchie fallen, da eine größere Bekanntheit angenommen werden kann. [Clough, 2005]

Die Eindeutigkeit der erkannten geographischen Eigennamen ist Grundlage für das *Geo-Coding*, der Zuweisung von Geodaten (Koordinaten) zu den Referenzen. Dieser Verarbeitungsschritt, für den abermals Gazetteers als externe Wissensressourcen nötig sind, ermöglicht räumliche Indexierungs- und Retrievalmethoden. So kann bspw. die räumliche Distanz im Koordinatensystem oder der Grad an Überlappung der Minimum Bounding Rectangles zweier Regionen als Maß für die Relevanz herangezogen werden [Frontiera und Larson, 2004; Chaves et al., 2005].

Ziel von GeoCLEF ist es, Ergebnisse aus den Arbeiten zum Geo-Parsing und Geo-Coding bzw. -Matching unter Beachtung der Mehrsprachigkeit zusammenzuführen.

3 GeoCLEF

2005 fand mit GeoCLEF erstmals ein geographischer Track innerhalb der CLEF-Initiative statt. Die Ergebnisse dieses Pilotprojektes zeigten nicht nur die prinzipielle Durchführbarkeit eines solchen Tracks, sondern auch das große Interesse an dem Themenfeld und vor allem den erheblichen Forschungsbedarf.

3.1 Ziele und Methoden

Entsprechend der Infrastruktur von CLEF werden auch in GeoCLEF Techniken und Systeme anhand einheitlicher Anfragen (*Topics*) gegen ein mehrsprachiges Korpus aus Zeitungsartikeln und Meldungen von Nachrichtenagenturen anhand von anschließenden Relevanzbewertungen miteinander verglichen. Die Topics werden dabei parallel für unterschiedliche Sprachen entwickelt, indem möglichst realistische Benutzeranfragen modelliert, recherchiert und dann von Muttersprachlern in die jeweilige Sprache übersetzt werden.

Im Jahre 2005 konnte in GeoCLEF mit 25 Topics in den Ausgangssprachen Deutsch, Englisch, Spanisch und Portugiesisch – monolingual oder bilingual – in der deutschen oder englischen CLEF-Kollektion experimentiert werden. 2006 waren auch monolinguale Versuche (*Runs*) gegen Kollektionen in Spanisch und Portugiesisch möglich und es kamen Topics in Japanisch hinzu. Die nachfolgende Abbildung zeigt ein Topic aus dem diesjährigen Track:

```
<top>
 <num>GC036</num>
 <DE-title>Automobilindustrie rund um das
  Japanische Meer</DE-title>
 <DE-desc>Küstenstädte am Japanischen Meer mit
  Automobilindustrie oder -werken</DE-desc>
 <DE-narr>Relevante Dokumente berichten von
  Automobilindustrie oder -werken in Städten an der
  Küste des Japanischen Meeres (auch Ostmeer (von
  Korea) genannt), einschließlich wirtschaftlicher oder
  sozialer Ereignisse wie geplante Joint Ventures oder
  Streiks. Neben Japan grenzen auch die Länder
  Nordkorea, Südkorea und Russland an das Japanische
  Meer.</DE-narr>
</top>
```

Abb. 1: Beispiel für ein GeoCLEF-Topic

Während im letzten Jahr die Teilnehmer für ihre Experimente Informationen aus den Tags Title (*title*) und Description (*desc*) sowie zusätzlich aus Concept-Tags, in denen die geographischen Entitäten extrahiert vorlagen, verwendeten, galt es 2006, diese Eigennamen automatisch zu erkennen. Speziell sollte die Nützlichkeit zusätzlicher geographischer Information zur Expansion der Anfrage evaluiert werden. Daher waren sowohl Runs mit den Feldern Title, Description zu absolvieren als auch Runs, die zudem unter den gleichen Parametern den Text des Feldes Narrative (*narr*) – meist beinhaltete dieser die Namen der (Bundes-)Länder einer Region – nutzen. Einige Topics zielten weniger auf eine solche Erweiterung als vielmehr auf geeignete Retrievaltechniken ab *(„Städte im Umkreis von 100 km um Frankfurt“)*.

Aus organisatorischer Sicht erwies es sich dabei als durchaus schwierig, anhand der gegebenen Kollektion Topics zu entwickeln, die einerseits realistische Benutzerbedürfnisse abbilden, zugleich geographisch interessant sind und in allen Sprachkollektionen Treffer vorweisen. So scheinen in diesem Zusammenhang bestimmte Mechanismen der Nachrichtenselektion wie bspw. geographische Nähe, sprachliche oder traditionelle Beziehung und wirtschaftliche Bedeutung zu beeinflussen, ob und

wie oft über ein Ereignis einer bestimmten Region in den Zeitungen einer Kollektion berichtet wird. In allen Kollektionen vertretene geographische Referenzen sind daher zumeist bekannte Regionen bspw. Länder [Clough et al., 2005b]. Trotz Beachtung dieser Problematik bei der Genierung der Topics waren in der Relevanzbewertung 2006 zu einigen Topics in bestimmten Sprachen kaum relevante Dokumente vorhanden.

Da die Ergebnisse aller Teilnehmer für GeoCLEF 2006 aktuell noch nicht vorliegen, sollen kurz die wesentlichen Ergebnisse des Tracks von 2005 genannt werden.

3.2 Ergebnisse und Erfahrungen 2005

Bereits 2005 führten nicht vorhandene relevante Dokumente in der deutschen Kollektion dazu, dass die durchschnittlichen Precison-Werte (MAP)[2] für die mono- und bilingualen Runs ins Deutsche sehr schlecht ausfielen, also die Aufgabe unbeabsichtigt schwieriger war. Die meisten Experimente wurden monolingual Englisch eingereicht und erreichten MAPs von Minimum 0.1464 bis Maximum 0.3936. Hingegen reichten MAPs für monolingual Deutsch lediglich von 0.0535 bis 0.2042. Die insgesamt doch bescheidenen Werte deuten auf die Eigenheiten des GIR und die Notwendigkeit von speziellen, geeigneten Methoden hin. [Clough et al., 2005b]

Dabei nutzten die Teilnehmer verschiedenste Techniken, von ‚einfachen' IR-Techniken ohne jeglichen geographischen Ansatz hin zu Matching mittels Geodaten und Verfahren des Natural Language Processing (NLP), um geographische Hinweise aus Anfrage und Dokument zu extrahieren. Die Erkennung von geographischen Named Entities (NEs) mithilfe verschiedener Techniken wurde jedoch von den meisten Teilnehmern angestrebt. Hauptkritik einiger Teilnehmer war, dass die Topics 2005 üblichen adhoc-Anfragen zu ähnlich waren. Eine reine Keyword-Suche schnitt demnach kaum schlechter ab als elaborierte geographische Methoden. [Clough et al., 2005b]

Die besten Ergebnisse gelangen der Universität Berkeley [Gey und Petras, 2005] für monolingual Englisch durch ein austariertes Blind Relevance Feedback (BRF), auch wenn für einige wenige Topics die Werte durch das Hinzufügen von 30 neuen Termen aus den 5 bestgerankten Dokumenten sanken. Während die Verbesserung durch das BRF im Englischen nur moderat war, zeigte sich im Deutschen eine beachtliche Steigerung der MAP um bis zu 72% des Ausgangswertes.

Manuelle Experimente zur Expansion von geographischen Referenzen (bspw. Europa wurde manuell angereichert um die zugehörigen Ländernamen), brachten überraschenderweise schlechte Ergebnisse für Deutsch und Englisch. Wegen dieser sinkenden Precisionwerte folgern Gey und Petras [2005], dass automatische Expansion mittels geographischer Thesauri nicht sehr viel versprechend ist, es sei denn, es wird ein Boolescher Ansatz – UND-Verknüpfung von inhaltlichem Konzept und geographischer Referenz – verfolgt [auch Larson, 2005]. Diesen Ansatz hat die Universität Hildesheim bei ihrer Teilnahme in GeoCLEF 2006 fokussiert.

[2] Die Mean Average Precison (MAP) gibt die durchschnittliche Precision über festgelegte Recall-Level an. Hier für einen Run gemittelt über alle Topics.

4 Eigene Experimente in GeoCLEF 2006

Nachdem GeoCLEF 2005 im Hinblick auf die Expansion geographischer Eigennamen negative oder zumindest mehrdeutige Ergebnisse gezeigt hatte, sollten in unseren Experimenten die Gesichtspunkte Boolesches Retrieval und Gewichtung bei der Nutzung von zusätzlichen geographischen Informationen untersucht werden. Zur Expansion dienen die Informationen des Narratives der Topics. Die automatische Expansion mittels Gazetteer und Wikipedia für nicht in Gazetteers enthaltene geographische Referenzen wurde noch nicht in die eingereichten Versuche eingebunden, wird aber im Anschluss an die Beschreibung der aktuellen Experimente skizziert.

4.1 Beschreibung der Experimente

Aufbauend auf die Versuche von Mandl et al. [2006] basieren die grundsätzlichen Retrievalfunktionen auf dem Lucene-Paket[3], mit dem Lucene-Stemmer für das Deutsche und Snowball-Stemmer für das Englische. Die Übersetzung der Topics für die bilingualen Versuche beruht auf Babelfish[4], Linguatec[5] und FreeTranslation[6]. Durch die Kombination mehrerer Übersetzer sollen einzelne Fehler abgemildert bzw. zusätzlich Synonyme in der Zielsprache hinzugenommen werden.

Für die Erkennung von Eigennamen wird das maschinelle Lerntool Lingpipe[7] eingesetzt, welches anhand eines trainierten Modells Eigennamen identifiziert und in die Kategorien PERSON, LOCATION, ORGANISATION und MISC klassifiziert. Als Modell diente für das Englische das mitgelieferte, an einem englischen Nachrichten-Korpus trainierte, News-Modell. Da für die deutsche Sprache kein Modell angeboten wird, wurde auf das von Mandl et al. [2006] trainierte zurückgegriffen.

Auch der Index der Kollektion enthält Felder für diese Klassen von Eigennamen, so dass innerhalb des BRF-Prozesses gezielt (geographische) Eigennamen bevorzugt zur Anfragereformulierung herangezogen werden können. Als Idee hinter diesem Schritt steht, dass somit aus den topgerankten Dokumenten zu einer geographischen Suche weitere Eigennamen von zugehörigen Städten, Regionen oder Ländern gefunden werden und in die Anfrage eingehen können. Derart ermittelte Geo-Entitäten werden der Anfrage dabei über ein Boolesches UND zugefügt, um auch das inhaltliche Kriterium zu erfüllen. Auch in Versuchen ohne BRF wurden daher erkannte Geo-NEs aus den Feldern Title, Description und Narrative über UND mit dem Inhalt verknüpft. Die generellen Verarbeitungsschritte können daher wie folgt schematisch dargestellt werden, wobei je nach Run nur bestimmte Schritte ausgeführt bzw. Parameter geändert wurden:

Topic → (Übersetzung) → (NER und Gewichtung) → Stoppworttilgung → Stemming → Anfrage Boolesches UND vs. ODER → (BRF ggfs. mit Gewichtung von geographischen Eigennamen)

Abb. 2: Verarbeitungsschritte des Systems

[3] http://lucene.apache.org/java/docs/
[4] http://babelfish.altavista.com/
[5] http://www.linguatec.de/onlineservices/pt
[6] http://www.freetranslation.com/
[7] http://www.alias-i.com/lingpipe/

Sprache	Feld	NEs	BRF	Query	MAP
En	title, desc	-	-	ODER	0,1676
En	title, desc, narr	-	-	ODER	0,1747
En	title, desc	gewichtet	5 docs, 25 terms, NEs + GeoNEs gewichtet	UND	0,1166
En	title, desc, narr	gewichtet	5 docs, 25 terms, NEs + GeoNEs gewichtet	UND	0,1213
En	title, desc	-	5 docs, 20 terms, GeoNEs gewichtet	ODER	0,1875
De	title, desc	-	5 docs, 25 terms	ODER	0,1558
De	title, desc, narr	-	5 docs, 25 terms	ODER	0,1601
De	title, desc	gewichtet	5 docs, 25 terms, NEs + GeoNEs gewichtet	UND	0,1214
De	title, desc, narr	gewichtet	5 docs, 25 terms, NEs + GeoNEs gewichtet	UND	0,1134
De → En	title, desc	-	-	ODER	0,1504
De → En	title, desc, narr	-	-	ODER	0,1903
De → En	title, desc	gewichtet	5 docs, 25 terms, NEs + GeoNEs gewichtet	UND	0,1456
De → En	title, desc, narr	gewichtet	5 docs, 25 terms, NEs + GeoNEs gewichtet	UND	0,1565
De → En	title, desc	-	5 docs, 20 terms, GeoNEs gewichtet	ODER	0,1603
En → De	title, desc	-	5 docs, 25 terms	ODER	0,1186
En → De	title, desc, narr	-	5 docs, 25 terms	ODER	0,1315
En → De	title, desc	gewichtet	5 docs, 25 terms, NEs + GeoNEs gewichtet	UND	0,0969
En → De	title, desc, narr	gewichtet	5 docs, 25 terms, NEs + GeoNEs gewichtet	UND	0,1046

Abb. 3: MAPs der einzelnen Hildesheimer Runs

So wurden für die Aufgaben monolingual Englisch und bilingual Deutsch → Englisch Versuche eingereicht, in denen als Base Run weder NEs erkannt noch BRF durchgeführt wurde, sondern nur die genutzten Felder variiert wurden. In zwei weiteren Runs wurden darüber hinaus NEs erkannt und stärker gewichtet, sowie im BRF das Hinzufügen von geographischen Eigennamen forciert. In einem fünften Versuch – ohne die Informationen des Feldes Narrative – wurde nur auf Geo-NEs im BRF getestet.

Die monolingualen Versuche Deutsch und die bilingualen Versuche Englisch → Deutsch waren analog konzipiert, jedoch entfielen die Runs ohne BRF, da diese im Training an den GeoCLEF-Daten von 2005 die schlechtesten Ergebnisse lieferten.

An dieser Stelle können nur erste Folgerungen aus den Ergebnissen der Teilnahme dargestellt werden, da bislang die eingesetzten Techniken der anderen Teilnehmer und deren Abschneiden[8] unklar sind sowie eine genaue Analyse der Ergebnisse für die einzelnen Topics noch aussteht.

4.2 Erste Ergebnisse

Die Ergebnisse sind über alle Versuche hinweg keineswegs zufrieden stellend, liegen jedoch im Durchschnitt der Experimente aller Teilnehmer. So zeigt sich auch in GeoCLEF 2006, dass die Aufgabe des GIR nicht trivial ist, denn auch die besten Teilnehmer konnten nur eine MAP von 0,3223 für monolingual Englisch, 0,2229 für monolingual Deutsch und 0,1682 für bilingual Englisch → Deutsch verzeichnen.

Trotz der geringen Veränderungen der MAP scheint in unseren Versuchen kein negativer Effekt einer Expansion um untergeordnete geographische Entitäten spürbar – sowohl bei Boolescher Verknüpfung als auch bei einfacher Gewichtung. Unter gleichen Parametern verbesserte sich stets die Precision unter Hinzunahme des Feldes Narrative. Im Falle des besten Laufes für bilingual Deutsch → Englisch dürfte der Anstieg der MAP auf 0,1903 jedoch daran liegen, dass die Erweiterung um den Begriff *Indonesien* das einzige relevante Dokument zu diesem Topic zurückliefern konnte.

Die im internen Vergleich recht guten Ergebnisse der Höhergewichtung von Geo-Entitäten innerhalb des BRF, könnten darauf hindeuten, dass die Art der zusätzlichen geographischen Information in den Narratives nicht unbedingt die für die Expansion geeigneteste sein könnte.

Die Verbesserung der Werte für die Runs mit Boolescher UND-Verknüpfung in der bilingualen Bedingung Deutsch → Englisch ggü. monolingual Englisch durch den kombinierten Einsatz mehrerer Übersetzer ist interessant. Zu prüfen ist, ob dies auf dadurch gewonnene Synonyme oder eine wegen Varianten in der Wortstellung erleichterte NER zurückzuführen ist. Für die Versuche mit Gewichtung und boolescher Verknüpfung von (Geo-)NEs ist eine weitere Analyse der Performanz der NER nötig, bevor Schlussfolgerungen gezogen werden können. Eine erste Durchsicht zeigt, dass einige NEs gerade im Deutschen nicht korrekt erkannt wurden.

Eine genaue Analyse sowohl der Performanz über die unterschiedlichen Topics mit ihren jeweils eigenen Anforderungen (die MAP-Werte für die einzelnen Topics variieren stark) und die Isolierung der Einzeleffekte der eingesetzten Verfahren ist der nächste Schritt, um erfolgreiche(re) Techniken zu ermitteln.

Ausblick

Die niedrigen MAP-Werte zeigen, dass im Anschluss an die GeoCLEF-Teilnahme 2006 besonders im Hinblick auf die Rolle von (geographischen) NEs eine genaue Analyse durchgeführt werden muss. Für weitere Versuche ist daher als Grundlage die Verbesserung der NER durch Fusion verschiedener Ansätze geplant.

Die automatische Expansion mithilfe eines Gazetteers wird umgesetzt, wobei den Heuristiken für eine erfolgreiche Anreicherung spezielle Beachtung geschenkt werden soll. So scheinen Bekanntheit, (wirtschaftliche) Relevanz, geographische Nähe einer Region zu beein-

[8] Es liegen für jede Aufgabe der Mittelwert aller eingereichten Experimente, der beste und schlechteste Wert vor. Diese sind der Email-Kommunikation mit Ray Larson (an die GeoCLEF-Organisatoren) vom 27.07.2006 entnommen.

flussen, ob und wie eine Expansion nötig oder sogar sinnvoll ist. Nicht nur zur Weiterverfolgung der Idee, im BRF für die Expansion geeignete andere geographische Eigennamen zu finden, sind zudem Techniken zur Disambiguierung und zur Ermittlung des geographischen Schwerpunktes eines Dokumentes zu integrieren.

Eine angedachte Anbindung von Wikipedia als externer Wissensressource, durch welche Referenzen ohne Gazetteer-Eintrag (z.B. *Norddeutschland, Warschauer Pakt*) geographisch expandierbar werden sollen, basiert ganz wesentlich darauf. Darüber hinaus ist im Hinblick auf Synonyme eine Erweiterung des Systems z.B. um den Wortschatz Leipzig geplant.

Literatur

[Amitay et al., 2004] Einat Amitay, Nadav Har'El, Ron Sivan und Aya Soffer. Web-a-Where: Geotagging Web content. In *SIGIR 2004: Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Seiten 273-280, Sheffield, Großbritannien, Juli 2004, ACM.

[Chaves et al., 2005] Marcirio Silveira Chaves, Bruno Martins und Mário J. Silva. Challenges and Resources for Evaluating Geographical IR. In *Proceedings of the 2nd International Workshop on Geographic Information Retrieval*, CKIM 2005, Seiten 65-69, Bremen, Deutschland, November 2005.

[Clough, 2005] Paul Clough. Extracting Metadata for Spatially-Aware Information Retrieval on the Internet. In *Proceedings of the 2nd International Workshop on Geographic Information Retrieval*, @CKIM 2005, Seiten 25-30, Bremen, Deutschland, November 2005.

[Clough et al. 2005a] Paul Clough, Hideo Joho und Ross Purves. Identifying imprecise regions for geographic information retrieval using the web. In *Proceedings of the GIS RESEARCH UK 13th Annual Conference*, Seiten 313-318, Glasgow, Großbritannien, 2005.

[Clough et al., 2005b] Paul Clough, Frederic Gey, Hideo Joho, Ray Larson, Vivien Petras und Mark Sanderson. GeoCLEF: the CLEF 2005 Cross-Language Geographic Information Retrieval Track. In *Working Notes for the CLEF 2005 Workshop*, Wien, Österreich, September 2005, http://www.clef-campaign.org/2005/ working_notes/.

[Frontiera und Larson, 2004] Patricia Frontiera und Ray R. Larson. Evaluation and Usability – Spatial Ranking Methods for Geographic Information Retrieval (GIR) in Digital Libraries. In *Proceedings of the 8th European Conference on Research and Advanced Technology for Digital Libraries*, ECDL 2004, Seiten 45-56, Bath, Großbritannien, September 2004, Lecture Notes in Computer Science 3232, Springer.

[Gey und Petras, 2005] Frederic Gey und Vivien Petras. Berkeley2 at GeoCLEF: Cross-Language Geographic Information Retrieval of German and English Documents. In *Working Notes for the CLEF 2005 Workshop*, Wien, Österreich, September 2005, http://www.clef-campaign.org/2005/ working_notes/.

[Goodchild et al., 2005] Michael Goodchild, Paul A. Longley, David J. Maguire und David W. Rhind. *Geographic Information Systems and Science*. John Wiley and Sons, Chichester, 2. aktualisierte Auflage 2005.

[Jones und Purves, 2004] Chris Jones und Ross Purves. *Workshop on Geographic Information Retrieval, SIGIR 2004*. SIGIR Forum, 38(2): 53-56, Dezember 2004.

[Jones und Purves, 2005] Chris Jones und Ross Purves, Herausgeber. *Proceedings of the 2005 Workshop on Geographic Information Retrieval*, GIR 2005, Bremen, Deutschland, November 2005, ACM.

[Larson, 2005] Ray R. Larson. Chesire II at GeoCLEF: Fusion and Query Expansion for GIR. In *Working Notes for the CLEF 2005 Workshop*, Wien, Österreich, September 2005, http://www.clef-campaign.org/2005/ working_notes/.

[Mandl et al., 2005] Thomas Mandl, René Schneider, Pia Schnetzler und Christa Womser-Hacker. Evaluierung von Systemen für die Eigennamenerkennung im cross-lingualen Information Retrieval. In B. Fisseni, Hans-Christian Schmitz, Bernhard Schröder und Petra Wagner, Herausgeber. *Sprachtechnologie, mobile Kommunikation und linguistische Ressourcen: Beiträge zur GLDV Tagung 2005 in Bonn*, [Sprache, Sprechen und Computer/ Computer Studies in Language and Speech 8], Seiten 145-157, Peter-Lang, Frankfurt/Main et al., 2005.

[Mandl et al., 2006] Thomas Mandl, René Schneider und Robert Strötgen. A Fast Forward Approach to Cross-lingual Question Answering for English and German. In Fredric C. Gey, Julio Gonzalo, Gareth J.F Jones, Michael Kluck, Bernardo Magnini, Henning Müller, Carol Peters and Maarten de Rijke, Herausgeber. *Accessing Multilingual Information Repositories: 6th Workshop of the Cross-Language Evaluation Forum*, CLEF 2005, Wien, Österreich, Revised Selected Papers. Lecture Notes in Computer Science 4022, Springer, 2006.

[Womser-Hacker 2006] Christa Womser-Hacker. Zur Rolle von Eigennamen im Cross-Language Information Retrieval. In Ilse Harms, Heinz-Dirk Luckhardt und Hans W. Giessen, Herausgeber. *Information und Sprache. Beiträge zu Informationswissenschaft, Computerlinguistik, Bibliothekswesen und verwandten Fächern*. Festschrift für Harald H. Zimmermann zum 65. Geburtstag, K.G. Saur, München, 2006.

Entwicklung eines dynamischen Entry Vocabulary Moduls für die Stiftung Wissenschaft und Politik

Benjamin Berghaus, Michael Kluck und Thomas Mandl

Universität Hildesheim
Informationswissenschaft
Marienburger Straße 22
31134 Hildesheim
benjamin.berghaus, mandl@uni-hildesheim.de

Stiftung Wissenschaft und Politik
Fachinformationsbereich
Ludwigkirchplatz 3-4
10719 Berlin
michael.kluck@swp-berlin.org

Abstract

Nicht übereinstimmendes Vokabular zwischen Anfrage und Dokumenten stellt ein Hauptproblem im Information Retrieval dar. Das Entry Vocabulary Modul hat sich in den letzten Jahren als Lösung hierfür etabliert. In diesem Beitrag wird ein dynamisches Entry Vocabulary Modul vorgestellt, das für einen Datenbestand mit mehreren inhaltsbezogenen Feldern in einem mehrstufigen Verfahren abhängig von Zwischenergebnissen die Anfrage erweitert. Das entwickelte System wurde anhand eines mehrsprachigen Datenbestands von rund 600.000 Fachtexten evaluiert und führte zu positiven Ergebnissen.

1 Einleitung

1.1 Die zentrale Frage des Vokabulars

Bei der Verbalisierung von Informationen wird die Nachricht mit Hilfe eines Vokabulars kodiert. Da es aufgrund verschiedener Sprachen und spezialisierter Fachsprachen viele verschiedene Vokabulare gibt, ist es essentiell, dass, sofern die Information ausgetauscht werden soll, sowohl der Sender als auch der Empfänger der Information das selbe Vokabular beherrschen und den Sinn der verbalen Abbildung der Information verstehen können. Ist das in der Kommunikation verwendete Vokabular einem der Kommunikationspartner unbekannt, wird der Austausch von Informationen nahezu unmöglich.

Bezogen auf die Welt des Information Retrieval ergibt sich in diesem Kontext ein ähnliches Problem. Je nach Aufgabe und Einsatzgebiet des IR-Systems variiert die Art der Datengrundlage und der in der Datengrundlage verzeichneten Informationen drastisch. Handelt es sich um eine hochspezialisierte Datenbank, beispielsweise die in [Gey *et al.*, 2001] herangezogene Datenbank von amerikanischen Im- und Exportstatistiken, so wird auch die Information in der Datengrundlage entsprechend in einem spezialisierten Vokabular kodiert sein. Im Falle der Außenhandelsstatistiken lässt sich beispielsweise nicht erfolgreich mit dem Begriff „automobile“ suchen - der entsprechende Begriff lautet in dem Zielvokabular des IR-Systems „Pass Mtr Veh“, was einen Abkürzung für „Passenger Motor Vehicle“ darstellt.

Üblicherweise würde eine solche Datengrundlage nur für die entsprechenden Spezialisten interessant sein, die mit dem entsprechenden Vokabular der Datengrundlage vertraut sein müssen. Allerdings kann es sein, dass auch eine solche Datenbank öffentlich zugänglich gemacht wird und somit auch Nutzern durchsucht wird, die sich des speziellen Vokabulars nicht bewusst sind. Hierbei entsteht, wie im oben beschriebenen Beispiel, die Situation, dass Nutzer, die des Vokabulars des Systems nicht mächtig sind, das System nicht auf eine zielführende Art und Weise bedienen können - nicht nur, weil sie das Ergebnis des Retrievalprozess eventuell nicht interpretieren, sondern weil auf das System unvorbereitete Nutzer ohnehin kaum eine sinnvolle Anfrage formulieren können.

Für die Lösung dieser Probleme der semantischen Heterogenität in Metadatensystemen existieren mehrere Ansätze, vgl. [Hellweg et al., 2001]. Um eine Brücke zwischen dem spezialisierten, kontrollierten Vokabular einer spezialisierten Datengrundlage und dem mehr oder weniger freien Vokabular eines untrainierten Nutzers zu bauen, wurden in den letzten Jahren zunehmend sogenannte Entry Vocabulary Module eingesetzt, vgl. [Buckland *et al.*, 1999]. Diese Module bestehen üblicherweise aus einem Entry Vocabulary Index, der die Beziehungen zwischen Termen des Freitexts und Deskriptoren oder Klassifikationsangaben auf Basis von Wahrscheinlichkeiten abbildet und einer Schnittstelle, die geeignete kontrollierte Vokabeln vorschlagen kann, vgl. [Norgard, 1998]. Auf diese Art und Weise kann eine Anfrage, die frei formuliert wurde, auf das eventuell kontrollierte Vokabular der Datengrundlage übersetzt oder um verwandte Terme oder Phrasen ergänzt werden.

Eine weitere, interessante Anwendungsmöglichkeit besteht außerdem darin, nicht nur einen „vertikalen“ Vokabularunterschied zu nivellieren, sondern auch einen „horizontalen“: Während der Unterschied zwischen spezialisiertem und freien Vokabular eindeutig ist, ist auch der Unterschied zwischen dem Vokabular verschiedener Sprachen - also der mehrsprachige Aspekt - durch den Einsatz von EVMs gegebenenfalls zu überbrücken. In [Petras, 2005] wurde bereits belegt, dass mehrsprachiges Information Retrieval durch den Einsatz von Metadaten verbessert werden kann: Petras wendete das EVM für die mit Thesaurustermen indexierte Fachdatenbank GIRT (German Indexing und Retrieval Testdatabase) an. GIRT wird zur Evaluie-

rung von mehrsprachigen Information Retrieval Verfahren im Rahmen des Cross Language Evaluation Forum[1] eingesetzt, vgl. [Mandl, 2006].

Neben der Übersetzung der eingegebenen Suchanfrage ist darüber hinaus deren Ergänzung um verwandte Terme und Phrasen möglich - hierbei steht nicht unmittelbar im Vordergrund, vollkommen verschiedene Vokabulare zu verknüpfen und somit überhaupt eine Suche möglich zu machen, sondern vielmehr die Retrievalleistung einer Anfrage zu verstärken.

1.2 Stiftung Wissenschaft und Politik

Die zugrundeliegenden Daten, für die das Information Retrieval System entwickelt und auf denen es evaluiert wird, werden von der Stiftung Wissenschaft und Politik, Berlin, (SWP) zur Verfügung gestellt. Es handelt sich um einen umfassenden Auszug aus der Literaturdatenbank des Fachinformationsverbunds für Internationale Beziehungen und Länderkunde (FIV). Die Literaturdatenbank ist beispielsweise über [Virtuelle Fachbibliothek Politikwissenschaften, 2006] zu nutzen, umfassende Informationen finden sich unter [Fachinformationsverbund IBLK, 2006].

Die SWP wurde 1962 bei München gegründet, hat ihren Hauptsitz seit 2001 in Berlin und ist ein deutsches Institut im Forschungsfeld der Außen- und Sicherheitspolitischen Fragen. Wichtigste Auftraggeber der SWP sind der Deutsche Bundestag, die Bundesregierung und die Ministerien, vorrangig hierbei das Auswärtigen Amt und das Verteidigungsministerium.

Die Datenbank des FIV umfasst rund 600.000 Einträge, die sich zum Großteil mit den Themengebieten Staat- und Gesellschaft, nationale und internationale Wirtschaft, Internationale Politik und Sicherheit befassen. Geographisch beziehen sich mehr als die Hälfte der verzeichneten Dokumente auf Europa, europäische Organisationen und die NATO. Weitere wichtige und berücksichtigte geographische Regionen schließen Afrika und den Nahen Osten, Nord- und Südamerika und Asien und Ozeanien neben anderen mit ein.

Die Datengrundlage verzeichnet zu 65% Bücher und Paper, zu 25% monographische Veröffentlichungen und zu jeweils 5% Periodika und Jahrbücher sowie Amtliche Veröffentlichungen. 24% der zu Verfügung gestellten Dokumente beinhalten ein Abstract, die restlichen Dokumente verfügen ausschließlich über einen Titel und ggf. diverse Deskriptoren als Metainformationen. Sprachlich dominieren die englischen Dokumente mit 51% die Datengrundlage. Deutsche Dokumente machen 28% des Umfangs aus, französische rund 11% und spanische 5%, während der Rest der Dokumente in sonstigen Sprachen verfasst ist. Die Deskriptoren sind insgesamt auf Deutsch verfasst. [Stiftung Wissenschaft und Politik, 2006]

2 Konzeption eines Entry Vocabulary Moduls

Das Konzept der Entry Vocabulary Modul wurde unter anderem in [Gey *et al.*, 2001] vorgestellt. In dieser Arbeit wurde die vier zentralen Komponenten wie folgt bezeichnet:

- eine ausreichend große Datengrundlage zum Trainieren des Entry Vocabulary Index
- ein Part-of-Speech-Tagger, der Substantive aus Dokumenttexten extrahiert
- ein Algorithmus, der die Beziehung zweier Begriffe anhand der Wahrscheinlichkeit ihrer Kooexistenz in einem Dokument errechnet
- das grundlegende Retrievalsystem, das die Suchanfrage entgegen nimmt und die Ergebnisse auflistet

Das von Gey vorgestellte System ist ein globaler (d.h. auf den gesamten Datenbestand bezogener) Ansatz zur Konstruktion eines Entry Vocabulary Index, in dem Terme des freien Vokabulars mit den kontrollierten Vokabeln der Metadaten auf Basis von probablistischen Untersuchungen verknüpft werden. Diese Verknüpfung basiert auf einer statistischen Analyse der Koexistenz von Termen und vergebenen Metainformationen für ein gegebenes Dokument.

Diese Entwicklung eines zusätzlichen Datenkonstrukts, des Entry Vocabulary Index, ist dabei grundsätzlich nicht zwingend erforderlich. Eine dynamische, lokale (d.h. auf eine Gruppe von potentiell relevanten Dokumenten angewendete) Lösung minimiert den Aufwand der Pflege und der stetigen Aktualisierung eines zusätzlichen Datenbestands. Da die Datenbasis des FIV stetig wächst und aktuellere Themen, mit denen sich die Beiträge befassen, ebenfalls in vielen Fällen neues, freies Vokabular mit sich bringen, wäre eine regelmäßige Neuberechnung notwendig. Schließlich ist zu erwarten, dass ein großes Interesse daran besteht, auch die neuesten Ergänzungen der Datenbank effektiv aufzufinden. Darüber hinaus wird in [Xu, Croft, 2006] beschrieben, dass zumindest für Terme aus dem Freitext eines Datenbestands ein lokaler, dem Relevance Feedback verwandter Ansatz nicht schlechter geeignet sein muss als eine globale Berechnung.

Ein weiteres Argument gegen eine globale Auswertung des paarweisen Auftretens von Termen des freien Vokabulars und Deskriptoren ist die begrenzte Anzahl von Dokumenten mit Freitexten in Form von Zusammenfassungen (rund ein Viertel aller Dokumente). Eine entsprechende Auswertung würde die Deskriptoren der mit Zusammenfassungen ausgestatteten Dokumente voraussichtlich anders in Relation zu den Termen des Freitextes stellen als es bei den Dokumenten der Fall wäre, in denen die Deskriptoren nur mit den Termen der Titel in Relation gebracht werden könnten. Würde sich eine solche globale Analyse nur auf die Zusammenfassungen beziehen, könnte nur ein Viertel der Dokumente entsprechend ausgewertet werden. Bei einer dynamischen Lösung können dagegen auch Metadaten zueinander in Relation gebracht werden, indem in einem Suchprozess anhand der bereits extrahierten Metadaten gesucht wird und weitere Deskriptoren extrahiert werden können, die besonders häufig in den gefundenen Dokumenten auftreten. Ein solcher Ansatz würde bei dem Datenbestand des FIV so gut wie alle Dokumente in einer Auswertung mit einbeziehen, da nahezu alle Dokumente mit Deskriptoren erschlossen sind.

Für das entwickelte Information Retrieval System wurden mehrere Open Source Bibliotheken verwendet. Fundament hierfür ist die Klassenbibliothek Apache Lucene in Version 1.9.1, zum Parsen und Indexieren der XML-Dateien wird Jakarta Commons Digester in Version 1.7 verwendet. Basis für die Systementwicklung bilden die an der Universität Hildesheim entwickelten Komponenten, die auf Lucene basieren und im Rahmen von CLEF erfolgreich für mehrsprachiges Retrieval eingesetzt wurden, vgl. [Hackl, Mandl, Womser-Hacker, 2005] und [Hackl, Mandl, 2006]. Zur Evaluierung des Systems kommen sowohl ein Relevanzbewertungsprogramm des Informationszentrum Sozialwissenschaften in Bonn als auch das an der Universität

[1] siehe auch: http://www.clef-campaign.org

Hildesheim nach Java portierte Programm Trec_Eval (im Original von Gerard Salton und Chris Buckley) in Version 0.7 zum Einsatz.

3 Die Indexierung

3.1 Datendateien

Die Indexierung der insgesamt 600 XML-Datendateien des Fachinformationsverbunds für Internationale Beziehungen und Länderkunde geschieht mit Hilfe des XML-Parser Jakarta Commons Digester. Es wird auf die Felder

- *file*
  (Dateiname der Datendatei)
- *id*
  (Collection/Publication/Identifier/Text)
- *title*
  (Collection/Publication/Text)
- *abstract*
  (Collection/Publication/Description/Text)
- *subject*
  (Collection/Publication/Subject/Text)
- *language*
  (Collection/Publication/Language/Text)
- *classification*
  (Collection/Publication/Classification/Text)
- *geo*
  (Collection/Publication/GeographicCoverage/Text)
- *temp*
  (Collection/Publication/TemporalCoverage/Text)

indexiert, wobei während der Indexierung der Lucene StandardAnalyzer verwendet wird. Außerdem wird eine manuell an den Datenbestand angepasste Stoppwortliste eingesetzt, die aus den besonders hochfrequent auftretenden Termen des Index mit Hilfe der Lucene Index Toolbox Luke in Version 0.6 entwickelt wurde und auch in einigen Termen an die Anfragen angepasst wurde.

Bei mehreren Einträgen in der gleichen XML-Elementebene eines Dokuments wird das entsprechende Feld des Lucene Index um alle weiteren Einträge erweitert.

3.2 Thesaurusdateien

Auch das Parsing der neun XML-Dateien des Thesaurus wird mit Hilfe von Digester realisiert, es wird auf die folgenden Felder indexiert:

- *subject*
  (Collection/Subject/Text)
- *translations*
  (Collection/Subject/Subject/Text, Typerkennung)
- *group*
  (Collection/Subject/Subject/Text, Typerkennung)
- *subGroup*
  (Collection/Subject/Subject/Text, Typerkennung)
- *connectedTerms*
  (Collection/Subject/Subject/Text, Typerkennung)

Aufgrund der im Vergleich zu den GIRT-Daten komplizierter zu parsenden XML-Architektur sowohl der Daten- als auch der Thesaurusdateien, wird beim Indexieren der Elemente unter Collection/Subject/Subject/Text im Thesaurus eine zusätzliche Methode verwendet, die auf Basis des Eintrags in dem Element entscheidet, ob es sich um eine Übersetzung, eine Gruppe, eine Untergruppe oder einen weiteren, verknüpften Begriff handelt. Diese Entscheidung basiert beispielsweise auf der Morphologie des Eintrags, der bei Gruppen z.B. einen gewissen Gruppen- oder Untergruppencode als Präfix enthält.

4 Entwicklung der einzelnen Module

Der Suchprozess verläuft im entwickelten System in einzelnen Etappen, in denen nacheinander zunächst die Suchanfrage auf sinntragende Elemente reduziert, dann die reduzierte Anfrage mit Hilfe des Thesaurus übersetzt und per Blind Relevance Feedback sowie vom Entry Vocabulary Modul erweitert wird.

Die eigentliche und abschließende Suche wird zum Ende des gesamten Prozess mit Hilfe der durch die einzelnen Schritte augmentierten Anfrage durchgeführt und das Ergebnis zu Evaluierungszwecken in ein TREC-übliches Evaluierungsdateiformat geschrieben. Der Cut-Off liegt hier bei 200 Dokumenten pro Suchanfrage.

4.1 Reduzierung der Suchanfrage auf sinntragende Elemente

Um aus der natürlichsprachlichen Anfrage die sinntragenden Elemente herauszufiltern und so eine sowohl im Umfang reduzierte als auch inhaltlich verdichtete Anfrage zu formulieren, wird im vorliegenden Retrievalsystem das Discriminator Modul eingesetzt.

Die Aufgabe des Discriminator Moduls ist es, Begriffe, die bereits beim Indexieren durch die Stoppwortliste ausgeschlossen wurden und Begriffe, die nicht während der Indexierung ausgeschlossen wurden, und besonders häufig im Index vorkommen, aus der Anfrage zu entfernen.

Hierbei ist das Ziel, dass das System hier nicht eine semantisch zentrale, freie Vokabel aus der Anfrage entfernt, nur weil sie nicht im kontrollierten Vokabular der Metadaten der einzelnen Dokumente vorkommt. Darum sucht das Discriminator Modul nach jedem einzelnen Term der Anfrage im Datenindex auf die freien Suchfelder *abstract* und *title*. Ergibt sich mindestens ein Treffer in einem der beiden Felder, verbleibt der Term in der weiter zu verwendenden Anfrage. Ergibt sich bei der Suche in den Daten bei einem Term eine Trefferliste mit mehr als 8000 Einträgen (Zahl auf Basis von Tests mit SWP-Evaluierungsanfragen festgestellt, Schwellenwert knapp über dem mit rund 7500 Nennungen am häufigsten verzeichneten Begriff „China“ von allen Begriffen der Evaluierungsanfragen), wird dieser Term im weiteren Suchprozess ignoriert. Diese Reduktion der Anfrage ist besonders für den Übersetzungsprozess im Translator Modul notwendig, da ohne eine solche Verkürzung oftmals versucht würde, sinnfreie Teile der Anfrage zu übersetzen und somit die Anfrage weiter um semantisch irrelevante Passagen zu ergänzen.

4.2 Übersetzung der Anfrage

In [Petras, 2005] wurden erfolgreiche Versuche der Übersetzung mit Hilfe eines mehrsprachigen Thesaurus, mit einem zusätzlichen Hinweis auf [Petras *et al.*, 2003], beschrieben. Entsprechend wurde versucht, auch in diesem Rahmen eine Übersetzung von Anfragetermen durch den Thesaurus zu realisieren. Da die Übersetzungsleistung in einem Retrievalsystem für eine inhaltlich spezifische Domäne zu großen Teilen von der thematischen Eignung des eingesetzten Wörterbuchs abhängt, war die Nutzung

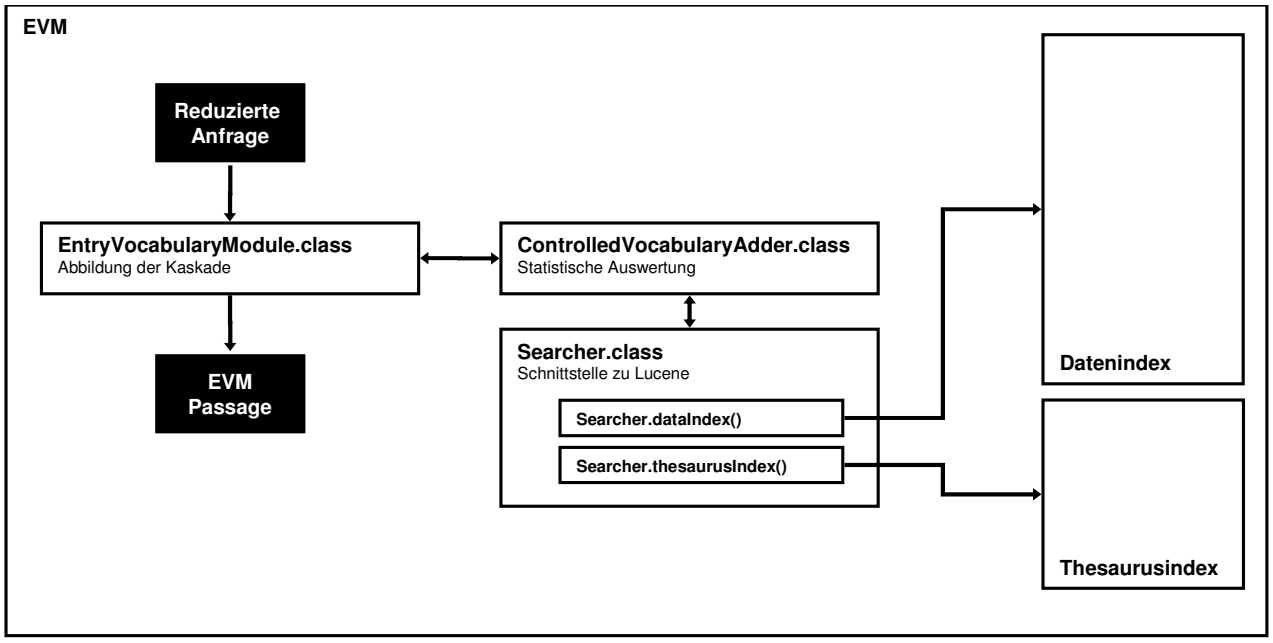


Abbildung 1: Prozessablauf des EVM

	Anfrage	Anfragefeld	Rückgabefeld	Ergebnisstring
An den Thesaurus gerichtete Anfragen:				
1.	stoppedQuery	subject	connectedterms	evmCtFromSub
2.	stoppedQuery	connectedterms	subject	evmSubFromCt
An den Datenindex gerichtete Anfragen:				
3.	stoppedQuery	title	subject	evmSubjectsFromTitleQuery
4.	stoppedQuery	abstract	subject	evmSubjectsQuery
5.	stoppedQuery	abstract	geo	evmGeoQuery
6.	evmGeoQuery	geo	subject	evmSubjectsFromGeoQuery
7.	evmSubjectsQuery	subject	classification	evmClassFromSubjectsQ.
8.	evmSubjectsQuery	subject	geo	evmGeoFromSubjectsQuery
9.	evmClassFromSubjectsQ.	classification	subject	evmSubjectsFromClassQ.

Tabelle 1: Detaillierter Verlauf der Kaskade

der Übersetzungen des vorliegenden Thesaurus naheliegend und vielversprechend.

Dem Translator Modul wird die vom Discriminator Modul reduzierte Suchanfrage übergeben. Das Übersetzungsmodul durchsuch daraufhin das Feld *subject* des Thesaurus und gibt bei einem Score von 1,0, also einer exakten Übereinstimmung, die Inhalte des Feldes *translations* zurück. Da das parallel entwickelte Modul zur Erkennung von Phrasen nicht rechtzeitig fertiggestellt werden konnte, muss sich das Translator Modul entsprechend auf termweise Übersetzungen beschränken, auch wenn bewusst ist, dass die Übersetzung von Phrasen mitunter bessere Ergebnisse erzielen kann.

Trotzdem hat eine solche Methode zur Übersetzung grundlegendes Potential: Beispielsweise lassen sich eintermige Ländernamen und Themenangaben, die zentralen Charakter für den Sinn einer Anfrage beinhalten, auf diese Art und Weise zuverlässig in Englisch, Französisch und Spanisch in der für die Datenbasis passenden Fachterminologie übersetzen.

Darüber hinaus sei darauf hingewiesen, dass der Einsatz des bereits beschriebene EVM in dem IR-System für die Datenbank des FIV einen Großteil der mehrsprachigen Retrievalleistung realisiert, da das kontrollierte Vokabular über die Dokumente aller Sprachen in einer einheitlichen Sprache verfasst ist und sich somit mit Hilfe der Deskriptoren ein grundlegendes, mehrsprachiges Retrievalverfahren realisieren lässt.

4.3 Blind Relevance Feedback

Das Blind Relevance Feedback Modul (BRF) wurde nach geringfügiger Anpassung auf den erzeugten Index aus dem System der Universität Hildesheim übernommen. Es wurde bereits in mehreren Systemen erprobt, vgl. beispielhaft [Hackl, Mandl, Womser-Hacker, 2005].

Während das im folgenden Kapitel vorgestellte EVM das lokale Feedback im Bezug auf Metadatenfelder realisieren soll, wird das BRF zusätzlich eingesetzt, um auch die Freitext-Felder *title* und *abstract* für Relevance Feedback zu nutzen. Auf diese Weise werden alle zur Verfügung stehenden Felder durch die beiden verschiedenen Varianten des Feedbacks genutzt.

Dem Blind Relevance Modul wird ebenfalls die durch den Discriminator reduzierte Anfrage übergeben. Im Rahmen der Evaluierung werden bei Einsatz des BRF pro Anfrage die 30 am besten bewerteten Dokumente untersucht und fünf Terme zur Anfragenergänzung zurückgegeben und auf die Felder *title* und *abstract* gerichtet. Die beiden Werte haben sich im Rahmen einer vorbereitenden Erprobung als vergleichsweise geeignet herausgestellt. Es

wird die Berechnungsmethode des Robertson Selection Value verwendet.

4.4 Entry Vocabulary Modul

Der Prozess dieses EVM ist in Abbildung 1 abgebildet: Zunächst wird die reduzierte Anfrage an die Klasse *EntryVocabularyModule* übergeben, dann, in den mehreren Schritten der Kaskade in der Klasse *ControlledVocabularyAdder* mit Hilfe der Klasse *Searcher* auf diverse Indexfelder angewendet und die Ergebnisse statistisch ausgewertet. Die am höchsten bewerteten Terme und Phrasen werden an die Klasse *EntryVocabularyModule* übergeben, wo alle Einzelergebnisse der Kaskadenelemente zusammengefasst und zur Ergänzung der ursprünglichen Anfrage zurückgegeben werden.

Im Falle der vorliegenden Datenbasis sind die drei für die Extraktion relevanten Felder der Datenbasis *geo*, *subject* und *classification*. Darüber hinaus werden die Felder *subject* und *connectedterms* des Thesaurus für eine Extraktion berücksichtigt. Die beiden freien Felder der Datenbasis, auf die zunächst die reduzierte Anfrage gerichtet wird sind *title* und *abstract*.

Das vorgestellte Modell wendet eine Anfragenkaskade an, die sowohl die reduzierte Anfrage, als auch aus der reduzierten Anfrage gewonnene Deskriptoren zur Gewinnung von weiteren Metadaten verwendet. Einen genauen Überblick über den Verlauf der Kaskade im evaluierten System gibt Tabelle 1. In der Tabelle werden detailliert die verwendete Anfrage, die Richtung der Anfrage auf das gegebene Feld und das ausgewertete Feld der gefundenen Dokumente sowie der zurückgegebene String aus potentiell nützlichen Deskriptoren genannt. Die Anfragenkaskade ist in der Klasse *EntryVocabularyModule* programmiert.

Im Rahmen der statistischen Auswertung wird das Suchergebnis jedes Kaskadenelements in *ControlledVocabularyAdder* untersucht. Der Umfang der Untersuchung lässt sich durch den Faktor *consideredDocs* steuern: Hier wird angegeben, wie viele der am höchsten bewerteten Dokumente in die statistische Auswertung eingehen. Über die in *consideredDocs* genannte Zahl von Dokumenten werden alle Terme und Phrasen des untersuchten Felds mit dem Score ihres Ursprungsdokuments verknüpft. Bei Mehrfachnennungen werden die Werte addiert. Über 30 Dokumente werden so z.B. im Feld *subject* bei durchschnittlich 12 Deskriptoren pro Dokument 360 Deskriptoren untersucht und ausgewertet. Abschließend werden die addierten Scores der einzelnen Deskriptoren normalisiert. Bevor die Deskriptoren zurückgegeben werden, nehmen die Parameter *cutOffScore* und *numberOfReturned* Einfluß auf die Anzahl der zurückgegebenen Dokumente. *numberOfReturned* limitiert die Anzahl der maximal zurückgegebenen Deskriptoren, falls sehr viele Deskriptoren ein höheres Ergebnis erzielt haben als in *cutOffScore* gefordert. In der Evaluierung wurden durchweg maximal sechs Terme oder Phrasen pro Kaskadenelement ergänzt.

5 Evaluierung

Im Folgenden werden die einzelnen Runs und ihre jeweiligen Unterscheidungen durch verschiedenen Parameter detailliert beschrieben. Zentrale Ansatzpunkte für die Unterscheidung der EVM-Runs sind sowohl die Wahl der Anzahl der in jedem einzelnen Kaskadenelement ausgewerteten Dokumente (30 oder 100), der auf die extrahierten Begriffe und deren Scores angewendeter Cut-Off-Score (0,3 oder 0,6) und die Anzahl der der Query hinzugefügten Terme und Phrasen (maximal sechs). Diese Parameter sind für das EVM von zentraler Bedeutung, es stellt sich die Frage, welche Einflüsse die unterschiedlichen Einstellungen auf die Retrievalleistung haben werden.

5.1 Die einzelnen Runs

Bezeichnung: SwpBase1-Nmd

- Eingesetzte Funktionen: Lucene vanilla, Stoppworte durch Indexierung
- Eingesetzte Anfragen: Eingegebene Anfrage auf die Felder *abstract* und *title*.
- Parameter des EVM: EVM nicht eingesetzt
- Globale Gewichtungen: Einfache Gewichtung der eingegebenen Anfrage

Bezeichnung: SwpBase2-Md

- Eingesetzte Funktionen: Lucene vanilla, Stoppworte durch Indexierung
- Eingesetzte Anfragen: Eingegebene Anfrage auf die Felder *abstract*, *title*, *subject* und *classification*.
- Parameter des EVM: EVM nicht eingesetzt
- Globale Gewichtungen: Einfache Gewichtung der eingegebenen Anfrage

Bezeichnung: SwpEvm1

- Eingesetzte Funktionen: Discriminator, Translator, Blind Relevance Feedback, Entry Vocabulary Modul
- Eingesetzte Anfragen: StoppedQuery (termweise angewendet auf *abstract* und *title*), RelevantTerms (termweise angewendet auf *abstract* und *title*), TranslatedQuery (termweise angewendet auf *abstract* und *title*), EvmQuery (termweise angewendet auf die Ursprungsfelder, connectedterms angewendet auf *abstract*)
- Parameter des EVM: Die maximal 30 am höchsten bewerteten Dokumente werden auf Terme und Phrasen ausgewertet. Die sechs am höchsten bewerteten Terme/Phrasen werden eingesetzt. Es gibt einen Score-Cut-Off bei 0,3 für EVM-Terme. Normalisierte Gewichtung an Termen/Phrasen entsprechend der statistischen Auswertung des EVM.
- Globale Gewichtungen: StoppedQuery: Gewichtung verzehnfacht, EvmQuery: Gewichtung einfach multipliziert mit Score-Gewicht, TranslatedQuery, RelevantTerms: Gewichtung einfach

Bezeichnung: SwpEvm2

- Eingesetzte Funktionen: Discriminator, Translator, Blind Relevance Feedback, Entry Vocabulary Modul
- Eingesetzte Anfragen: StoppedQuery (termweise angewendet auf *abstract* und *title*), RelevantTerms (termweise angewendet auf *abstract* und *title*), TranslatedQuery (termweise angewendet auf *abstract* und *title*), EvmQuery (termweise angewendet auf die Ursprungsfelder, connectedterms angewendet auf *abstract*)
- Parameter des EVM: Die maximal 100 am höchsten bewerteten Dokumente werden auf Terme und Phrasen ausgewertet. Die maximal (siehe Cut-Off)

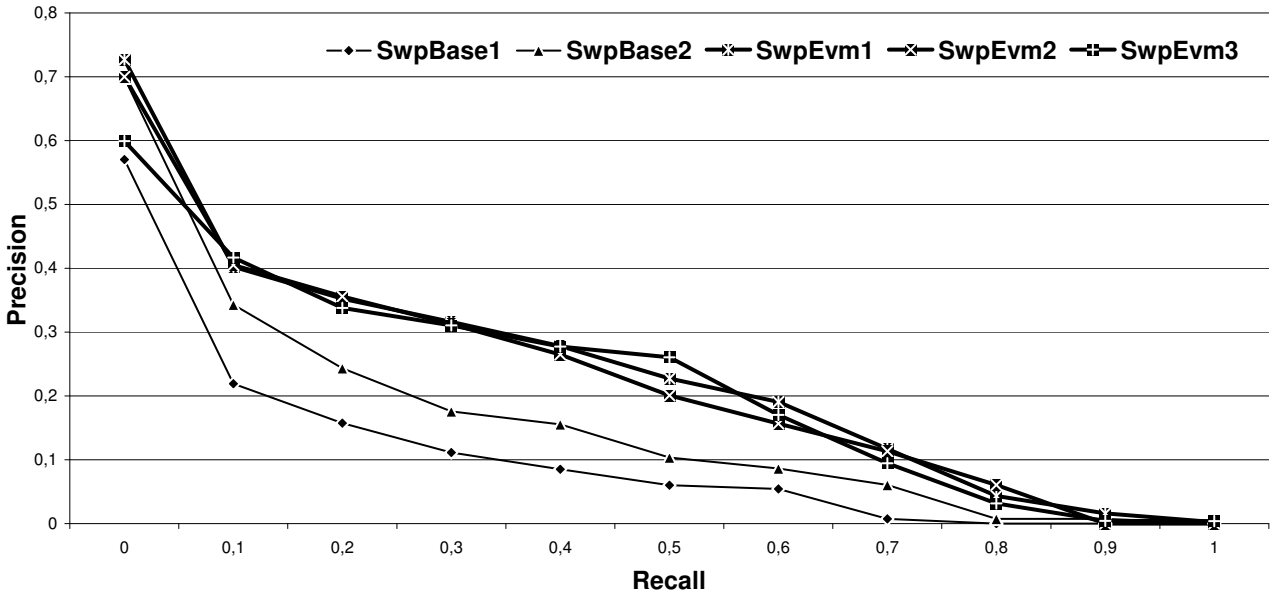


Abbildung 2: Vorläufige Ergebnisse der Evaluierung über 19 von 25 Anfragen

	BaseRun1	BaseRun2	SwpEvm1	SwpEvm2	SwpEvm3
Retrieved	3800	3800	3800	3800	3800
Relevant	1395	1395	1395	1395	1395
Rel-Ret	440	596	863	858	869
0,0	0,5705	0,6984	0,7268	0,7001	0,5992
0,1	0,2190	0,3428	0,4022	0,4049	0,4169
0,2	0,1575	0,2432	0,3525	0,3560	0,3377
0,3	0,1112	0,1759	0,3159	0,3125	0,3104
0,4	0,0851	0,1554	0,2780	0,2649	0,2775
0,5	0,0602	0,1036	0,2274	0,2008	0,2606
0,6	0,0543	0,0865	0,1907	0,1567	0,1703
0,7	0,0075	0,0604	0,1172	0,1131	0,0945
0,8	0,0000	0,0076	0,0435	0,0607	0,0315
0,9	0,0000	0,0076	0,0162	0,0000	0,0043
1,0	0,0000	0,0000	0,0022	0,0000	0,0041
Avg.Prec.	0,0865	0,1410	0,2126	0,2049	0,2070

Tabelle 2: Quantitative Ergebnisse nach Auswertung von 19 aus 25 Evaluierungsfragen

sechs am höchsten bewerteten Terme/Phrasen werden eingesetzt. Es gibt einen Score-Cut-Off bei 0,3 für EVM-Terme. Normalisierte Gewichtung an Termen/Phrasen entsprechend der statistischen Auswertung des EVM.

- Globale Gewichtungen: StoppedQuery: Gewichtung verzehnfacht, EvmQuery: Gewichtung einfach, multipliziert mit Score-Gewicht, TranslatedQuery, RelevantTerms: Gewichtung einfach

Bezeichnung: SwpEvm3

- Eingesetzte Funktionen: Discriminator, Translator, Blind Relevance Feedback, Entry Vocabulary Modul
- Eingesetzte Anfragen: StoppedQuery (termweise angewendet auf *abstract* und *title*), RelevantTerms (termweise angewendet auf *abstract* und *title*), TranslatedQuery (termweise angewendet auf *abstract* und *title*), EvmQuery (termweise angewendet auf *abstract* und *title*)
- Parameter des EVM: Die maximal 100 am höchsten bewerteten Dokumente werden auf Terme und Phrasen ausgewertet. Alle Terme/Phrasen werden eingesetzt, die über dem Score-Cut-Off von 0,6 liegen. Es wird keine spezielle Gewichtung der einzelnen Terme vorgenommen.
- Globale Gewichtungen: StoppedQuery: Gewichtung verzehnfacht, EvmQuery, TranslatedQuery, RelevantTerms: Gewichtung einfach

5.2 Der Evaluierungsprozess und erste Ergebnisse

Zur Evaluierung wurden durch die SWP 25 Evaluierungsanfragen zur Verfügung gestellt, die in ihrem Informationsgehalt, d.h. Umfang an geopolitischen und thematischen Anhaltspunkten für das IR-System, und ihrer Konkretisierung stark schwanken und somit einen hohen Anspruch an die Leistungsfähigkeit des Systems stellen. Beispielhaft können hier die folgenden zwei Fragen vorgestellt werden:

- *Frage 3: Welche Faktoren bestimmen die Beziehungen zwischen China und der EU / den einzelnen EU-Ländern?*

- *Frage 4: Welche Gefährdungen bestehen für die maritime Sicherheit in Südostasien?*

Die Evaluierung der Suchmaschine mit sämtlichen Zusatzmodulen erfolgt über einen Vergleich der Retrievalergebnisse von zwei verschiedenen BaseRuns und drei verschiedenen Evm-Runs bei denen das grundsätzliche EVM-System in diversen Parametern angepasst wurde, um den Einfluss der Parameter auf das Retrievalergebnis zu untersuchen. Zusätzlich wurde in dem Projekt das System auf den Datensatz GIRT3 evaluiert sowie auch die Leistungsunterschiede zwischen den einzelnen Modulen untersucht.

Da die Evaluierung noch andauert, kann in diesem Paper nur ein vorläufiger Überblick über das Retrievalergebnis von 19 der insgesamt 25 Evaluierungsanfragen gegen werden.

Insgesamt ist die Retrievalleistung gegenüber beiden BaseRuns deutlich angestiegen. Sowohl die Precision als auch der Recall der Evm-Runs übertrifft die Retrievalleistung, die erreicht wird, wenn die Metadaten (wie im ersten BaseRun abgebildet) nicht zum Suchvorgang hinzugezogen werden, deutlich. Im Vergleich zum zweiten BaseRun hängt die Retrievalleistung der EVM-aktivierten System allerdings deutlich von der untersuchten Query ab, trotzdem zeigt sich bei den vorliegenden Ergebnissen einen drastische Steigerung der Retrievalleistung.

6 Fazit

Da die Ergebnisse des vorgestellten IR-Systems noch nicht abschließend evaluiert und ausgewertet wurden, kann nur ein aktueller Einblick in die Entwicklungsarbeit gegeben werden.

Nach der Evaluierung von 19 aus insgesamt 25 Evaluierungsanfragen lässt sich aber vorläufig zusammenfassen, dass die EVM-basierten Runs im direkten Vergleich zu den Runs ohne entsprechende Unterstützung eine drastische Steigerung des Recalls verzeichnen können. Gleichzeitig liegt die Precision der einzelnen EVM-Runs stetig über denen der beiden BaseRuns, insgesamt lässt sich eine signifikante Steigerung der Retrievalleistung belegen.

Das dynamische EVM hat sich angesichts der ersten Evaluierungsergebnisse bewährt. In einem weiteren Schritt soll das hier vorgestellte System auch auf die GIRT Datenbasis angewandt werden, um einen besseren Vergleich mit bereits implementierten Systemen zu ermöglichen.

Literatur

[Berghaus, 2006] Benjamin Berghaus. *Mehrsprachiges Information Retrieval durch Entry Vocabulary Modul am Bespiel der Datengrundlage des Fachinformationsverbunds für Internationale Beziehungen und Länderkunde*. Magisterarbeit, Universität Hildesheim, Informationswissenschaft. 2006. erscheint.

[Buckland *et al.*, 1999] Michael Buckland, Aitao Chen, Hui-Min Chen, Youngin Kim, Byron Lam, Ray Larson, Barbara Norgard, Jacek Purat and Fredric Gey. *Mapping Entry Vocabulary to Unfamiliar Metadata*. In: Meta-Data '99 Third IEEE Meta-Data Conference, April 1999, Bethesda, USA.

[Fachinformationsverbund IBLK, 2006] World Affairs Online http://www.fiv-iblk.de *verifiziert am 30. Juli 2006*.

[Gey *et al.*, 2001] Fredric Gey, Michael Buckland, Aitao Chen and Ray Larson. *Entry vocabulary - a technology to enhance digital search*. In: Proceedings of the first international conference on Human language technology research, März 2001, San Diego, USA, S. 91-95.

[Hackl, Mandl, 2006] René Hackl, Thomas Mandl. *Bilingual Retrieval Experiments with Social Science Documents*. In: Carol Peters, Fredric Gey, Julio Gonzalo, Gareth Jones, Michael Kluck, Bernardo Magnini, Henning Müller, Maarten de Rijke. Accessing Multilingual Information Repositories: 6th Workshop of the Cross-Language Evaluation Forum, CLEF 2005, Vienna, Austria, Revised Selected Papers. Berlin et al.: Springer [Lecture Notes in Computer Science 4022]

[Hackl, Mandl, Womser-Hacker, 2005] René Hackl, Thomas Mandl, Christa Womser-Hacker. *Mono- and Cross-lingual Retrieval Experiments at the University of Hildesheim*. In: Carol Peters, Paul Clough, Julio Gonzalo, Michael Kluck, Gareth Jones, Bernard Magnini: Multilingual Information Access for Text, Speech and Images: Results of the Fifth CLEF Evaluation Campaign. Berlin et al.: Springer [Lecture Notes in Computer Science 3491] S. 165-169.

[Hellweg et al., 2001] Heiko Hellweg, Jürgen Krause, Thomas Mandl, Jutta Marx, Matthias Müller, Peter Mutschke, Robert Strötgen. *Treatment of Semantic Heterogeneity in Information Retrieval*. IZ-Arbeitsbericht Nr. 23, IZ Sozialwissenschaften, Bonn. http://www.gesis.org/Publikationen/Berichte /IZ_Arbeitsberichte/index.htm#ab23

[Kluck, 2004] Michael Kluck. *The GIRT Data in the Evaluation of CLIR Systems - from 1997 until 2003*. In: Comparative Evaluation of Multilingual Information Access Systems: 4th Workshop of the Cross-Language Evaluation Forum, CLEF 2003, Trondheim, Norway, August 21-22, 2003, Revised Selected Papers. S. 376-390

[Mandl, 2006] René Hackl, Thomas Mandl, Christa Womser-Hacker. *Neue Entwicklungen bei den Evaluierungsinitiativen im Information Retrieval*. Thomas Mandl, Christa Womser-Hacker (Hrsg.): Effektive Information Retrieval Verfahren in der Praxis: Proceedings Vierter Hildesheimer Evaluierungs- und Retrievalworkshop (HIER 2005) Hildesheim, 20.7.2005. Konstanz: Universitätsverlag [Schriften zur Informationswissenschaft 45] S. 117-128.

[Norgard, 1998] Barbara Norgard. *Entry Vocabulary Modules and Agents*. Technical Report

[Petras, 2005] Vivien Petras. *How One Word Can Make all the Difference - Using Subject Metadata for Automatic Query Expansion and Reformulation*. Working Notes for the CLEF 2005 Workshop, September 2005, Wien, Österreich.

[Stiftung Wissenschaft und Politik, 2006] Die Datenbasis: Inhalte. http://www.fiv-iblk.de/db/inhalte.htm *verifiziert am 30. Juli 2006*.

[Virtuelle Fachbibliothek Politikwissenschaften, 2006] Rechercheportal für die Politikwissenschaft. http://www.vifapol.de/ *verifiziert am 30. Juli 2006*.

[Xu, Croft, 2006] Query Expansion Using Local and Global Document Analysis. In: Proceedings of the Nineteenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 1996, S.4 - 11

[Petras, 2005] Vivien Petras. Multilingual Information Access for Text, Speech and Images. In: GIRT and the Use of Subject Metadata for Retrieval, 2005, Band 3491/2005, S.298-309

[Petras *et al.*, 2003] Vivien Petras, Natalia Perleman und Fredric Gey. Using Thesauri in Cross-Language Retrieval of German and French Indexed Collections. In: Advances in Cross-Language Information Retrieval, 2003, S.349-362

Information Retrieval is for Everybody – Beobachtungen und Thesen

Christian Wolff
Universität Regensburg
D-93040 Regensburg
christian.wolff@sprachlit.uni-regensburg.de

Abstract

In this paper, the idea of *ubiquitous information retrieval* is presented in a storytelling manner. Starting from a rough review of information retrieval system usage, some empirical hints on IR in everyday life are given. Ch. 4 explores the heterogeneity of interaction with IRS for one day in the life of a (common search engine) user. In ch. 5 summarizes these observations and suggests research approaches for modelling information retrieval as an essential component of interaction in the information society.

1 Einleitung

> Information Retrieval (IR) has become, mainly as a result of the huge impact of the World Wide Web (WWW) and CD-ROM industry, one of the most important theoretical and practical research topics in Information and Computer Science. (Dominich, 2001, p. xv)

Im Zentrum dieses Aufsatzes steht die These, dass im Kontext der Verfügbarkeit von Information Retrieval-Systemen (IRS) nicht nur im „klassischen" Bereich der Fachinformation, sondern für buchstäblich alle Bevölkerungs- und Altersschichten und Lebenslagen neue Herausforderungen auf die IR-Forschung zukommen, was sich als Forderung nach einem im Sinne der Fachinformation „deprofesssionlasisiertem *ubiquitous information retrieval"* – oder eben dem *Information Retrieval for Everybody* zuspitzen lässt.

Kap. 2 versucht einen groben Abriss der Entwicklung typischer Nutzungsformen von IRS zu geben. Kap. 2 begründet die Ausgangsthese anhand empirischer Daten und einzelner Beobachtungen. Kap. 3 stellt am Beispiel eines Tages im Leben eines *common search engine user* exemplarisch dar, wie weit Informationssysteme bereits den Alltag durchdringen und welche Probleme sich dabei ergeben. Wesentliche Beobachtungen und Thesen, die sich daraus ableiten lassen, sind in Kap. 4 zusammengefasst.

2 Entwicklungsphasen des Information Retrieval

Information Retrieval hat im Vergleich mit anderen Feldern informationswissenschaftlicher oder informatiknaher Forschung eine durchaus lange Tradition: Bibliothekswissenschaftliche Wurzeln reichen weit vor das Zeitalter der digitalen Informationstechnik zurück, frühe Visionen wie Bushs *As we may Think* (Bush, 1991) enthalten bereits wesentliche Konzepte zukünftiger IR-Systeme und die ersten IR-Systeme gehören sicher zu den frühesten Beispielen (professionell) genutzter Informationssysteme (Lilley & Trice, 1989). Auch wenn sich eine zeitliche Einteilung nicht mit letzter Trennschärfe vollziehen lässt, so kann man die (Nutzungs-)Geschichte der Information Retrieval-Systeme (IRS) grob in folgende Phasen einteilen.

2.1 IRS als Instrumente professioneller Dienstleister in vermittelter Kommunikation

In diesem Szenario (idealtypisch: bibliographische Datenbanken auf einem Host, zugänglich über formale Recherchesprache für professionelle *information workers*) existiert der Endnutzer eines Informationssystems noch nicht, Informationsbedürfnisse werden dem professionellen Rechercheur kommuniziert, der sie dann in eine Retrievalsprache umformuliert und Recherchen durchführt, Feedback des Benutzers erfolgt allenfalls bei Durchsicht der Suchergebnisse.

2.2 IRS als Werkzeug des Fachwissenschaftlers

Mit Verfügbarkeit graphisch-direktmanipulativer Benutzerschnittstellen, spätestens aber mit der Verbreitung formularbasierter Suchzugänge zu professionellen Informationssystemen im WWW, tritt der vermittelten Suche die unmittelbare des IRS durch den Endnutzer an die Seite. Auch wenn die vermittelte Recherche in forschungsnahen unternehmen, Hochschulen und Informationszentren bis heute eine wichtige Rolle spielt, dürfte doch die Mehrzahl von Recherchen mittlerweile direkt durch den primär Suchenden erfolgen. Die immer bessere Kopplung von bibliographischen Nachweissystemen mit den Volltexten digitaler Bibliotheken (insbesondere Journals und Proceedings) macht diesen unmittelbaren Zugriff besonders attraktiv.

2.3 Webbasierte Suche

Mit der Verfügbarkeit großer Informationsmengen im World Wide Web und den Suchmaschinen kommt ab Mitte der 90er Jahre des vergangenen Jahrhunderts ein weiteres Szenario hinzu: Information ganz unterschiedlicher Qualität wird ohne die Barrieren, die durch Lizenzen und eingeschränkte Verfügbarkeit gesetzt sind, unmittelbar zugänglich und deckt dabei auch Bereiche ab, die durch die traditionelle Fachinformation nicht erschlossen werden.

2.4 Post-web Suche: „Jeder sucht nach Allem“

Erst in Ansätzen erkennbar ist ein letztes Szenario, dem sich der nachfolgende Teil dieses Aufsatzes widmen wird: Die weitgehende „Digitalisierung“ (*the digitization of the world picture* (Ceruzzi, 2003, p. 346ff)) immer weiterer Bereiche der alltäglichen Informationsnutzung und die zunehmend nur noch digital vorliegende persönliche und private Information.

Deutlich wird aus den Szenarien, dass einerseits immer größere Anforderungen an den Endnutzer hinsichtlich seiner Informationskompetenz zu stellen sind (Kenntnis von Retrievalsprachen, Bewertung der Informationsqualität frei verfügbarer Information, Kenntnis geeigneter Informationssysteme (Suchmaschinen etc.)), andererseits eine klare Zielgruppe für Informationssysteme nicht mehr zu erkennen ist: Buchstäblich jeder, in jedem Alter, verwendet Informationssysteme in Beruf und Alltag.

3 Benutzer und Informationskompetenz

Einige empirische Daten und Beobachtungen mögen dies verdeutlichen: Nach der der jährlichen ARD-/ZDF-Online-Studie (hier: Ausgabe 2005) nutzen mittlerweile knapp 60% der Deutschen regelmäßig das Internet, in der Gruppe der 14-19jährigen sind es sogar 90% (van Eimeren & Frees, 2005). Lediglich die höheren Altersgruppen weisen (noch) eine deutlich niedrigere Nutzungsrate für Online-Medien auf. Eine empirische Studie zur Informationskompetenz, die der Autor 2005 an einem Oberpfälzer Gymnasium und an der Universität Regensburg bei ca. 250 Schülern und Studenten durchgeführt hat, ergab erwartungsgemäß eine fast 100%ige private Verfügbarkeit von Computer und Internet und ebenso eine durchgängige Nutzung von Suchmaschinen, insbesondere Google (Hochholzer & Wolff, 2005)[1]. Kleinere Vergleichsstudien, die im Sommersemester 2006 an Regensburger Haupt- und Realschulen im Rahmen eines Seminars zur Informationskompetenz durchgeführt wurden, zeigen ein grundsätzlich identisches Bild. Anlässlich eines Vortrags im Rahmen der Regensburger Universität für Kinder konnte der Autor im Juli 2006 bei einem Auditorium von ca. 700 Kindern im Alter zwischen sechs und zehn Jahren feststellen, dass nahezu 100% der Zuhörer nicht nur Internetnutzer sind, sondern auch Suchmaschinen wie Google benutzen (Hammwöhner & Wolff, 2006 (erscheint)). Auch wenn dies sicher keine belastbare empirische Evidenz ist, wird man doch feststellen können, dass Information Retrieval mittlerweile ein Gebiet ist, das auch junge und jüngste Zielgruppen erreicht.

Deutlich wird dabei auch, dass Suchmaschinen traditionelle Mittel der Informationserschließung wie Bibliotheken bereits verdrängen. Gleichzeitig sind Kompetenzdefizite offensichtlich, wie sie bereits frühe Benutzerstudien zu Suchmaschinen nahe legen (Wolff, 2000 m.w.N.): Kaum Nutzung von Suchoperatoren, Verwendung nur weniger Suchbegriffe, vage Vorstellung von den Möglichkeiten, gezielte Recherchestrategien aufzubauen, kein Bewusstsein der sprachlichen Problematik (Flexion, Komposita) der Suche, ein mangelhaftes Verständnis von geeigneten Kriterien, die Qualität von Information beurteilen.

4 IR im Alltag – *a Day in the Life of the Common (Search Engine) User*

Michael Lesk hat in seinem bekannten Aufsatz *The Seven Ages of Information Retrieval* die Entwicklungsgeschichte des IR mit den Prognosen von Vannevar Bush (Bush, 1991) kontrastiert und sich dabei der Shakespeareschen Metapher von den sieben Lebensaltern bedient (Lesk, 1995). Nachfolgend wird in ähnlicher Weise der hypothetische *Tages*ablauf eines Internet- bzw. WWW-Nutzers (*the common search engine user*) nachvollzogen. Als Ausgangsmaterial werden dabei einige Studien zur Evaluation von IRS herangezogen, die 2005 und 2006 im Rahmen von Information Retrieval-Projektseminaren an der Universität Regensburg durchgeführt wurden. Bewertet wurden dabei unter anderem: Bilddatenbanken, Bild-, Audio- und Videoverwaltungssysteme, Musikdownload-Plattformen, Partnerbörsen im World Wide Web, Desktopsuchmaschinen und Produktsuchmaschinen.

Bevor er (oder sie) in den Tag startet, möchte der Nutzer seinen MP3-Player mit geeigneter Musik bestücken. Dazu recherchiert er in eigenen Beständen, die er mit Hilfe online verfügbarer Datendienste wie *Freedb* (*http://freedb.org*) mit Metadaten (ID3-Tags) aufbereitet hat oder er lädt sich Musikstücke von einer Online-Plattform (z. B. *MusicLoad* (*http://www.musicload.de*), *iTunes*, (*http://www.apple.com/de/itunes/*)) herunter. Die Recherche scheitert teilweise schon an orthographischen Schwierigkeiten (*Mozart* vs. *Mozart, W.A.* vs. *Wolfgang Amadeus Mozart* …), an uneinheitlicher Verwendung der Metadaten („Interpret“ vs. „Komponist“ vs. „Teilnehmender Interpret“) oder an unterschiedlichen Genreklassifikationen (vgl. dazu die empirischen Studien (Bainbridge, Cunningham, & Downie, 2003; Cunningham, Jones, & Jones, 2004)). *Query by humming* oder *whistling* kommt morgens nicht in Frage und eine Ähnlichkeitssuche steht nicht zur Verfügung (zum Stand der Music IR vgl. die ISMIR-Proceedings[2] (*http://ismir.net*) und (Byrd & Crawford, 2002; Downie, 2004)). Neben den Problemen der Recherche treten vergleichbare Schwierigkeiten auch bei der lokalen Verwaltung einer Vielzahl von Musikstücken (als MP3 konvertierte CDs, online erworbene Lizenzen etc.) auf; zur inhaltlichen Erschließungs- und Rechercheproblematik tritt die Frage nach der rechtlichen Verfügbarkeit auf verschiedenen Rechnersystemen bzw. Medienplayern (Desktop im Büro; MP3-Player, Laptop zu Hause …).

Am Vormittag – nehmen wir an, der Benutzer sei wissenschaftlich tätig – sucht er nach Literatur, zunächst in frei verfügbaren Datenbanken wie *Google Scholar* (*http://scholar.google.de/*) oder *Citeseer* (*http://citeseer.ist.psu.edu/*). Trotz ähnlich anmutender Benutzerschnittstellen sind die Ergebnisse aber höchst unterschiedlich. Dem Benutzer ist weder klar, welches Retrievalmodell jeweils zugrunde liegt, noch wie sich die Datenbestände zusammensetzen oder ob er Zugriff auf elektronische Volltexte erwarten kann. Er beschließt daher, seine Suche mit professionellen Werkzeugen fortzusetzen. Mit Hilfe einer „Meta-Meta-Datenbank“ wie dem Datenbank-Infosystem

[1] Die Studie befindet sich derzeit noch in der Auswertungsphase; eine größere Publikation zu diesem Thema ist noch für 2006 geplant.

[2] International Symposium on Music Information Retrieval.

DBIS (*http://www.bibliothek.uni-regensburg.de/dbinfo/?bib_id=ub_r*) wählt er zunächst eine geeignete bibliographische Datenbank (z. B. SCOPUS, http://www.scopus.com) aus, recherchiert dort und greift, falls die *Elektronische Zeitschriftenbibliothek* (EZB, *http://rzblx1.uni-regensburg.de/ezeit/*) Lizenzen nachweist, über einen *link resolver* auf Volltexte elektronischer Zeitschriftenartikel zu und lädt diese auf seinen Rechner; alternativ sucht er nach Pre- oder Postprints bei *Citeseer* oder auf Publikationsservern von Hochschulen. Nicht immer ist ihm dabei klar, ob er gerade nur in Metadaten oder auch in Volltexten recherchiert; auch die Kriterien für die Verfügbarkeit von Inhalten (Bibliotheken, Verlage, Informationsdienstleister als Betreiber) sind unübersichtlich. Immerhin ist er vom ursprünglichen Informationsbedürfnis über verschiedene Systeme (DBIS, SCOPUS, EZB, Verlagswebsite) bis zu einigen Volltexten vorgedrungen.

Die Menge der für den Nutzer verfügbarer digitaler Information hat mittlerweile ein kaum zu überschauendes Maß angenommen. Um sein Wissensmanagement zu optimieren, bedient er sich daher einer Desktopsuchmaschine (z. B. *Google Desktop* (*http://desktop.google.de*) oder x-friend (*http://xfriend.de/*). Diese lässt ihn gezielt in seinen mittlerweile rund 500.000 Dateien suchen.[3] Eine ansprechende Visualisierung seiner Dateiinhalte, die Zusammenfassung zu inhaltlich verwandten Klassen oder das Aussortieren älterer Dateiversionen vermisst er allerdings. Er hofft auf eine grundlegende „Reform" des zugrunde liegenden Ordnungskonzeptes (hierarchisches Dateisystem), wird darauf aber sicher noch warten müssen (vgl. dazu das Scheitern von Microsoft, ein datenbankorientiertes Dateisystem einzuführen (Clark, 2006)). Er überlegt, ob er seine Dateien durch gezieltes manuelles Tagging vielleicht einfacher ordnen könnte (Macgregor & McCulloch, 2006), kapituliert aber vor der bereits aufgelaufenen Menge an Information. Weitergehende Lösungen, die sich der Technologien und Konzepte des *semantic web* bedienen und dedizierte Software für das *personal information management* (Bruce, 2005; Czerwinski et al., 2006) bereitstellen, befinden sich bestenfalls in einer experimentellen Phase.

Nachmittags ist etwas Zeit für einen Einkauf, er will ein elektronisches Haushaltsgerät online möglichst günstig bestellen. Die Vielfalt geeigneter Preisberatungsplattformen (z. B. *Froogle* (*http://froogle.google.de/*), *Kelkoo* (*http://www.kelkoo.de/*), *Prosuma* (*http://www.prosuma.de/*) oder *Itsbetter* (*http://www.itsbetter.de*) macht die Auswahl schwer. Auch ist ihm nicht klar, nach welchen Kriterien welche Händler auf den Plattformen vertreten sind. Neben inhaltlichen (d. h. hier: produktfunktionsbezogenen) Relevanzkriterien erschweren weitere Aspekte wie Preis, Verfügbarkeit, Lieferzeiten, Versicherungen oder Vertrauen in den Händler die Suche, die zudem medial sehr unterschiedlich unterstützt wird (je nach Plattform u. a. durch Bilder, Datenblätter, Kundenmeinungen und Erfahrungsberichte, Verkaufsrankings).

Für eine Geburtstagsfeier will er eine Glückwunschkarte mit digitalem Bildmaterial gestalten, sucht zunächst bei kommerziellen Agenturen (z. B. Corbis (*http://pro.corbis.com*), Comstock (*http://www.comstock.com/*) oder Photodisc (*http://www.photodisc.com*), stellt aber fest, dass die Bildsuche für nicht unmittelbar objektbezogene Suchkriterien keine Ergebnisse bringt („Abendstimmung"), während ihn die *folksonomies* (Marlow, Naaman, Boyd, & Davis, 2006) bei der Bildplattform Flickr (*http://www.flickr.com*))mit rund 1000 Treffern schneller zu einem passenden Ergebnis führen:

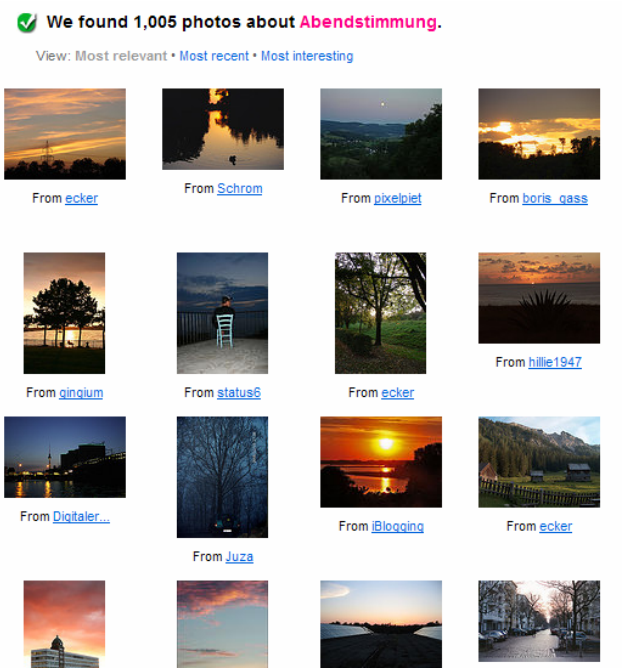


Abb. 1: Beispielrecherche „Abendstimmung" bei Flickr (http://www.flickr.com, Juli 2006)

Am Abend sieht er fern, fremdwortreiche Ausführungen eines bekannten Meteorologen veranlassen ihn, zum Laptop zu greifen und nach diesen Begriffen mit einer Suchmaschine zu recherchieren, die ihn zu entsprechenden Einträgen in der Online-Enzyklopädie Wikipedia (*http://de.wikipedia.org*) leitet. Einfacher wäre ein direkter Zugriff auf das Web über einen MHP-fähigen Digitalfernsehdecoder (*multimedia home platform*, (European Telecommunications Standards Institute, 2003)) oder die Nutzung der Zusatzmaterialen einer interaktiven Fernsehsendung, die geringe Verfügbarkeit insbesondere in Deutschland lässt dies aber als Zukunftsszenario erscheinen (Commission of the European Communities, 2006). Steigende Zahl von Sendeangeboten und ein wenigstens partieller Wechsel vom programmgetriebenen Sendemodus (*broadcasting*) zu bedürfnisgetriebenen *on demand*-Systemen lassen weitere Anwendungsgebiete für IRS (insbesondere intelligente Suchagenten) bereits erkennen.

Später trifft er bei einem Klassentreffen auf viele Bekannte und Freunde, die er zum Teil seit vielen Jahren nicht gesehen hat. Selbstverständlich werden viele Bilder gemacht, neben eigenen lädt er sich wenige Tage später auch Kopien der Bilder aller Anwesenden von einer Online-Plattform und fügt sie seiner bereits auf einige tausend Bilder angewachsenen Sammlung hinzu. Dabei stellt er fest, dass für ihn für viele Personen der Name schon wieder entfallen ist. Die digitalen Bilddaten liefern zwar präzise Informationen über Verschlusszeit, Blende und Kameramodell (und Aufnahmezeitpunkt, vgl. Abb. 2), aber

[3] Schätzung nach Angaben von *Google Desktop* für den Datenbestand des Autors; davon allein ca. 100.000 HTML-Dateien und ca. 150.000 Bilddateien.

nichts über Ort und abgebildete Person, individuelles Tagging scheidet schon mangels Wissens aus.

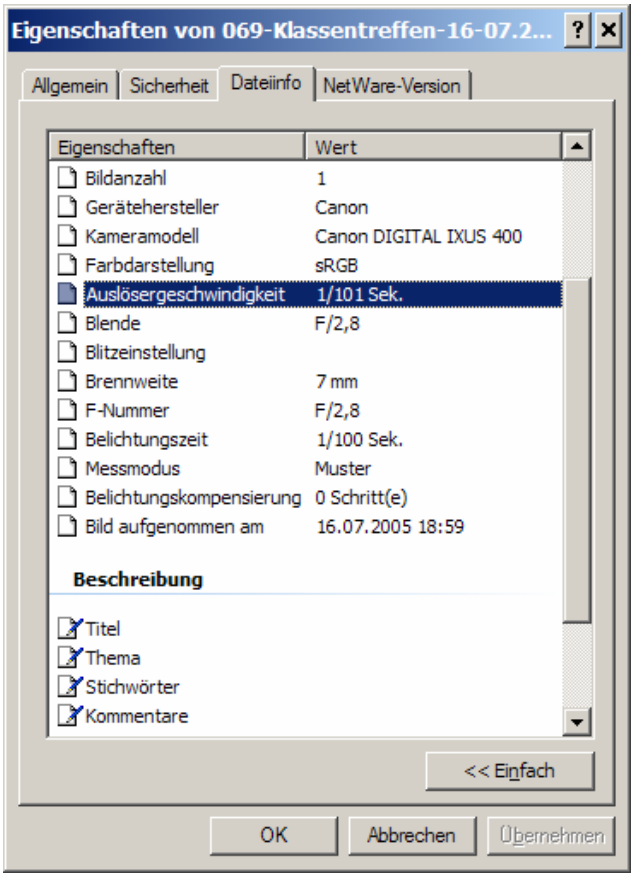


Abb. 2: Technische und (fehlende) inhaltliche Metadaten eines Digitalphotos

Nachts erneuert die Desktopsuchmaschine ihren Index, Suchagenten durchforschen das Netz nach weiteren relevanten Informationen (z. B. *alerting services* digitaler Bibliotheken, Einkaufsprofile etc.).

Die genannten Szenarien lassen sich für weitere Lebenslagen beinahe beliebig erweitern (z. B. Suche nach Erfahrungsberichten von Leidensgenossen bei Krankheit, Suche nach Kontakten und Partnern (Partnersuchbörsen, *social networking*-Plattformen wie *openBC* (*http://www.openbc.com*), sie dienten hier primär dem Zweck die Vielfältigkeit der Interaktion mit IRS zu illustrieren.

5 Beobachtungen und Thesen

Mit Blick auf die Information Retrieval Forschung ist zunächst zu fragen, ob die beobachteten Phänomene überhaupt Gegenstand der IR-Forschung sind oder sein sollen.

5.1 IR und Alltag als Gegenstand der Wissenschaft

Betrachtet man traditionelle Lehrbuchdefinitionen, so zeigt sich hier kein Widerspruch, da dort generisch alle relevanten Prozesse der Informationserschließung ohne Einschränkung auf bestimmte Informationsarten oder Medien angesprochen werden:

> Information Retrieval (IR) is concerned with the representation, storage, organization, and accessing of information items. In principle, no restriction is placed on the type of item handled in information retrieval. (Salton & McGill, 1983, p. 1)

> Information Retrieval (IR) deals with the representation, storage, organization of, and access to information items. The representation and organization of the information items should provide the user with easy access to the information in which he is interested. (Baeza-Yates & Ribeiro-Neto, 1999, p. 1)

Es erscheint legitim, die Phänomene des Information Retrieval *im Alltag* als Erweiterung der Forschungsperspektive des IR zu begreifen. Im historischen Vergleich wird man feststellen, dass die Frage, wie etwa jemand seine Bibliothek, Diasammlung oder private Korrespondenz organisiert und nutzt, von einem Nischenthema der Volkskunde, Kulturwissenschaft oder Literatursoziologie, zu einer auch technologisch interessanten Frage der Informationswissenschaft und Informatik geworden ist. Im Sinne der Generalhypothese von Ben Shneiderman („the new computing is about what users can do“, Shneiderman, 2002, p. 2) kann man hierin die Verwissenschaftlichung des Alltags erkennen – in der Informations- und Wissensgesellschaft sind IR-Systeme zum zentralen und alltäglichen Instrument geworden:

- Jeder ist Benutzer von sehr heterogenen IRS.
- Zunehmend werden *alle Medien* zum Gegenstand des IR: Nicht nur die Recherche an sich (Suchmaschinen-Paradigma), sondern auch die Aufbereitung, insbesondere die Erschließung von Information durch textuelle Metadaten wird für jedermann/-frau relevant.
- Nicht nur das WWW als globaler Datenspeicher ist dramatisch angewachsen, auch die Menge individuell verfügbarer digitaler Information wächst und über traditionelle Mediengrenzen hinweg (Text, Bild, Musik, Video). Extrapoliert man die schon heute erreichte Situation für einige Jahre oder gar Jahrzehnte, so ist leicht zuerkennen, dass neue Verfahren der Informationserschließung, -aufbereitung und -organisation und dringend erforderlich sind..
- Zeitliche, ökonomische, organisatorische Kriterien gehen in die Effektivitätsbewertung mit ein, in vielen Fällen wird man bei der Systembewertung kaum abstrahieren können („lieber ein kostenloses Bild bei Flickr als ein besseres, aber teures bei Corbis“).
- Phänomene der Medienkonvergenz in den digitalen Medien gewinnen an Bedeutung; die klassischen Rundfunkmedien werden in Kürze über die „traditionellen“ digitalen Datennetze (Internet, WWW) verfügbar sein, umgekehrt zeichnet sich (siehe oben) auch die Kopplung von WWW und digitalem Fernsehen ab (Theunert, 2002).[4]
- Die Potentiale der Social Software (Bächle, 2006; Möller, 2005; Sixtus, 2005) für die Informationserschließung sind zunächst nur phänomenologisch erkannt worden: Eine Reihe von Portalen erleichtern die gemeinsame Erschließung und Distribution unterschiedlicher Medien für sehr heterogene Nutzerkreise

[4] Medien*konvergenz* der *Inhalte* könnte dabei mit Medien*divergenz* der Nutzungs*geräte* und –*szenarien* einhergehen, wenn im Sinne des *ubiquitous computing* ganz unterschiedliche Endgeräte verfügbar werden.

(z. B. CiteULike (*http://www.citeulike.org*) im Bereich wissenschaftlicher Literatur, Flickr im Bereich Bildmedien).

- Als Rechercheparadigma ist dabei die begriffsbasierte, textuelle Suche immer noch das primäre Mittel der Anfrageformulierung, da z. B. automatische Bilderkennung (Santini, 2001) oder Videoanalyse (Feng, Siu, & Zhang, 2003) noch nicht flächendeckend verfügbar sind und innovative Visualisierungsformen als Interaktions- und Präsentationsparadigmen sich bisher nicht durchsetzen konnten (Arnold & Wolff, 2005).

Vor dem Hintergrund der Heterogenität von Nutzern und Systemen dürften sich die bekannten Probleme der IR-Forschung verschärfen, insbesondere im Bereich der Informationskompetenz – der viel diskutierte *digital divide* (Kizza, 2003; Shneiderman, 2002) bezieht sich mittlerweile (in den Industrienationen) weniger auf verfügbare *Hardware* oder den *Zugang* zu den Datennetzen als auf die Kompetenz im Umgang mit Informationssystemen (für 1998/99 noch anders: Wagner, Pischner, & Haisken-DeNew, 2002). Dazu gehören:

- Aspekte der sprachlichen Aufbereitung von Suchbegriffen (Vollformen, Eigennamen, Komposita).
- Kenntnisse von den Möglichkeiten der Anfrageformulierung (fehlendes oder falsches mentales Modell von der Retrievalfunktion).
- Die besonderen Probleme nicht-textueller Medien (Musik, Bild, Film etc.), insbesondere die Beschreibung durch textuelle Metadaten.

Über den „common user" ist generell noch recht wenig bekannt, für die Frühphase der Suchmaschinennutzung ab 1995 konstatiert (Ceruzzi, 2003, p. 329) zwar: "Computer-savvy Internet users did not need a portal. They preferred brute-force search engines and where not afraid to construct complex searches using Boolean algebra to find what they wanted." Diese Beobachtung steht in deutlichem Gegensatz zu empirischen Studien zum Rechercheverhalten und der Informationskompetenz von WWW-Nutzern (sieh oben Kap. 3).

Der alltäglichen und nicht berufsbedingten Nutzung des Internet und seiner Informationsdienste sind mittlerweile einige Studien gewidmet worden (Bakardjieva, 2005; Silverstone, 2005; Wellman & Haythornthwaite, 2002), der besonderen Problematik der Interaktion mit IRS schenken diese aber kaum Beachtung. Die stärkere Nutzerorientierung hat im IR-Bereich zwar schon seit längerem im *cognitive viewpoint* des IR ihren Niederschlag gefunden (Ellis, 1998; Ford, 2004; Ingwersen, 1999), bezieht sich aber ebenfalls auf Fachinformation und den akademischen Nutzer, also auf wissenschaftsnahe Kontexte. Auch ungeachtet dieser theoretischen und methodischen Perspektive liegt der Schwerpunkt der publizierten IR-Forschung sicher eher im Bereich der Weiterentwicklung von IR-Algorithmen bzw. der Technologieentwicklung.[5]

[5] Als Indiz kann man hier die thematischen und methodischen Schwerpunkte der *Annual ACM Conference on Research and Development in Information Retrieval* heranziehen, die als internationale IR-Leittagung gelten kann, vgl. http://portal.acm.org/browse_dl.cfm?linked=1&part=series&idx=SERIES278&coll=ACM&dl=ACM&CFID=626045&CFTOKEN=83337271.

5.2 Fazit: Konsequenzen für die Forschung

Viele der in Kap. 4 angedeuteten Probleme der Informationssuche im Alltag lassen sich durch *technologische Weiterentwicklungen* abmildern und beheben. Dies kann man wiederum anhand eines praktischen Beispiels – hier: Die Metadatenerfassung in der Digitalphotographie leicht illustrieren:

- Die Kopplung von Photographie mit Spracherkennung, könnte die Erfassung von Metadaten (*tagging*) an der Quelle, d. h. in unmittelbarem Zusammenhang mit der Bildentstehung erlauben – eine Kamera oder ein Kamerahandy hat eher ein Mikrophon als eine Tastatur).
- Maschinelles Lernen und Mustererkennung können helfen, bekannte Personen auf Bildern – im Sinne von Shneidermans Schichtenmodell des persönlichen Umfelds sicher eine überschaubare Zahl (Shneiderman, 2002, p. 80ff) – automatisch zu erschließen.
- Mit Hilfe von RFID-Tags[6] und einer Modellierung der Informationsfreigabe nach Bekanntheitsgrad ließen sich Metadaten von einer photographierten Person leicht auf eine Kamera übertragen.
- Die Integration eines GPS-Empfängers[7] in die Kamera, gekoppelt mit einem geographischen Informationssystem, würde die automatische Übernahme von Ortsinformation ermöglichen.

Weniger offensichtlich sind technologische Lösungen für das komplexere Problem der persönlichen Wissensorganisation (*personal information management*). Voraussetzung dafür wäre – und das ergibt eine zusätzliche Forschungsperspektive – ein geeignetes Modell unseres informationellen Alltags und unserer informationellen Umwelt. Ansatzpunkte kann man hierfür in der Informationskompetenz-Forschung finden, allerdings ist diese stark bibliotheks- und damit auch wissenschaftsorientiert (Eisenberg, Lowe, & Spitzer, 2004; Neely, 2002; Pickering Thomas, 1999). Neben der notwendigen theoretischen Modellbildung für das *ubiquitous information retrieval* im Alltag sind vor allem alters- und gruppenspezifische empirische Studien zur Informationskompetenz, typischen „alltäglichen" Informationsbedürfnissen und zur Praxis des Umgangs mit IRS ein dringliches Desiderat. Das etwas plakative Fazit kann daher lauten: *Information Retrieval is for everybody, but we don't know much about anybody.*

Literatur

Arnold, C., & Wolff, C. (2005). Evaluierung von Visualisierungsformaten bei der webbasierten Suche. In Forschungszentrum Jülich (Ed.), *Knowledge eXtended. Die Zusammenarbeit von Wissenschaftlern, Bibliothekaren und IT-Spezialisten* (pp. 275-286). Jülich: Forschungszentrum Jülich GmbH.

Bächle, M. (2006). Social software. *Informatik-Spektrum, 29*(2), 121-124.

Baeza-Yates, R., & Ribeiro-Neto, B. (1999). *Modern Information Retrieval.* Harlow et al. / New York: Addison-Wesley / ACM Press.

Bainbridge, D., Cunningham, S. J., & Downie, J. S. (2003). *Analysis of queries to a Wizard-of-Oz MIR system: Challenging assump-*

[6] Radio Frequency Identification.
[7] Global Positioning System.

tions about what people really want. Paper presented at the Proceedings of the 4th International Conference on Music Information Retrieval, Baltimore, MD.

Bakardjieva, M. (2005). *Internet Society. The Internet in Everyday Life*. London / Thousand Oaks, CA / New Dehli: Sage Publications.

Bruce, H. (2005). Personal, anticipated information need. *Information Research, 10*(5).

Bush, V. (1991). As We May Think (1945) In J. M. Nyce & P. Kahn (Eds.), *From Memex to Hypertext: Vannevar Bush and the Mind's Machine*. Boston: Academic Press.

Byrd, D., & Crawford, T. (2002). Problems of Music Information Retrieval in the Real World. *Information Processing & Management, 38.*(2), 249-272.

Ceruzzi, P. E. (2003). *A History of Modern Computing* (2nd ed.). Cambridge, MA / London: The MIT Press.

Clark, Q. (2006). What's in Store. Update to the Update [Electronic Version]. *Microsoft Developer Network (MSDN). WinFS Team Blog*, 2006. Retrieved July 31, 2006 from http://blogs.msdn.com/winfs/archive/2006/06/26/648075.aspx.

Commission of the European Communities. (2006). *Communication From The Commission to the Council, the European Parliament, the European Economic and Social Committee and the Committee of the Regions on Reviewing the Interoperability of Digital Interactive Television Services Pursuant to Communication Com(2004) 541 Of 30 July 2004*. Brüssel: Commission of The European Communities.

Cunningham, S. J., Jones, M., & Jones, S. (2004). *Organizing Digital Music for Use: An Examination of Personal Music Collections.* Paper presented at the Proceedings of the 5th International Conference on Music Information Retrieval, Barcelona.

Czerwinski, M., Gage, D., W., Gemmell, J., Marshall, C., C. , Pérez-Quiñonesis, M. A., Skeels, M., M. , et al. (2006). Digital memories in an era of ubiquitous computing and abundant storage. *Communincations of the ACM, 49*(1), 44-50.

Dominich, S. (2001). *Mathematical Foundations of Information Retrieval*. Dordrecht et al.: Kluwer Academic Publishing.

Downie, J. S. (2004). A Sample of Music Information Retrieval Approaches. *Journal of the American Society for Information Science and Technology 55*(12), 1033–1036. .

Eisenberg, M. B., Lowe, C. A., & Spitzer, K. L. (2004). *Information Literacy. Essential Skills for the Information Age*. New York et al. : Macmillan Publishers.

Ellis, D. (1998). Paradigms and research traditions in information retrieval research. *Information Services and Use, 18*(4), 225-241.

European Telecommunications Standards Institute. (2003). *Digital Video Broadcasting (DVB). Globally Executable MHP (GEM) Specification 1.0.0.* (No. Dokument ETSI TS 102 819).

Feng, D. D., Siu, W.-C., & Zhang, H.-J. (Eds.). (2003). *Multimedia Information Retrieval and Management - Technological Fundamentals and Applications*: Berlin: Springer Verlag.

Ford, N. (2004). Modeling cognitive processes in information seeking: From popper to pask. *Journal of the American Society for Information Science and Technology, 55*(9), 769-782.

Hammwöhner, R., & Wolff, C. (2006 (erscheint)). Wie funktioniert das Internet? In M. Fölling-Albers (Ed.), *Regensburger Universität für Kinder 2006*. Regensburg: pro Regensburg e.V.

Hochholzer, R., & Wolff, C. (2005). *Informationskompetenz – status quo und Desiderate für die Forschung*. Regensburg: Universität Regensburg, Institut für Germanistik und Institut für Medien-, Informations- und Kulturwissenschaft.

Ingwersen, P. (1999). Cognitive information retrieval. *Annual Review of Information Science and Technology, 34*, 3-52.

Kizza, J. M. (2003). *Ethical and Social Issues in the Information Age* (2nd ed.). New York et al.: Springer.

Lesk, M. (1995). *The Seven Ages of Information Retrieval*: International Federation of Library Associations and Institutions, Universal Dataflow and Telecommunications Core Activity (UDT).

Lilley, D. B., & Trice, R. W. (1989). *A History of Information Science. 1945 - 1985*. San Diego, CA: Academic Press.

Macgregor, G., & McCulloch, E. (2006). Collaborative Tagging as a Knowledge Organisation and Resource Discovery Tool. *Library Review, 55*(5), 291-300.

Marlow, C., Naaman, M., Boyd, D., & Davis, M. (2006). *Position Paper, Tagging, Taxonomy, Flickr, Article, ToRead.* Paper presented at the WWW2006 Tagging Workshop Edinburgh.

Möller, E. (2005). *Die heimliche Medienrevolution. Wie Weblogs, Wikis und freie Software die Welt verändern*. Hannover: Heise.

Neely, T. Y. (2002). *Sociological and Psychological Aspects of Information Literacy in Higher Education*. Lanham/MD: Scarecrow Press.

Pickering Thomas, N. (1999). *Information Literacy and Information Skills Instruction, Applying Research to Practice in the School Library Media Center*. Westport/CT: Libraries Unlimited.

Salton, G., & McGill, M. J. (1983). *Introduction to Modern Information Retrieval*. New York et al.: McGraw-Hill.

Santini, S. (2001). *Exploratory Image Databases. Content-based Retrieval.* . San Diego, CA: Academic Press.

Shneiderman, B. (2002). *Leonardo's Laptop: Human Needs and the New Computing Technologies*. Cambridge, MA / London: The MIT Press.

Silverstone, R. (Ed.). (2005). *Media, Technology and Everyday Life in Europe. From Information to Communication*. Aldershot / Burlington, VT: Ashgate.

Sixtus, M. (2005). Das Web sind wir. *Technology Review*(Juli 2005), 44-52.

Theunert, H. (Ed.). (2002). *Medienkonvergenz: Angebot und Nutzung. Eine Fachdiskussion veranstaltet von BLM und ZDF* (Vol. 70). München: Fischer.

van Eimeren, B., & Frees, B. (2005). ARD/ZDF-Online-Studie 2005. Nach dem Boom: Größter Zuwachs in internetfernen Gruppen. *Media Perspektiven*(8/2005), 362-379.

Wagner, G. G., Pischner, R., & Haisken-DeNew, J. P. (2002). The Changing Digital Divide in Germany. In B. Wellman & C. Haythornthwaite (Eds.), *The Internet in Everyday Life.* (pp. 164-185). Malden, MA / Oxford / Carlton: Blackwell Publishin.

Wellman, B., & Haythornthwaite, C. (Eds.). (2002). *The Internet in Everyday Life*. Malden, MA / Oxford / Carlton: Blackwell Publishing.

Wolff, C. (2000). Vergleichende Evaluierung von Such- und Metasuchmaschinen im World Wide Web. In G. Knorz & R. Kuhlen (Eds.), *Informationskompetenz – Basiskompetenz in der Informationsgesellschaft. Proc. 7. Intern. Symposium f. Informationswissenschaft, ISI 2000* (pp. 31-48). Konstanz: UVK.

Anmerkung

Alle WWW-Ressourcen, die in diesem Artikel erwähnt sind, wurden im Juli 2006 verifiziert.

Inferring the user interests using the search history

Lynda Tamine, Mohand Boughanem, Nesrine Zemirli
IRIT-SIG
118 Route de Narbonne
31062 Toulouse CEDEX 06 France
tamine@irit.fr, bougha@irit.fr, nzemirli@irit.fr

Abstract

Personalization involves the process of gathering user-specific information during interaction with the user, which is then used to deliver appropriate results to the user's needs. This paper presents a statistical method that learns the user interests by collecting evidence from his search history. The method focuses on the use of both user relevance point of view on familiar words in order to infer and express his interests and the use of a correlation metric measure in order to update them.

1 Introduction

It is widely assumed nowadays that because of the explosive growth of web documents, keyword based search technologies are not effective, in the sense that they are not able to deliver appropriate results in response to specific user's information needs. The major reason is that they don't take into account the user profile in the retrieval process [Numberg, 2003; Budzik and Hammond, 1985].

Although, relevance feedback techniques [Rocchio, 1971] improve the retrieval accuracy by considering the user's preferences, they are not effective in real world applications [Kelly and Teevan, 2003]. [Budzik and Hammond, 1985] In order to tackle this problem, contextual information retrieval emerged recently as an active area. It explores the impact of context, viewed as a set of social, cultural and task features, on human information behaviour. Our interest in context is namely in defining user's profiles in order to constraint the semantic space of information determining the topical relevance. In this sense, several approaches explored techniques for building user's profile using implicit feedback [Pazzani and Billsus, 1997; Mc Gowan, 2003; Lieberman, 1995; Pretshner and Gauch, 1999; Liu and Yu, 2004; Budzik and Hammond, 1985]. Most of them model the user long-term interests as retrieval contexts represented by word vectors [Pazzani and Billsus, 1997; Lieberman, 1995; Budzik and Hammond, 1985],class of concepts [Mc Gowan, 2003] or a hierarchy of concepts [Liu and Yu, 2004; Pretshner and Gauch, 1999].

This paper presents a new technique for building and learning the user interests accross past search sessions. Comparatived to previous work, our approach has the following new features:

- related and unrelated user's interests are dynamically inferred from the search history using a statistical rank-order correlation operator,
- rather than using a basic Tf-Idf word weighting scheme in the user profile representation, we propose a new measure to estimate the relevance of the words according to the user interests.

In section 2, we present the strategy of collecting and modeling the user's search history . In section 3, we explain how they are used to learn the user's interests.

2 Building the user profile using search history

In our point of view, a user profile expresses the user long-term interests. It contains two related components: an aggregative representation of the user search history and a library of user contexts reflecting his interests when seeking information. More precisely, our approach uses the evidence collected across successive search sessions in order to track potential changes in the user's interests. At time s, the user profile is represented as $U = (H^s, I^s)$ where H^s and I^s represent respectively the search history and a set of interests of the user U at time s. Our method runs in two main steps. The first one consists of representing the user search history by collecting information from user feedback at each retrieval session, and then gathering this information in order to infer the user contexts expressed using a set of weighted dominant keywords. The second step consists of learning the user interests by using the contexts discovered during the previous step. The learning algorithm is based on a correlation measure used to estimate the level of changes in the user interests structure during a period of time.

2.1 Representation of the user search history

Let q^s be the query submitted by a specific user U at the retrieval session performed at time s. We assume that a document retrieved by the search engine with respect to q^s is relevant if it generates some observable user behaviours (reading during a reasonable duration, saving, printing etc) or it is explicitly judged as relevant by the user. Let D^s be the related set of assumed relevant documents during session S^s, $R_u^s = \cup_{s_0..s} D^s$ represents the potential space search of the user across the past search sessions. We use matrices to represent both user search session and search history. The construction of this matrix, described below, is based on the user's search record and some features inferred on the user relevancy point of view. The user search session is represented by a Document-Term matrix S^s $D^s * T^s$ where T^s is the set of terms indexing D^s (T^s is a part of all the representative terms of the previous relevant documents, denoted $T(R_u^s)$. Each row in the matrix S^s represents a document $d \in D^s$, each column represents a term $t \in T^s$. In order to improve the accuracy of document-term representation, we aim at introducing in the weighting scheme a

factor that reflects the user's interest for specific terms. For this purpose, we use term dependencies as association rules checked among T^s [Lin *et al.*, 1998] in order to compute the user term relevance value of term t in document d at time s denoted $RTV^s(t,d)$:

$$RTV^s(t,d) = \frac{w_{td}}{dl} * \sum_{t' \neq t, t' \in D^s} cooc(t,t') \quad (1)$$

w_{td} is the common Tf-Idf weight of the term t in the document d, dl is the length of the document d, $cooc(t,t')$ is the confidence value of the rule $(t \rightarrow t')$, $ccoc(t,t') = \frac{n_{tt'}}{n_t}$, $n_{tt'}$ is the number of documents among D^s containing t and t', n_t is the number of documents among D^s containing t. $S^s(d,t)$ is then determined as:

$$S^s = (RTV^s)^t \quad (2)$$

The user search history is a $R_u^s * T(R_u^s)$ matrix, denoted H^s, build dynamically by reporting document information from the matrix S^s and using an aggregative operator combining for each term its basic term weight and relevance term value computed across the past search sessions as described above. More precisely, the matrix H^s is built as follows:

$$H^0(d,t) = S^0(d,t)$$

$$H^{s+1}(d,t) = \begin{cases} \alpha * w_{t,d} + \beta * S^{s+1}(d,t)\ if \\ t_j \notin T(R_u^{(s-1)}) \\ \alpha * H^s(d,t) + \beta * S^{s+1}(d,t) if \\ t_j \in T(R_u^{(s-1)}) and\ d \in R_u^{(s-1)} \\ H^s(d,t)\ otherwise \end{cases} \quad (3)$$

$(\alpha + \beta = 1), s \succ s_0$

2.2 Learning the user's interests

The goal of this step is to extract the user contexts from his search history in order to learn his long-term interests. For this purpose, we propose a statistical method that constructs and updates a set of user's interests I^s. This method induces at each learning period, a set of beliefs on the user contexts represented each one as a set of weighted key words. At learning time s, an ordered vector denoted c^s reflecting a query context, is built using the formula:

$$c^s(t) = \sum_{d \in R_u^s} H^s(d,t) \quad (4)$$

$c^s(t)$ is normalised as follows: $c^s(t) = \frac{c^s(t)}{\sum_{t \in T^s} c^s(t)}$. In order to track the changes in the user's interests, we compare the current context cc^s and the previous one pc^s using Kendall rank-order correlation operator $\circ$:

$$\Delta I = (cc^s \circ pc^s) = \sum_{t \in T(R_u^s)} (cc^s(t) - pc^s(t)) \quad (5)$$

The coefficient value ΔI is in the range [-1 1], where a value closer to -1 means the query contexts are not similar and a value closer to 1 means that the query contexts are very related each other. Based on this coefficient value, we apply the following strategy in order to learn the user's interests and so update the set of user interests I^s :

1. $\Delta I > \sigma$ (σ represents a threshold correlation value). No potential changes in the query contexts, no information available to update I^s;
2. $\Delta I < \sigma$. There is a change in the query contexts. In this case we gauge the level of change: is this reflects a refinement of a prior detected user interest or the occurrence of a novel one? In order to answer this question we do as follows:
   - select $c^* = argmax_{c \in I^s}(c \circ cc^s)$,
   - if $cc^s \circ c^* > \sigma$ then refine the user interest c^*, update the matrix H^s by dropping the rows representing the least recently documents updated, update consequently R_u^s,
   - if $cc^s \circ c^* < \sigma$ then add the new tracked interest in the library I^s, try to learn a period of time c^*: set $H^{s+1} = S^s$, $s_0 = s$

3 Conclusion and future work

In this paper, we described a new approach for user profiling using statistical methods to gather the search history and track changes in user's interests. Unlike most previous related work, we focus on the updating of the search history representation using user relevance point of view on familiar words, in order to build and learn different user's interests. The design of an experimental evaluation of our approach requires a large scale of quantitative data on user search sessions and accurate contexts provided by the related queries during a reasonable period of testing a particular search engine. We currently develop an evaluation methodology which includes the construction of such collections test and the definition of accurate performance measures.

4 Ackowledgments

This research was partially supported by the French Ministry of Research and New Technolologies under the ACI program devoted to Data Masses (ACI-MD), project MD-33.

References

[Budzik and Hammond, 1985] J. Budzik, K.J Hammond. Users interactions with everyday applications as context for just-in-time information access. In *Proceedings of the 5th international conference on intelligent user interfaces*, pp 44-51, 2000.

[Mc Gowan, 2003] J.P Mc Gowan. A multiple model approach to personalised information access. Master Thesis in computer science, Faculty of science, University College Dublin, February 2003.

[Kelly and Teevan, 2003] D. Kelly, J. Teevan. Implicit feedback for inferring user preference: A bibliography. SIGIR Forum, 2003

[Lieberman, 1995] H. Lieberman. Letizia, *"An agent that assists web browsing"*. In *Proceedings of the International Joint Conference on Artificial Intelligence* (IJCAI'95), pp 924:929, Montreal, August 1995

[Lin *et al.*, 1998] S.H. Lin, C.S. Shih, M.C. Chen, J. Ho, M. Ko and Y. M. Huang. Extracting classification knowledge of Internet documents with mining term-associations: A semantic approach. In *the 21th International SIGIR Conference on Research end Development in Information Retrieval*, 1998

[Liu and Yu, 2004] F. Liu, C. Yu. Personalized Web search for improving retrieval effectiveness. IEEE Transactions on knowledge and data engineering, 16(1), pp 28-40, 2004

[Numberg, 2003] G. Nunberg As Google goes, so goes the nation, New York times, May 2003

[Pazzani and Billsus, 1997] M. Pazzani, D. Billsus. Learning and revising user profiles : The identification of interesting Web sites, Machine learning, Vol 27, pp 313-331, 1997

[Pretshner and Gauch, 1999] A. Pretshner, S. Gauch. Ontology based personalised search, In *Proceedings of the 8th IEEE International Conference, Tools with Artificial Intelligence* (ICTAI), pp 391-198, 1999

[Rocchio, 1971] J. Rocchio. Relevance feedback in information retrieval, In G. Salton editor, The SMART retrieval system - experiments in automated document processing. Prentice-Hall, Englewood Cliffs, NJ, 1971

FolkRank: A Ranking Algorithm for Folksonomies

Andreas Hotho, Robert Jäschke, Christoph Schmitz, Gerd Stumme
Knowledge & Data Engineering Group, Department of Mathematics and Computer Science
University of Kassel, Wilhelmshöher Allee 73, D–34121 Kassel, Germany
http://www.kde.cs.uni-kassel.de

Research Center L3S, Expo Plaza 1, D–30539 Hannover, Germany
http://www.l3s.de

Abstract

In social bookmark tools users are setting up lightweight conceptual structures called folksonomies. Currently, the information retrieval support is limited. We present a formal model and a new search algorithm for folksonomies, called *FolkRank*, that exploits the structure of the folksonomy. The proposed algorithm is also applied to find communities within the folksonomy and is used to structure search results. All findings are demonstrated on a large scale dataset. A long version of this paper has been published at the European Semantic Web Conference 2006 [3].

1 Introduction

Social resource sharing tools, such as Flickr,[1] del.icio.us,[2], or our own system *BibSonomy*[3] (see Fig. 1) have acquired large numbers of users within less than two years. The reason for their immediate success is the fact that no specific skills are needed for participating, and that these tools yield immediate benefit for each individual user (e.g. organizing ones bookmarks in a browser-independent, persistent fashion) without too much overhead. Large numbers of users have created huge amounts of information within a very short period of time. The frequent use of these systems shows clearly that web- and folksonomy-based approaches are able to overcome the knowledge acquisition bottleneck, which was a serious handicap for many knowledge-based systems in the past.

Social resource sharing systems are web-based systems that allow users to upload their resources (e. g., bookmarks, publications, photos; depending on the system), and to label them with arbitrary words, so-called *tags*. For an overview over the state of the art of folksonomy research, we refer to [3].

In their core, these systems are all very similar. Once a user is logged in, he can add a resource to the system, and assign arbitrary tags to it. The collection of all his assignments is his *personomy*, the collection of all personomies constitutes the *folksonomy*. The user can explore his personomy, as well as the personomies of the other users, in all dimensions: for a given user one can see all resources he had uploaded, together with the tags he had assigned to them (see Fig. 1); when clicking on a resource one sees which other users have uploaded this resource and how they tagged it; and when clicking on a tag one sees who assigned it to which resources.

[1] http://www.flickr.com/

[2] http://del.icio.us

[3] http://www.bibsonomy.org

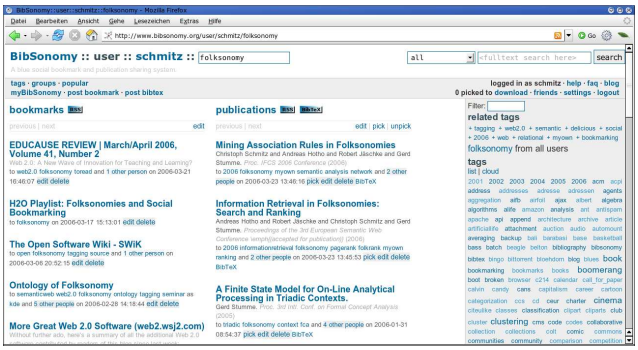


Figure 1: Bibsonomy displays bookmarks and BibTeX based bibliographic references simultaneously.

The systems allow for additional functionality. For instance, one can copy a resource from another user, and label it with one's own tags. Overall, these systems provide a very intuitive navigation through the data. However, the resources that are displayed are usually ordered by date, i. e., the resources entered last show up at the top. A more sophisticated notion of 'relevance' – which could be used for ranking – is still missing.

To this end, we propose a formal model for folksonomies, and present a new algorithm, called *FolkRank*, that takes into account the folksonomy structure for ranking search requests in folksonomy based systems. The algorithm will be used for two purposes: determining an overall ranking, and specific topic-related rankings.

2 Folksonomies

A folksonomy describes the users, resources, and tags, and the user-based assignment of tags to resources. The following definition of folksonomies is also underlying our BibSonomy system.

A *folksonomy* is a tuple $\mathbb{F} := (U, T, R, Y)$ where U, T, and R are finite sets, whose elements are called *users*, *tags* and *resources*, resp., and Y is a ternary relation between them, i. e., $Y \subseteq U \times T \times R$, whose elements are called tag assignments (TAS for short). [4]

Users are typically described by their user ID, and tags may be arbitrary strings. What is considered as a resource depends on the type of system. For instance, in del.icio.us, the resources are URLs, and in flickr, the resources are pictures. From an implementation point of view, resources are internally represented by some ID.

This structure is known in Formal Concept Analysis [2] as a *triadic context* [5; 8]. An equivalent view on folksonomy data is that of a tripartite (undirected) hypergraph

[4] In the long version of this paper, we introduce additionally a user-specific tag hierarchy $\prec$.

$G = (V, E)$, where $V = U \dot\cup T \dot\cup R$ is the set of nodes, and $E = \{\{u, t, r\} \mid (u, t, r) \in Y\}$ is the set of hyperedges.

In order to evaluate our retrieval technique detailed in the next section, we have analyzed the popular social bookmarking sytem del.icio.us, which is a server-based system with a simple-to-use interface that allows users to organize and share bookmarks on the internet. It is able to store in addition to the URL a description, an extended description, and tags (i. e., arbitrary labels). We chose del.icio.us rather than our own system, BibSonomy, as the latter went online only after the time of writing of this article. For our experiments, we collected, from July 27 to 30, 2005, data from the del.ico.us system, and obtained a core folksonomy with $|U| = 75,242$ users, $|T| = 533,191$ tags and $|R| = 3,158,297$ resources, related by in total $|Y| = 17,362,212$ tag assignments.

3 Ranking in Folksonomies using Adapted PageRank

Current folksonomy tools such as del.icio.us provide only very limited search support in addition to their browsing interface. Searching can be performed over the text of tags and resource descriptions, but no ranking is done apart from ordering the hits in reverse chronological order. Using traditional information retrieval, folksonomy contents can be searched textually. However, as the documents consist of short text snippets only (usually a description, e. g. the web page title, and the tags themselves), ordinary ranking schemes such as TF/IDF are not feasible.

As discussed above, a folksonomy induces a graph structure which we will exploit for ranking in this section. Our *FolkRank* algorithm is inspired by the seminal PageRank algorithm [1]. The PageRank weight-spreading approach cannot be applied directly on folksonomies because of the different nature of folksonomies compared to the web graph (undirected triadic hyperedges instead of directed binary edges). In the following we discuss how to overcome this problem.

3.1 Adaptation of PageRank

We implement the weight-spreading ranking scheme on folksonomies in two steps. First, we transform the hypergraph between the sets of users, tags, and resources into an undirected, weighted, tripartite graph. On this graph, we apply a version of PageRank that takes into account the edge weights.

First we convert the folksonomy $\mathbb{F} = (U, T, R, Y)$ into an *un*directed tripartite graph $\mathbb{G}_\mathbb{F} = (V, E)$ as follows.

1. The set V of nodes of the graph consists of the disjoint union of the sets of tags, users and resources: $V = U \dot\cup T \dot\cup R$. (The tripartite structure of the graph can be exploited later for an efficient storage of the – sparse – adjacency matrix and the implementation of the weight-spreading iteration in the FolkRank algorithm.)
2. All co-occurrences of tags and users, users and resources, tags and resources become undirected, weighted edges between the respective nodes: $E = \{\{u, t\}, \{t, r\}, \{u, r\} \mid (u, t, r) \in Y\}$, with each edge $\{u, t\}$ being weighted with $|\{r \in R : (u, t, r) \in Y\}|$, each edge $\{t, r\}$ with $|\{u \in U : (u, t, r) \in Y\}|$, and each edge $\{u, r\}$ with $|\{t \in T : (u, t, r) \in Y\}|$.

The original formulation of PageRank [1] reflects the idea that a page is important if there are many pages linking to it, and if those pages are important themselves. The distribution of weights can thus be described as the fixed point of a weight passing scheme on the web graph. We employ the same underlying principle for our ranking scheme in folksonomies. The basic notion is that a resource which is tagged with important tags by important users becomes important itself. The same holds, symmetrically, for tags and users. Thus we have a graph of vertices which are mutually reinforcing each other by spreading their weights.

Like PageRank, we employ the random surfer model, a notion of importance for web pages that is based on the idea that an idealized random web surfer normally follows hyperlinks, but from time to time randomly jumps to a new webpage without following a link. Formally, we spread the weight as follows: $\vec{w} \leftarrow dA\vec{w} + (1 - d)\vec{p}$ where A is the row-stochastic[5] version of the adjacency matrix of $\mathbb{G}_\mathbb{F}$, $\vec{p}$ is the random surfer component, and $d \in [0, 1]$ is a constant which controls the influence of the random surfer.

Usually, one will set $\vec{p}$ as the vector where all values equal 1. In order to compute personalized PageRanks, however, $\vec{p}$ can be used to express user preferences by giving a higher weight to the components which represent the user's preferred web pages.

We call the iteration according to the assignment above– until convergence is achieved – the *Adapted PageRank* algorithm. Note that, if $||\vec{w}||_1 = ||\vec{p}||_1$ holds,[6] the sum of the weights in the system will remain constant.

As the graph $G_\mathbb{F}$ is undirected, part of the weight that went through an edge at moment t will flow back at $t + 1$. The results are thus rather similar (but identical only if $d = 1$) to a ranking that is simply based on edge degrees. The reason for applying the more expensive PageRank approach nonetheless is that its random surfer vector allows for topic-specific ranking, as we will discuss in the next section.

3.2 Results for Adapted PageRank

We have evaluated the Adapted PageRank on the del.ico.us dataset. As there exists no 'gold standard ranking' on these data, we evaluated our results empirically.

First we ran the algorithm with $d = 1$. We obtained the highest ranks for the tags, followed by the users, and the resources. This ranking provides an overview over the content of del.ico.us. The most important tag is "system:unfiled" which is used to indicate that a user did not assign any tag to a resource. It is followed by "web", "blog", "design" etc. The resource ranking shows that Web 2.0 web sites like Slashdot, Wikipedia, Flickr, and a del.icio.us related blog appear in top positions. This is not surprising, as early users of del.ico.us are likely to be interested in Web 2.0 in general.

When using the random surfer vector to express user-specific preferences (e. g., by giving a considerably higher weight to a tag like 'boomerang'), we observed that although tags, users, and resources that are related to this preference are now ranked higher in the result, many of the general results mentioned above still hold the top positions. We ran this experiment with several settings of the preference vector, always with similar results. (For details see [3]). Apparently the preference vector is not strong enough to overcome the global graph structure.

[5] I. e., each row of the matrix is normalized to 1 in the 1-norm.

[6] ... and if there are no rank sinks – but this holds trivially in our graph $G_\mathbb{F}$.

4 FolkRank – Topic-Specific Ranking in Folksonomies

In order to reasonably focus the ranking around the topics defined in the preference vector, we have developed a differential approach, which compares the resulting rankings with and without preference vector. This resulted in our new *FolkRank* algorithm.

4.1 The FolkRank Algorithm

The FolkRank algorithm computes a topic-specific ranking in a folksonomy as follows:

1. The preference vector $\vec{p}$ is used to determine the topic. It may have any distribution of weights, as long as $||\vec{w}||_1 = ||\vec{p}||_1$ holds. Typically a single entry or a small set of entries is set to a high value, and the remaining weight is equally distributed over the other entries. Since the structure of folksonomies is symmetric, we can define a topic by assigning a high value to either one or more tags and/or one or more users and/or one or more resources.
2. Let $\vec{w_0}$ be the fixed point from the iteration with $d = 1$.
3. Let $\vec{w_1}$ be the fixed point from the iteration with $d < 1$.
4. $\vec{w} := \vec{w_1} - \vec{w_0}$ is the final weight vector.

Thus, we compute the winners and losers of the mutual reinforcement of resources when a user preference is given, compared to the baseline without a preference vector. We call the resulting weight $\vec{w}[x]$ of an element x of the folksonomy the *FolkRank* of x.

Whereas the Adapted PageRank provides one global ranking, independent of any preferences, FolkRank provides one topic-specific ranking for each given preference vector. Note that a topic can be defined in the preference vector not only by assigning higher weights to specific tags, but also to specific resources and users. These three dimensions can even be combined in a mixed vector. Similarly, the ranking is not restricted to resources, it may as well be applied to tags and to users. We will show below that indeed the rankings on all three dimensions provide interesting insights.

4.2 Comparing FolkRank with Adapted PageRank

To analyse the proposed FolkRank algorithm, we generated rankings for several topics, and compared them with the ones obtained from Adapted PageRank. We will here discuss one set of search results, for the tag 'boomerang'. Our other experiments all provided similar results.

The top leftmost part of Table 1 contains the ranked list of tags according to their weights from the Adapted PageRank by using $d = 0.625$ and 5 as a weight for the tag "boomerang" in the preference vector $\vec{p}$, while the other elements were given a weight of 0. As expected, the tag "boomerang" holds the first position while tags like "shop" or "wood" which are related are also under the Top 20. The tags "software", "java", "programming" or "web", however, are on positions 4 to 7, but have nothing to do with "boomerang". The only reason for their showing up is that they are frequently used in del.icio.us. The right column in Table 1 contains the results of our FolkRank algorithm, again for the tag "boomerang". Intuitively, this ranking is better, as the globally frequent words disappear and related words like "wood" and "construction" are ranked higher.

A closer look reveals that this ranking still contains some unexpected tags; "kassel" or "rdf" are for instance not obviously related to "boomerang". An analysis of the user ranking (not displayed) explains this fact. The top-ranked user is "schm4704", and he has indeed many bookmarks about boomerangs. A FolkRank run with preference weight 5 for user "schm4704" shows his different interests, see the right column in Table 2. His main interest apparently is in boomerangs, but other topics show up as well. In particular, he has a strong relationship to the tags "kassel" and

Table 1: Ranking results with preference for the tag "boomerang" for the tags (left: Adapted PageRank, right: FolkRank for tags) and for the URLs (bottom: FolkRank)

Tag	ad. PRank
boomerang	0,4036883
shop	0,0069058
lang:de	0,0050943
software	0,0016797
java	0,0016389
programming	0,0016296
web	0,0016043
reference	0,0014713
system:unfiled	0,0014199
wood	0,0012378
kassel	0,0011969
linux	0,0011442
construction	0,0011023
plans	0,0010226
network	0,0009460
rdf	0,0008506
css	0,0008266
design	0,0008248
delicious	0,0008097
injuries	0,0008087
pitching	0,0007999

Tag	FolkRank
boomerang	0,4036867
shop	0,0066477
lang:de	0,0050860
wood	0,0012236
kassel	0,0011964
construction	0,0010828
plans	0,0010085
injuries	0,0008078
pitching	0,0007982
rdf	0,0006619
semantic	0,0006533
material	0,0006279
trifly	0,0005691
network	0,0005568
webring	0,0005552
sna	0,0005073
socialnetworkanalysis	0,0004822
cinema	0,0004726
erie	0,0004525
riparian	0,0004467
erosion	0,0004425

Url	FolkRank
http://www.flight-toys.com/boomerangs.htm	0,0047322
http://www.flight-toys.com/	0,0047322
http://www.bumerangclub.de/	0,0045785
http://www.bumerangfibel.de/	0,0045781
http://www.kutek.net/trifly_mods.php	0,0032643
http://www.rediboom.de/	0,0032126
http://www.bws-buhmann.de/	0,0032126
http://www.akspiele.de/	0,0031813
http://www.medco-athletics.com/.../elbow_shoulder_injuries/	0,0031606
http://www.sportsprolo.com/...pitching%20injuries.htm	0,0031606
http://www.boomerangpassion.com/english.php	0,0031005
http://www.kuhara.de/bumerangschule/	0,0030935
http://www.bumerangs.de/	0,0030935
http://s.webring.com/hub?ring=boomerang	0,0030895
http://www.kutek.net/boomplans/plans.php	0,0030873
http://www.geocities.com/cmorris32839/jonas_article/	0,0030871
http://www.theboomerangman.com/	0,0030868
http://www.boomerangs.com/index.html	0,0030867
http://www.lmifox.com/us/boom/index-uk.htm	0,0030867
http://www.sports-boomerangs.com/	0,0030867
http://www.rangsboomerangs.com/	0,0030867

Table 2: Ranking results with preference for user "schm4704" for the tags (left: Adapted PageRank, right: FolkRank)

Tag	ad. PRank
boomerang	0,0093549
lang:ade	0,0068111
shop	0,0052600
java	0,0052050
web	0,0049360
programming	0,0037894
software	0,0035000
network	0,0032882
kassel	0,0032228
reference	0,0030699
rdf	0,0030645
delicious	0,0030492
system:unfiled	0,0029393
linux	0,0029393
wood	0,0028589
database	0,0026931
semantic	0,0025460
css	0,0024577
social	0,0021969
webdesign	0,0020650
computing	0,0020143

Tag	FolkRank
boomerang	0,0093533
lang:de	0,0068028
shop	0,0050019
java	0,0033293
kassel	0,0032223
network	0,0028990
rdf	0,0028758
wood	0,0028447
delicious	0,0026345
semantic	0,0024736
database	0,0023571
guitar	0,0018619
computing	0,0018404
cinema	0,0017537
lessons	0,0017273
social	0,0016950
documentation	0,0016182
scientific	0,0014686
filesystem	0,0014212
userspace	0,0013490
library	0,0012398

"rdf". When a community in del.ico.us is small (such as the boomerang community), already a single user can thus provide a strong bridge to other communities, a phenomenon that is equally observed in small social communities.

A comparison of the FolkRank ranking for user "schm4704" with the Adapted PageRank result for him (right column) confirms the initial finding from above, that the Adapted PageRank ranking contains many globally frequent tags, while the FolkRank ranking provides more personal tags. While the differential nature of the FolkRank algorithm usually pushes down the globally frequent tags such as "web", though, this happens in a differentiated manner: FolkRank will keep them in the top positions, *if* they are indeed relevant to the user under consideration. This can be seen for example for the tags "web" and "java". While the tag "web" appears in schm4704's tag list – but not very often, "java" is a very important tag for that user. This is reflected in the FolkRank ranking: "java" remains in the Top 5, while "web" is pushed down in the ranking.

The ranking of the resources for the tag "boomerang" given at the bottom of Table 1 also provides interesting insights. As shown in the table, many boomerang related web pages show up (their topical relatedness was confirmed by a boomerang aficionado). Comparing the Top 20 resources for "boomerang" with the Top 20 resources given by the "schm4704" ranking (not shown here), there is no "boomerang" web page in the latter. This can be explained by analysing the tag distribution of this user. While "boomerang" is the most frequent tag for this user, in del.icio.us, "boomerang" appears rather infrequently. The first boomerang web page in the "schm4704" ranking is the 21st URL (i. e., just outside the listed TOP 20). Thus, while the tag "boomerang" itself dominates the tags of this user, in the whole, the semantic web related tags and resources prevail. This demonstrates that while the user "schm4704" and the tag "boomerang" are strongly correlated, we can still get an overview of the respective related items which shows several topics of interest for the user.

This example– as well as the other experiments we performed (see also [3]) – show that FolkRank provides good results when querying the folksonomy for topically related elements. Overall, our experiments indicate that topically related items can be retrieved with FolkRank for any given set of highlighted tags, users and/or resources.

As detailed above, our ranking is based on tags only, without regarding any inherent features of the resources at hand. This allows to apply FolkRank to search for pictures (e. g., in flickr) and other multimedia content, as well as for all other items that are difficult to search in a content-based fashion. The same holds for intranet applications, where in spite of centralized knowledge management efforts, documents often remain unused because they are not hyperlinked and difficult to find. Full text retrieval may be used to find documents, but traditional IR methods for ranking without hyperlink information have difficulties finding the most relevant documents from large corpora.

5 Conclusion and Outlook

In this paper, we have argued that enhanced search facilities are vital for emergent semantics within folksonomy-based systems. We presented a formal model for folksonomies, the *FolkRank* ranking algorithm that takes into account the structure of folksonomies, and evaluation results on a large-scale dataset. In the long version [3] of this paper, we discuss also how to use FolkRank for generating recommendations.

The FolkRank ranking scheme has been used in this paper to generate personalized rankings of the items in a folksonomy. We have seen that the top folksonomy elements which are retrieved by FolkRank tend to fall into a coherent topic area, e.g. "boomerang". This leads naturally to the idea of extracting *communities of interest* from the folksonomy, which are represented by their top tags and the most influential persons and resources. If these communities are made explicit, interested users can find them and participate, and community members can more easily get to know each other and learn of others' resources.

Acknowledgement. Part of this research was funded by the EU in the Nepomuk project (FP6-027705).

References

[1] Sergey Brin and Lawrence Page. The Anatomy of a Large-Scale Hypertextual Web Search Engine. *Computer Networks and ISDN Systems*, 30(1-7):107–117, April 1998.

[2] B. Ganter and R. Wille. *Formal Concept Analysis: Mathematical foundations*. Springer, 1999.

[3] Andreas Hotho, Robert Jäschke, Christoph Schmitz, and Gerd Stumme. Information retrieval in folksonomies: Search and ranking. In York Sure and John Domingue, editors, *The Semantic Web: Research and Applications*, volume 4011 of *LNAI*, pages 411–426, Heidelberg, June 2006. Springer.

[4] Jon M. Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM*, 46(5):604–632, 1999.

[5] F. Lehmann and R. Wille. A triadic approach to formal concept analysis. In G. Ellis, R. Levinson, W. Rich, and J. F. Sowa, editors, *Conceptual Structures: Applications, Implementation and Theory*, volume 954 of *Lecture Notes in Computer Science*. Springer, 1995.

[6] S. Staab, S. Santini, F. Nack, L. Steels, and A. Maedche. Emergent semantics. *Intelligent Systems, IEEE [see also IEEE Expert]*, 17(1):78–86, 2002.

[7] L. Steels. The origins of ontologies and communication conventions in multi-agent systems. *Autonomous Agents and Multi-Agent Systems*, 1(2):169–194, October 1998.

[8] Gerd Stumme. A finite state model for on-line analytical processing in triadic contexts. In Bernhard Ganter and Robert Godin, editors, *ICFCA*, volume 3403 of *Lecture Notes in Computer Science*, pages 315–328. Springer, 2005.

[9] R. Wille. Restructuring lattice theory: An approach based on hierarchies of concepts. In I. Rival, editor, *Ordered Sets*, pages 445–470. Reidel, Dordrecht-Boston, 1982.

[10] W. Xi, B. Zhang, Y. Lu, Z. Chen, S. Yan, H. Zeng, W. Ma, and E. Fox. Link fusion: A unified link analysis framework for multi-type interrelated data objects. In *Proc. 13th International World Wide Web Conference*, New York, 2004.

Contextual Retrieval in Knowledge Intensive Business Environments

Mark Kröll[1], **Andreas S. Rath**[1], **Michael Granitzer**[1], **Stefanie Lindstaedt**[1], **Klaus Tochtermann**[1,2]
Know-Center[1] & Knowledge Management Institute TU Graz[2]
A-8010, Graz, Austria
{mkroell, arath, mgrani, slind, ktochter}@know-center.at

Abstract

Knowledge-intensive work plays an increasingly important role in organisations of all types. This work is characterized by a defined input and a defined output but not the way how to transform the input to an output. Within this context, the research project DYONIPOS aims at encouraging the two crucial roles in a knowledge-intensive organization - the process executer and the process engineer. Ad-hoc support will be provided for the knowledge worker by synergizing the development of context sensitive, intelligent, and agile semantic technologies with contextual retrieval. DYONIPOS provides process executers with guidance through business processes and just-in-time resource support based on the current user context, that are the focus of this paper.

1 Introduction

Workflow Management Systems (WFMS) have become quite popular in organizations, because they promise to solve the problems arising from their complex organizational processes. This significant contribution of WFMS to increase the productivity is generally accepted [Riss, 2005].

In spite of the fact that the most important key feature of WFMS that has been identified is the flexibility to deal with changes [van der Aalst *et al.*, 1998; Ellis *et al.*, 1995] van der Aalst and Weske [van der Aalst and Weske, 2005] reference nine articles indicating that Workflow Management Systems are still too restrictive. The usual modeling process where business processes are designed by a process engineer based on interviews and observations of work practices can be seen as a top-down approach. In contrast to this one is the bottom-up approach, referred to as process mining [van der Aalst and Weijters, 2004; Wen *et al.*, 2004; Maruster and Bosch, 2002] where the process model can be derived from workflow, task or/and event logs. In order to enhance the monitored data stored in the logs to tasks or even workflows, innovative information retrieval approaches and mining techniques are needed. The extracted workflow information can be used to model a guide which not just facilitates the work process but also enhances the work quality by just-in-time information retrieval based on the current user, work and organizational context. The consideration of the context in the information retrieval step is on the one hand a big challenge but on the other hand offers the possibility of significant quality improvements [Fuhr, 2005] and ad-hoc accuracy of the results.

This leads us to the objective of this paper which is the presentation of the research project DYONIPOS (DYnamic ONtology based Integrated Process OptimiSation). DYONIPOS strives for encouraging the two crucial roles in a knowledge-intensive organization - the process executer and the process engineer - by synergizing the development of context sensitive, intelligent, and agile semantic technologies with contextual retrieval. The approach of DYONIPOS incorporates the development of solutions based on automatic and semi-automatic knowledge management methods and technologies such as knowledge discovery, semantic systems, and knowledge flow analysis. For a general overview of the DYONIPOS project followed by a detailed description of the advanced features for the process executer and the process engineer we refer to [Tochtermann *et al.*, 2006].

The paper is structured as follows: Section 2 stresses the need for carrying out DYONIPOS and underlines the motivation. The following section outlines the steps DYONIPOS has to pass through in order to provide high quality ad-hoc information to the process executer. In Section 4 is briefly described which algorithms are intended to be used. The paper closes with an overview of the current project state and the list of references.

2 Overview

The goal of DYONIPOS is to solve the dilemma of the organizational need for standardization and control on the one hand and on the other hand the essential freedom of a knowledge worker in his daily job. The research project DYONIPOS aims at mitigating this dilemma by developing context sensitive, intelligent, and agile semantic technologies.

In the business process environment, DYONIPOS will support both the process executer and the process engineer. Process executers are provided with support to find, perform, and record ad hoc processes within their work environment such that the ad hoc process retrieval, application, and definition take place within the executer's current work context. Process engineers will be enabled to review and analyze recorded ad hoc processes, compare them to the standardized processes and automatically enhance them.

The emphasized part of Figure 1 (left part) shows an overview of the DYONIPOS system. It illustrates how DYONIPOS provides information and support for the process executer and the process engineer. The process executer interacts with the system and obtains ad-hoc

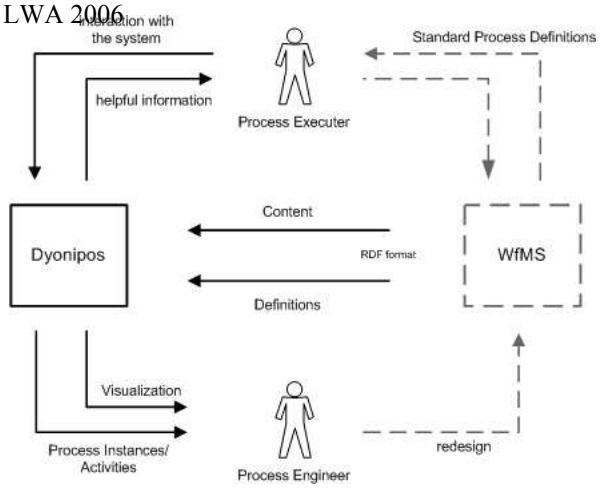


Figure 1: The scope of DYONIPOS

information based on his recent interaction. The process engineer benefits from the preprocessing of identified process structures.

In this paper we mainly focus on the process executer, i.e., featuring guidance through the daily work and providing supportive resources. Resource retrieval is intended to be just-in-time based upon automatic extraction of the context. In the following section we are going to present our ideas and intended procedures.

3 Project Approach

This section is structured in the manner of a question-and-answer game. The question sequence reflects more or less the steps DYONIPOS has to pass through in order to provide the process executer with the promised features. We answer these questions by presenting our approach. To avoid starting each time with "How do we" these three words are replaced by dots.

3.1 ... obtain the context?

First of all we introduce a short synopsis of the required steps to collect information of the process executer's actions and thus to obtain the context [Dey *et al.*, 2001; Coutaz *et al.*, 2005]. It starts with the recording of all events, i.e., the entire user interaction. Events belonging to a logical unit are grouped together into event blocks. Corresponding event blocks form semantic sets and are eventually assigned to the knowledge worker's tasks. Hence one task is represented as a sequence of event blocks. Since event blocks are transfered into graphs, a task can also be modelled as a large graph containing all event blocks in form of subgraphs.

Each level of granularity (events, event blocks and tasks) provides a different representation of the data regarding the semantic quality (see Figure 2). A sequence of events, that is eventually combined to an event block, provides structural information, i.e., how something is done, as well as information due to the content, i.e., what is done. Semantic quality is a measure of this enrichment and is permanently enhanced by passing through the layers and ending at the task level. In the following the individual layers are described in more detail. The data on which DYONIPOS operates consists of the monitored interactions between the user and his computer. Here we use a key logger program, referred to as *event logger*, which records and logs all *events* that occur on the user's computer - a quite similar approach as described by [Fenstermacher, 2005]. Events are user inputs, e.g., mouse movement, mouse clicks, starting a program or creating a folder or file and the reactions to these inputs on the system's side. The sum of all recorded events are stored in the so called *event log*. To ensure security and privacy for the user he has the ability to modify the event log and to delete events from the event log. An important basis for our work in this area build the results from the MISTRAL project [Tochtermann *et al.*, 2005] that aims at extracting semantic concepts from text, audio and video data. It is even conceivable to incorporate into and further process user conversations within the DYONIPOS project.

The knowledge worker is producing quite a bit of data throughout his daily work. Since the event logger is monitoring on a fine granular basis huge amount of data is recorded. To cope with this data mass we perform the following two steps, filtering and relation analysis. Separating relevant from irrelevant data reduces the size of the event log and enhances the output quality for the relation analysis step. In the relation analysis step the events stored in the event log are combined to so called *event blocks* as depicted in Figure 2. Event blocks are built based on predefined static rules. These static rules are a mapping of a set of events to an event block. An example of such a static mapping can be as follows: The user moves the mouse over a program icon, double clicks this icon and the system starts the program. This set of events can be combined to an event block called *starting a program*. An interesting open question here is if it is possible to automatically find a mapping based on the data in the event log, i.e., automatically generating relation rules.

Since not all events and therefore event blocks of a knowledge worker's daily work can be captured automatically the user has the ability to manually add event blocks. Event blocks of this kind can be meeting appointment, talk with a colleague on the corridor, or signing a report.

In Subsection 3.4 it is described how similar event blocks are grouped together into semantic sets. The resulting set represents one task of the process executer. It is intended that the labeling of that task is done automatically by comparing the set of event blocks to other sets. The comparison of entire sets of event blocks is most likely unwise. Thus we select only a few of them (not necessarily sequential) and try to find corresponding event blocks in other sets. Section 4 deals with graph matching algorithms that can be applied to carry out similarity measurements between event blocks.

A high-quality semantic description of the process executer's tasks is thus obtained and is going to be used for further processing. The semantic hierarchy containing three layers is illustrated in Figure 2. Hence, by means of process mining a larger workflow emerges by concatenating individual tasks that were conducted by a number of different knowledge workers.

3.2 ... represent the context?

In DYONIPOS we focus on the knowledge worker's context, i.e., the user context. The user context describes who

the user is (organizational context), what he does (work context), how he does it (behavioral context), with whom he is collaborating (social context), and which and what kind of resources he uses (document context). Further contexts that are addressed in DYONIPOS are the process context, that describes the position of the knowledge worker in a business process and the environmental context that captures the nature of the location of the knowledge worker, e.g., computer desktop, meeting room or corridor.

To obtain as much information as possible about the various contexts we rely on recording all user interactions with the system as stated in Subsection 3.1. The collected data is going to be represented as an RDF[1] graph. This representation allows an incorporation and a further processing of relations between graph entities. Certain steps (event blocks) within a task can thus be regarded as graphs where relations between persons, documents and other resources can be easily embedded. These event block graphs are then merged to form a larger graph that represents the entire task and thus part of the user context.

All the different contexts will be related to each other to ensure highly supportive information providing for the knowledge worker. A further application of the contextual information is to identify different and similar tasks. The idea here is to analyze the state of the user context for finding deviations. These deviations could be detectors for switching from one task to another. Identifying context switches could potentially be used as indicators for an update of the provided supportive information. Since contextual retrieval is rather application specific [Fuhr, 2005] further research has to be done in applying contextual retrieval in the area of knowledge-intensive business environments.

3.3 ... store the context?

The knowledge worker's privacy is ensured by law. Thus a natural dilemma arises when trying to gather as much information as possible about the worker's interactions with the system while abiding by the law.

Still, to be able to provide guidance and resources, we have to know about the context of the process executer. As stated in Subsection 3.1, no user interaction remains hidden from the system. Nevertheless data that is going to be stored needs explicit permission. Moreover event blocks are transfered into an abstract form containing the essential data in an encrypted way. The level of encryption is tunable and could be a term vector representation or a hash coding.

3.4 ... exploit the context?

The knowledge worker is not bound to remain within a given task, i.e., executing one task after the other. Switching between tasks might be necessary and even more efficient. In other words, we are given lots of event blocks that have to be grouped together according to their topic as shown in Figure 2. Event blocks that exhibit similar content are identified by analyzing textual information, e.g., which documents were written or read. The degree of similarity indicates the affiliation to a certain set of event blocks. Standard text mining algorithms provide us with the means to extract keywords and compare textual contents.

[1]Resource Description Framework: http://www.w3.org/RDF/

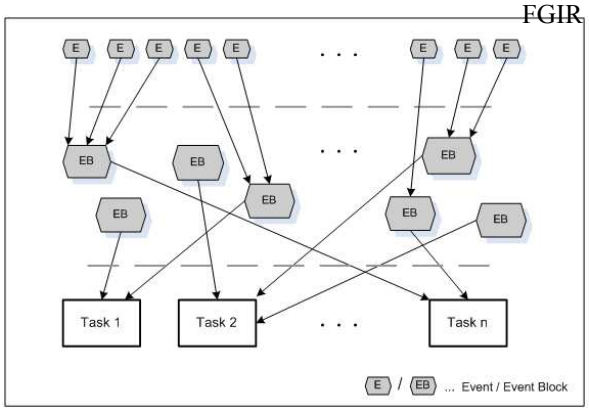


Figure 2: Three semantic layers are shown from the bottom up. The next level can be reached by sensibly grouping together elements from the same layer.

DYONIPOS guides the knowledge worker through given tasks, i.e., suggesting successive actions that have already been performed by other co-workers. Therefore it is necessary to assign the event blocks to the right task. We are going to exploit information regarding the content, e.g., written documents, read reports, visited websites, as well as structural information, e.g., which experts were consulted, which documents were searched for. Similar event blocks of co-workers are identified by applying text mining and graph matching algorithms. Section 4 deals with attempts to exploit the graph structure of event blocks to obtain a measure of similarity. Once similar event blocks are found a possible course of action can be suggested by presenting subsequent event blocks of other process executers.

The guidance for a knowledge worker ends with a recommendation for the next step in the overall process. Therefore the current task has to be assigned to a process, i.e., the position within the overall process has to be figured out. Having assigned the current task, the next step in the process can be identified.

3.5 ... benefit?

The DYONIPOS system provides the process executer with various kinds of resources, e.g., documents other co-workers have read or written performing the same task, useful websites dealing with similar contents, suggesting experts inside or outside the company that could give helpful advice. Resource delivery is based on comparing event blocks as stated in Subsection 3.4. Once similar event blocks of other co-workers have been identified, resources used or generated by them can be proposed. However, in our opinion the presentation of information must not be done prematurely. First the current actions of the knowledge worker have to be assigned to an existing group of event blocks. Resources that have been allocated in the meantime for that group of event blocks are then visualized. If the knowledge worker switches tasks, current resource propositions disappear and DYONIPOS identifies the new content affiliation before providing further resources.

DYONIPOS does not claim to be merely more efficient at retrieving information as traditional search engines. The overall objective is that process executers actually use more information than they would with search engines since there is no effort in obtaining resources. Due to the effort-

less accessibility the knowledge worker can incorporate additional resources into his work thus improving the overall work quality.

DYONIPOS intends to follow the same policy as JITIR agents [Rhodes, 2000], i.e., proactivity, presentation of retrieved information in an accessible yet nonintrusive manner, and awareness of the context.

4 Algorithms

As stated above, the internal representation of data will be in form of graphs. In general, given a graph that was constructed by the interaction of the process executer with the system, we would like to find other, similar graphs. In the following it is briefly described which algorithms for measuring the similarity between graphs are intended to be used. Based on the similarity measure, graph classification is carried out. Previous work outlined in [Lux *et al.*, 2006] is utilized for graph retrieval.

Relations between graph entities provide valuable information for mining tasks. Thus, the relatively nascent research areas *Link Mining* [Getoor, 2003] and *Graph Mining* [Washio and Motoda, 2003] successfully exploit the topological view of structured data. Regarding the classification of graphs, relational learning [Mitchell, 1997], finding frequently appearing substructures in graphs [Inokuchi *et al.*, 2000] and kernel methods such as Support Vector Machines [Vapnik, 1995] can be applied. In our work, we focus on kernels for structured data [Gärtner, 2003] and kernel methods [Schölkopf and Smola, 2001].

Convolutional kernels were introduced by [Haussler, 1999] providing a general framework for handling discrete data structures by kernel methods. [Kashima and Inokuchi, 2002; Kashima *et al.*, 2003] concentrated on the construction of graph kernels. Their graph kernel performs a random walk on the vertex product graph of two graphs. The idea behind this kernel is including local information, i.e., taking into account similar edges and vertices of the vicinity.

Another framework of kernel function related with graph structures is called diffusion kernel that was introduced by [Kondor and Lafferty, 2002]. The main difference to the above mentioned approach is that diffusion kernels do not compare two graphs but rather return a similarity measurement between two objects that are represented as vertices of a graph in the input space. Diffusion kernels were applied to document classification [Kandola *et al.*, 2002] where the documents are represented as vertices in a graph.

[Joachims *et al.*, 2001] proposes a combination of kernels each dealing with a different aspect of the data. One kernel deals with the content of objects and another kernel takes the link structure between the objects into account.

Ideas coming outside the world of kernels are considered as well. [Blondel *et al.*, 2004] introduces a concept of similarity between vertices of directed graphs.

5 Project State

The DYONIPOS project is a two year project (March 2006- February 2008) which has started a few months ago. Currently we are implementing semantic technologies and IR methods. The objective is to have a functional environment with which we can match the power of the technologies with our requirements. Furthermore we are analyzing technological synergies with other similar research projects such as AVALON (http://www.iwm.tugraz.at/research/projects/avalon) and MISTRAL (http://www.mistral-project.at).

6 Acknowledgements

The project results have been developed in the DYONIPOS project (DYnamic ONtology based Integrated Process OptimiSation). DYONIPOS is financed by the Austrian Research Promotion Agency (www.ffg.at) within the strategic objective FIT-IT under the project contract number 810804/9338.

The Know-Center is funded by the Austrian Competence Center program Kplus under the auspices of the Austrian Ministry of Transport, Innovation and Technology (http://www.ffg.at), by the State of Styria and by the City of Graz.

References

[Blondel *et al.*, 2004] Vincent D. Blondel, Anahi Gajardo, Maureen Heymans, Pierre Senellart, and Paul Van Dooren. A measure of similarity between graph vertices: Applications to synonym extraction and web searching. *SIAM Rev.*, 46(4):647–666, 2004.

[Coutaz *et al.*, 2005] Joelle Coutaz, James L. Crowley, Simon Dobson, and David Garlan. Context is key. *Commun. ACM*, 48(3):49–53, 2005.

[Dey *et al.*, 2001] A. Dey, D. Salber, and G. Abowd. A conceptual framework and a toolkit for supporting the rapid prototyping of context-aware applications, 2001.

[Ellis *et al.*, 1995] C. Ellis, K. Keddara, and G. Rozenberg. Dynamic change within workflow systems. In *COCS '95: Proceedings of conference on Organizational computing systems*, pages 10–21, 1995.

[Fenstermacher, 2005] Kurt D. Fenstermacher. Revealed processes in knowledge management. In *Wissensmanagement*, pages 397–400, 2005.

[Fuhr, 2005] N. Fuhr. Information retrieval — from information access to contextual retrieval. In M. Eibl, Ch. Wolff, and Ch. Womser-Hacker, editors, *Designing Information Systems. Festschrift für Jürgen Krause*, pages 47–57. UVK Verlagsgesellschaft, 2005.

[Getoor, 2003] L. Getoor. Link mining: a new data mining challenge. *SIGKDD Explor. Newsl.*, 5(1):84–89, 2003.

[Gärtner, 2003] Thomas Gärtner. A survey of kernels for structured data. *SIGKDD Explor. Newsl.*, 5(1):49–58, 2003.

[Haussler, 1999] D. Haussler. Convolution kernels on discrete structures, 1999.

[Inokuchi *et al.*, 2000] Akihiro Inokuchi, Takashi Washio, and Hiroshi Motoda. An apriori-based algorithm for

mining frequent substructures from graph data. In *Principles of Data Mining and Knowledge Discovery*, pages 13–23, 2000.

[Joachims *et al.*, 2001] Thorsten Joachims, Nello Cristianini, and John Shawe-Taylor. Composite kernels for hypertext categorisation. In Carla Brodley and Andrea Danyluk, editors, *Proceedings of ICML-01, 18th International Conference on Machine Learning*, 2001.

[Kandola *et al.*, 2002] J. Kandola, J. Shawe-Taylor, and N. Cristianini. On the application of diffusion kernel to text data. Technical report, Neurocolt, 2002.

[Kashima and Inokuchi, 2002] H. Kashima and A. Inokuchi. Kernels for graph classification. ICDM Workshop on Active Mining 2002, 2002.

[Kashima *et al.*, 2003] H. Kashima, K. Tsuda, and A. Inokuchi. Marginalized kernels between labeled graphs. In *Proceedings of the 20th International Conference on Machine Learning*, 2003.

[Kondor and Lafferty, 2002] Kondor and Lafferty. Diffusion kernels on graphs and other discrete structures. In *In Proc. 19th Intl. Conf. on Machine Learning*, 2002.

[Lux *et al.*, 2006] M. Lux, S. Meyer zu Eissen, and M. Granitzer. Graph retrieval with the suffix tree model. In *Proceedings of the Workshop on Text-Based Information Retrieval TIR 06*, 2006.

[Maruster and Bosch, 2002] Laura Maruster and Antal Van Den Bosch. Process mining: Discovering direct successors in process logs. 2002.

[Mitchell, 1997] Thomas M. Mitchell. *Machine Learning*. McGraw-Hill Higher Education, 1997.

[Rhodes, 2000] Bradley J. Rhodes. *Just-In-Time Information Retrieval*. PhD thesis, MIT Media Laboratory, Cambridge, MA, May 2000.

[Riss, 2005] Uwe V. Riss. Knowledge, action, and context: A process view on knowledge management. In *Wissensmanagement*, pages 555–558, 2005.

[Schölkopf and Smola, 2001] Bernhard Schölkopf and Alexander J. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press, Cambridge, MA, USA, 2001.

[Tochtermann *et al.*, 2005] K. Tochtermann, M. Granitzer, V. Sabol, and W. Klieber. Mistral: Service-orientierte cross-media techniken zur extraktion von semantik aus multimedia daten und deren anwendung. In *Proceedings Semantics 2005, Reich S., Güntner G., Pellegrini T., Wahler A. (Hrsg.) Trauner Verlag*, 2005.

[Tochtermann *et al.*, 2006] K. Tochtermann, D. Reisinger, M. Granitzer, and S. Lindstaedt. Integrating ad hoc processes and standard processes in public administrations. In *Proceedings of the OCG eGovernment Conference, Linz (Austria)*, 2006.

[van der Aalst and Weijters, 2004] W.M.P. van der Aalst and A.J.M.M. Weijters. Process mining: A research agenda. *Comput. Ind.*, 53(3):231–244, 2004.

[van der Aalst and Weske, 2005] W.M.P. van der Aalst and Mathias Weske. Case handling: A new paradigm for business process support. *Data Knowl. Eng.*, 53(2):129–162, 2005.

[van der Aalst *et al.*, 1998] W.M.P. van der Aalst, G. De Michelis, and C.A. Ellis. Workflow management: Net-based concepts, models, techniques, and tools (wfm'98). In *Computing Science Report 98/7*, 1998.

[Vapnik, 1995] Vladimir N. Vapnik. *The nature of statistical learning theory*. Springer-Verlag New York, Inc., New York, NY, USA, 1995.

[Washio and Motoda, 2003] Takashi Washio and Hiroshi Motoda. State of the art of graph-based data mining. *SIGKDD Explor. Newsl.*, 5(1):59–68, 2003.

[Wen *et al.*, 2004] Lijie Wen, Jianmq Wang, Zhe Wang, and Jiaguang Sun. A novel approach for process mining based on event types. 2004.

The Role of Information Retrieval in the Question Answering System IRSAW

Johannes Leveling

Fakultät für Mathematik und Informatik
Intelligent Information- and Communication Systems (IICS)
FernUniversität in Hagen
58095 Hagen, Germany
johannes.leveling@fernuni-hagen.de

Abstract

Information on the internet is a vast resource for question answering. As the amount of available information from web pages increases, novel methods for finding precise answers to user queries and questions must be found. Standard information retrieval methods are efficient, but often fail to provide a user with short, precise answers. A deep linguistic analysis of all information is time consuming, but it offers more advanced means to find answers to a user's question. Shallow natural language processing methods seem to work well on a limited range of questions, but they are not suitable for finding answers to more complex questions.

This paper describes work in progress on the question answering system IRSAW [1](Intelligent Information Retrieval on the Basis of a Semantically Annotated Web), a system that combines information retrieval with a deep linguistic analysis of texts to obtain answers to natural language questions. In IRSAW, different techniques for finding answers lead to different sets of answer candidates, which are then merged to produce a final answer.

The system's architecture and functionality are described before evaluation results of a first prototype are presented.

1 Introduction

The amount of information available on the WWW (world wide web) and information needs of users increase, yet it becomes harder to find relevant answers to questions. Pure information retrieval (IR) approaches fail to provide a user with short, precise answers to information requests. However, IR has managed to scale up with the amount of documents.

Applying natural language processing (NLP) methods to textual information does not scale very well, because a deep linguistic analysis is costly in terms of CPU cycles, but NLP offers a chance to find more precise answers.

Giving short, precise answers to user questions will become more important than returning collections of URLs (Uniform Resource Locators) or whole web pages to read. To this end, combining traditional IR and deep NLP seems to be a promising approach. This paper describes the work in progress on the IRSAW system (Intelligent Information Retrieval on the Basis of a Semantically Annotated Web), a question answering system combining IR approaches with methods for semantic retrieval and logic-based question answering. The project result will be a question answering system capable of answering natural language user questions on the basis of information available on the web.

The IR approach in IRSAW originates from the NLI-Z39.50, a natural language interface to information resources available on the internet [Leveling, 2006a]. It includes features such as blind feedback and query expansion with semantically related search terms and with constituents of compounds (compound nouns are written as one word in German). These methods have been evaluated at the Cross Language Evaluation Forum (CLEF 2003–2006) and achieved a performance of 0.3537 mean average precision (MAP) for the monolingual German domain specific task at CLEF 2006 [Leveling, 2006b].

Methods for question answering (QA) are part of InSicht, a question answering system using a deep linguistic analysis of queries and documents based on a semantic network representation. The InSicht subsystem was first evaluated at the monolingual German QA@CLEF (Question Answering at the Cross Language Evaluation Forum) task in 2004. For the 200 questions of the monolingual German QA@CLEF task in 2005, InSicht found 86 correct answers and 8 inexact answers [Hartrumpf, 2006b]. InSicht is highly oriented towards precision. Only a few inexact and wrong answers were given for the 200 test questions in 2005 and 2006. Furthermore, InSicht has been adapted for a search in web resources [Hartrumpf, 2006a]. Both approaches will be integrated into the IRSAW system.

Current state-of-the art systems often employ IR methods, passage retrieval, or shallow techniques separately or as a combination to pinpoint answers. Neumann et al. [Neumann and Xu, 2003; Neumann and Sacaleanu, 2004] investigate a similar approach, but rely heavily on redundancy of information on the web and on a more expensive preprocessing phase.

Ravichandran and Hovy [2002] suggested and implemented pattern matching based on surface structures (i.e., words) to find answers in the Webclopedia QA system. Since their approach also depends on a specialised and incomplete question/answer typology, it will not be portable

[1]The research presented in this paper was funded by the DFG – Deutsche Forschungsgemeinschaft as part of the project *Intelligent Information Retrieval on the Basis of a Semantically Annotated Web* (IRSAW; GZ: LIS 4 – 554975(2) Hagen, BiB 48 HGfu 02-01).

to open-domain QA.

Ahn, Jijkoun et al. [2006] propose generating several query streams, each returning a set of answers candidates. In contrast to this approach, IRSAW will not merge parallel answer streams, but will favour answer candidates originating from more sophisticated methods (NLP) to those from shallower approaches, such as pattern matching. However, different answer streams will be created in parallel by separate subsystems.

2 Architecture of IRSAW

IRSAW processes user questions in three phases, accessing three kinds of resources: two IR phases in which web search engines and local databases are accessed and a QA phase, in which a semantic network database is accessed. Figure 1 shows the architecture of the IRSAW system. During the first phase, the user question is transformed into an IR query and meta information such as the question type and the expected answer type is determined. The IR query is delivered to web search engines and web portals. Results from the web typically consist of pages with lists of URLs. The web contents (i.e., HTML pages, electronic documents and metadata for audio-visual data) referenced by these URLs are retrieved and converted into text.

In the second phase, the text passages from the web are segmented and indexed in one or more local databases. A local database serves several purposes: First, it is a cache containing texts with answers to previously asked questions and it can be accessed in parallel while the web search is initiated. Second, it is a mediating service with a uniform search interface to heterogeneous services: web systems typically differ in query syntax and in structural elements supported in queries (e.g. support for wildcards, phrases, proximity search). Furthermore, web services may differ in syntax or in the format in which answers are returned (i.e., hierarchical structures or simple lists of URLs). Finally, the local database provides access to units of textual information (text segments) of the same type or length (chapters, paragraphs, sentences, or phrases).

In the third phase of IRSAW, several methods are employed to pinpoint answers. In the InSicht subsystem, a linguistic parser analyses the text segments and semantically annotates them [Hartrumpf, 2003]. The parser returns the representation of the meaning of a text as a semantic network. The semantic representations of questions and texts are compared to intelligently find answers. In addition, a shallower technique creates a different answer stream by applying pattern matching to the answer candidates found in the second phase.

Evaluation of the first prototype of IRSAW is based on data provided for the QA@CLEF task in previous years. The data includes 600 question-answer pairs from 2003–2005 (200 per year) for which missing answers were manually added, and a document collection of 276.581 texts. In the evaluation of the IRSAW prototype, the document collection is a replacement for documents retrieved from the web. The following section describes question processing in IRSAW in more detail. In this paper, the focus is mainly on the IR phases to find answer candidates and pattern matching to find precise answers.

3 Question Processing

3.1 Creating an IR Query

The user poses a natural language question at the client interface, which initiates question processing. For instance an example question might be *"Where was Galileo Galilei born?"*[2]

The natural language question is transformed into an IR query for external web sources (see step 1 in Figure 1). In this process, the set of search terms for the IR query, S, is constructed and optional weights w for search terms are determined. The term weights are only of use for the local database and for search engines supporting a weighted search; they are ignored for the search, otherwise. The set of search terms is constructed from the empty set as follows:

- add all proper nouns and all quoted expressions to the set of search terms to S ($w = 1.0$)
- add all words in upper case (e.g. nouns) to S ($w = 0.9$)
- add all words in lower case (e.g. verbs, adjectives, adverbs) to S ($w = 0.7$)
- add all remaining words (e.g. numeric expressions, etc.) to S ($w = 0.5$)

For the example question, the IR query is *"Galileo_Galilei, born"* with term weights 1.0 and 0.7, respectively.

3.2 Question and Answer Type Classification

Question and answer types are calculated using a Naïve Bayes classifier trained on features representing the first N words of the question. Using $N = 3$ suffices to correctly identify the answer type for 575 of 600 (95.6%) questions in our test corpus. We followed the classification of answers for the QA@CLEF task, defining locations (LOC), persons (PER), organisations (ORG) temporal expressions (TIM), etc. For the example question given above, the answer type location (LOC) is determined.

Question types (see [Helbig, 2006]) include yes-no questions, essay questions, and questions starting with *WH*-words (*why*, *where*, *who*, *when*, ...). The question type will influence the length of the answer and what type of answer is returned.

3.3 Accessing Web Resources

The IR query is sent to external web resources (search engines) which return result pages containing URLs. All web contents referred to by an URL are retrieved and their contents are converted into text. Documents are preprocessed using a sentence and paragraph boundary detection [Grefenstette and Tapanainen, 1994] adapted to German. The resulting texts are then segmented into units and fed to the local database.

3.4 Accessing a Local Database

The IR query is also processed by the local database, which returns a ranked list of text segments. These segments represent answer candidates, i.e. they probably contain an answer to the stated question. If multiple local databases are employed, the local databases and external web resources will be accessed in parallel. Results from the web are added to and indexed in one single local database (using a Round Robin scheme). The question can be processed by

[2] Examples have been translated from German into English.

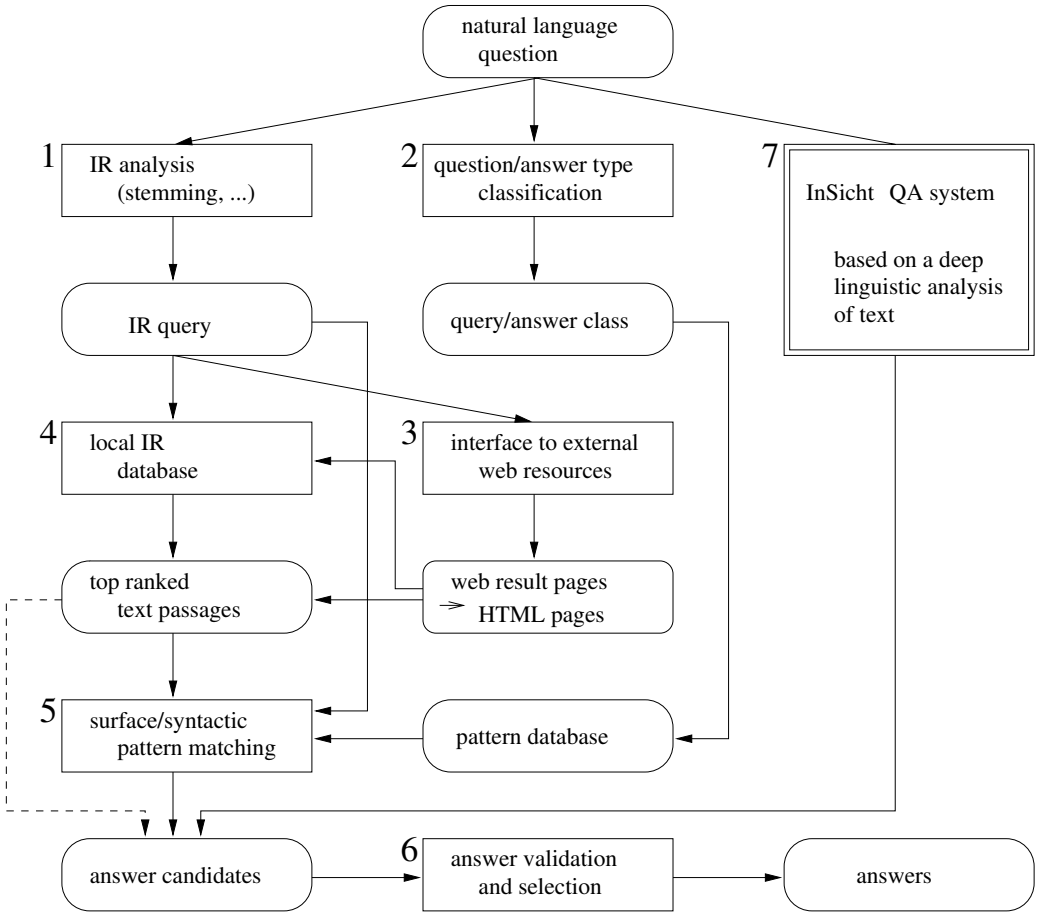


Figure 1: Architecture of the IRSAW system as a combination of IR methods and the QA subsystem InSicht. Arcs indicate data flow, rectangles represent processes, and ovals represent data.

all local databases which are not waiting for web search results and are not busy indexing new results while a web search is in progress.

A promising answer candidate for the example question would be the starting sentence in the Wikipedia article on Galilei: *"Galileo was born in Pisa, in the Tuscany region of Italy on February 15, 1564."* Note that the IR setup described so far already serves as a baseline QA system (indicated by the slashed arc in Figure 1): The top ranked answer candidates obtained from the local database often contain a precise answer if the length of the text unit is adequately chosen.

A pattern database was created using question-answer pairs from previous question answering tasks at CLEF. There were 200 questions for each each of the QA tasks at CLEF in 2003, 2004, and 2005. For these, 732 answers were found – for some questions a text text corpus does not contain a corresponding answer, but for other questions, different correct answers or paraphrases of the same answer are found. Building the pattern database did not involve other manual work such as annotating answers. In the near future, the training data will be largely extended. For a limited range of questions, question-answer pairs can be extracted from available information. One such example is the normalised biographical data about persons (PND – Personennamendatei, see [Hengel and Pfeifer, 2005]) contains information about where and when a person was born and died, and his or her name and professions.

The pattern database contains for each answer type (LOC, PER, ORG, etc.) a number of patterns created from the answer candidates found in the QA@CLEF newspaper corpus. Every answer candidate obtained using the IR query was tokenized and patterns were automatically extracted.

3.5 Pattern Matching

The shallow technique for finding answers in IRSAW is based on pattern matching. A pattern consists of a sequence of variables, symbols, and strings. Variables start with a leading "?". They represent the words at the start and end of a text unit as well as the answer string, matching with zero or more words (tokens). There are two types of symbols: part-of-speech tags (POS tags) from the *Stuttgart-Tübingen Tagset* (STTS) are assigned to words from closed categories (excluding nouns, verbs, adjectives, and adverbs). Other symbols are created from search terms in the IR query that are classified into lower case words (LWORD), upper case words (UWORD), proper nouns (NAME), and numeric expressions (NUM).

The expected answer type for the question is used as a key to find patterns in which the instantiation of the answer variable will be of the expected type. For example, data for the LOC answer type includes all patterns for which the answer variable is a location. Table 1 shows how a pattern for the LOC answer type with a context size of 4 tokens is extracted from the example sentence.

A context window containing a maximum of 5 tokens on each side of the answer variable was used to form

the patterns. The pattern matching returns an instantiation of the answer variable (answer string). For example, if the pattern training had been applied to data containing the example answer candidate (*"Galileo was born in Pisa, in the Tuscany region of Italy on February 15, 1564"*), the question (*"Where was Galileo Galilei born?"*), and the answer *"Pisa"*, the corresponding pattern would be *"?words1* NAME ?w0 LWORD appo ?answer+ $comma appo art ?w1 ?words2*"*.

Table 1: Constructing a pattern from the example sentence. The pattern consists of atomic symbols for POS tags, variables corresponding to one or more tokens and special symbols representing words and derived words from the question. The resulting pattern is *"?words1* NAME ?w0 LWORD appo ?answer+ $comma appo art ?w1 ?words2*"*. This pattern has been modified at the start and at the end with a variable matching zero or more tokens (+ denotes a sequence of one or more tokens (words); * a sequence of zero or more tokens).

Text	Tagged Text	LOC pattern
Galileo	NAME	NAME
was	?w0	?w0
born	LWORD	LWORD
in	appo	appo
Pisa	?answer+	?answer+
,	$comma	$comma
in	appo	appo
the	art	art
Tuscany	?w1	?w1
region	?w2	–
of	art	–
Italy	?w3	–
.	$colon	–

All patterns are applied to the top N answer candidates found ($N = 250$) found in the IR phase. Patterns corresponding to matches not containing an instantiation of the answer variable are removed from the list of useful patterns. The remaining patterns are added to the pattern database together with the key (the expected answer type) for lookup.

3.6 Merging, Validating, and Selecting Answers

Answer streams from different sources are merged by preferring answers from InSicht to answers found by the pattern matching. A de-duplication of answer candidates is performed on the answer stream produced by the shallow system. Answers are then ranked by cumulative frequency and the top answer is returned. In the example above, the instantiation of the answer variable *"?answer+"* would be *"Pisa"*.

3.7 The InSicht Subsystem

To produce a second answer stream, IRSAW interfaces to InSicht, a QA system employing a deep linguistic analysis based on a semantic network representation of question and textual information. InSicht has several advantageous characteristics:

- A deep syntactico-semantic analysis for documents and text.
- Independence from other document collection and independence from domains.
- Generation of answers from the semantic network representation of documents, i.e. answers are not extracted from the documents.

InSicht performs best when applied on syntactically correct texts (86 correct and 8 inexact answers were found for 200 questions at QA@CLEF 2005), but it will fail to produce a meaning representation (in this case, a semantic network) for malformed sentences. InSicht's syntactico-semantic parser is able to produce a complete semantic network for about 48.7% and a partial semantic network for for 20.4% of all sentences in the newspaper corpus. Hartrumpf gives an overview over common errors with the WOCADI parser [Hartrumpf, 2005], including the limited robustness of the parser and missing lexicon entries (although the parser relies on a large set of lexicons including full morphological and syntactico-semantic information). For sentences containing grammatical or spelling errors or conflated sentence parts originating from erroneous preprocessing, the parser often fails to produce a semantic network. Thus, InSicht will not to able to find many of those answers appearing in malformed sentences only.

In addition to parser errors and missing lexicon entries, the news articles in the test corpus often contain artefacts from preprocessing as well as metadata such as the date and time of the article, the name of the agency responsible, and the initials of the author conflated into the text. Grammatically, these sentences are not well-formed and thus, any deep linguistic analysis should fail to produce parse results.

However, one should assume that text fragments relevant to an IR query contain to some extent correct answers to questions on the query topic. Therefore, information retrieval methods can be employed a) to interface to IR engines to retrieve textual information for a deeper linguistic analysis, and b) to provide a more robust method to identify answer candidates (because higher ranked answer candidates are more likely to contain an answer).

4 Evaluation Results for the First Prototype

A first evaluation of the IRSAW setup was performed within the question answering task at QA@CLEF 2006. This task consists of finding answers for a set of 200 questions targeting a test corpus of newspaper articles. System answers are assessed manually for correctness. The test corpus contains 276.581 newspaper articles and newswires from the *Frankfurter Rundschau*, *Der Spiegel*, and *Die Schweizerische Depeschenagentur* from the years 1994 and 1995.

A sentence boundary detector and a tokenizer was applied to the test corpus and documents were split into single sentences and indexed in a local database (omitting the phase with web access). Then, question-answer pairs for the QA task in previous years were constructed from the MultiEight corpus [Magnini *et al.*, 2005], augmented manually and used as a training set to build the pattern database.

At QA@CLEF 2006, the pattern matching approach found 17 answers for the 200 test questions, while the method employing deep linguistic processing found correct 61 answers. In total, 64 correct answers were found (one additional answer found by the pattern matching was assessed as inexact). All of the 13 remaining answers found with pattern matching were correct as well. Table 2 shows accuracy and Mean Reciprocal Rank (MRR) for both runs submitted. This was our first approach to combine the deep processing with a shallower method and it leaves many chances for further improvements.

Table 2: Results of the IRSAW QA system for the QA@CLEF task 2006 for 198 assessed questions. Two questions were removed from assessment. (R = right, U = unsupported, I = inexact; A = overall accuracy, MRR = Mean Reciprocal Rank).

QA system	R	U	I	A	MRR
InSicht only	62	4	0	32.28%	32.11
IRSAW + InSicht	65	4	1	33.68%	33.86

The monolingual German QA task in 2006 was more complex in comparison with tasks in previous years, and new types of questions were introduced, e.g. questions including temporal restrictions and list questions. Therefore, a comparison with results from previous years would not be adequate.

The number of correct answers found by IRSAW is expected to increase even more when the shallow method is improved, because the deep linguistic methods were not able to produce answers for some easy questions. However, systems that can be characterised to employ shallower techniques were able to find answers for the same set of topics. The concept of difficult and easy questions is difficult to define, because it depends on the methods employed by a QA system. For easy questions, answers typically are given explicitly in the text (word by word). Answering complex questions will involve paraphrasing and reasoning.

An example shall demonstrate that the overlap in answers produced by the shallow QA subsystem and InSicht is expected to be small, i.e., a substantial performance gain in the combination of deep and shallow processing is likely (see Table 3). For the QA@CLEF question set in 2006, InSicht did not find an answer to a seemingly easy question (topic 0173), but the pattern matching found one. A closer look at the answer snippet reveals that that concept *"satire"* is used metonymically, i.e., it actively participates in or causes an act of failing. This conflicts with the semantic information in the semantic computer lexicon the parser relies on, since a literature genre is not an animated object and therefore can not take the role of an agent. The pattern matching approach simply takes into account the keywords from the question and looks for numeric expressions near them.

In contrast, resolving temporal deictic expressions is beyond the capabilities of a pattern matching approach, but InSicht's inference rules reason at the level of meaning representation. For the first example question (topic 0079), InSicht resolves the temporal deixis *"25 years ago"* to 1970. Viewed from the perspective of the time the article was written (1995), the answer is correct.

The results already show that a combination of processing methods will further improve performance for the IRSAW system. The combination of both traditional IR with deep analysis techniques will provide a highly performing and a more robust system. However, the evaluation task QA@CLEF aims at evaluating a static corpus of newspaper articles. A test targeting web resources has not been performed, yet.

The statement given above is supported from a different view as well. The combination of InSicht with other shallower approach promises a performance boost, because results of the QA@CLEF task show that the performance of a system combining answers (an ideal system) would obtain more correct answers, i.e. there is little overlap between correct answers from systems with different approaches. Magnini et al. [Magnini *et al.*, 2006] give an estimate of a 22% performance increase in accuracy for an ideal system combining results of the monolingual German QA task.

5 Outlook

This paper presented work in progress on IRSAW, a question answering system relying on IR in its initial phases. The evaluation of a first prototype, merging answers from pattern matching based on IR with answers from InSicht was performed with the German question answering task at CLEF 2006 (QA@CLEF). Results for the shallower IR approach (only 17 answers for 200 questions were found) indicate that pattern extraction methods will have to be improved. However, the combination already demonstrates an increase in performance. For the monolingual German QA task at CLEF 2006, IRSAW achieved the second best result of eight monolingual German runs submitted for assessment. Due to increased question complexity, a comparison with results from previous years would not be appropriate.

Future work will include a separate evaluation of the three phases described: retrieval from web resources, retrieval from a local database to find answer candidates, and matching semantic networks to find precise answers.

Major limitations of the shallow approach are the size of the training corpus and the coarse-grained classification of questions and expected answer types. The training corpus will be further extended by large sets of facts extracted from resources available online. Question and answer types will in future be based on semantic properties of the concept representing the answer (i.e., the semantic sort of the concept and its semantic relation to the situation stated in the question). Furthermore, methods involving pattern matching on the surface level and on the syntactic level, used separately and as a combination will be investigated.

References

[Ahn *et al.*, 2006] David Ahn, Valentin Jijkoun, Karin Müller, Maarten de Rijke, Erik Tjong, and Kim Sang. The University of Amsterdam at QA@CLEF 2005. In Carol Peters, Fredric C. Gey, Julio Gonzalo, Gareth J. F. Jones, Michael Kluck, Bernardo Magnini, Henning Müller, and Maarten de Rijke, editors, *Accessing Multilingual Information Repositories: 6th Workshop of the Cross-Language Evaluation Forum, CLEF 2005, Vienna, Austria*, volume 4022 of *Lecture Notes in Computer Science*. Springer, Berlin, 2006.

[Grefenstette and Tapanainen, 1994] Gregory Grefenstette and Pasi Tapanainen. What is a word, what is a sentence? Problems of tokenization. In *3rd International Conference on Computational Lexicography*, pages 79–87, Budapest, 1994.

[Hartrumpf, 2003] Sven Hartrumpf. *Hybrid Disambiguation in Natural Language Analysis*. Der Andere Verlag, Osnabrück, Germany, 2003.

Table 3: Results for two example questions from the QA@CLEF task 2006 (topic 0079 and topic 0173). The first answer was obtained by InSicht, the second answer was produced by pattern matching.

Question	Document sentence	Answer
"In which year did Charles de Gaulle die?"	France's chief of state Jacques Chirac acknowledged the merits of general and statesman Charles de Gaulle, who died 25 years ago.	1970
"In welchem Jahr starb Charles de Gaulle?"	Frankreichs Staatschef Jacques Chirac hat die Verdienste des vor 25 Jahren gestorbenen Generals und Staatsmannes Charles de Gaulle gewürdigt. (SDA.951109.0236)	1970
"In which year was the Russian Revolution?"	The satire inspired by the Russian revolution 1917 lets the dream of liberty and equality fail because of humans.	1917
"In welchem Jahr fand die russische Revolution statt?"	Die von der Russischen Revolution 1917 inspirierte Satire läßt den Traum von Freiheit und Gleichheit an den Menschen scheitern. (FR940612-000533)	1917

[Hartrumpf, 2005] Sven Hartrumpf. Question answering using sentence parsing and semantic network matching. In Carol Peters, Paul Clough, Gareth J. F. Jones, Julio Gonzalo, Michael Kluck, and Bernardo Magnini, editors, *Multilingual Information Access for Text, Speech and Images: 5th Workshop of the Cross-Language Evaluation Forum, CLEF 2004*, volume 3491 of *Lecture Notes in Computer Science (LNCS)*, pages 512–521. Springer, Berlin, 2005.

[Hartrumpf, 2006a] Sven Hartrumpf. Adapting a semantic question answering system to the web. In *Proceedings of the EACL 2006 Workshop on Multilingual Question Answering (MLQA'06)*, pages 61–68, Trento, Italy, April 2006.

[Hartrumpf, 2006b] Sven Hartrumpf. Extending knowledge and deepening linguistic processing for the question answering system InSicht. In Carol Peters, Fredric C. Gey, Julio Gonzalo, Gareth J. F. Jones, Michael Kluck, Bernardo Magnini, Henning Müller, and Maarten de Rijke, editors, *Accessing Multilingual Information Repositories: 6th Workshop of the Cross-Language Evaluation Forum, CLEF 2005*, volume 4022 of *Lecture Notes in Computer Science*. Springer, Berlin, 2006.

[Helbig, 2006] Hermann Helbig. *Knowledge Representation and the Semantics of Natural Language*. Springer, Berlin, 2006.

[Hengel and Pfeifer, 2005] Christel Hengel and Barbara Pfeifer. Kooperation der Personennamendatei (PND) mit Wikipedia. *Dialog mit Bibliotheken*, Jg. 17, H.3:18–24, 2005.

[Leveling, 2006a] Johannes Leveling. *Formale Interpretation von Nutzeranfragen für natürlichsprachliche Interfaces zu Informationsangeboten im Internet*. Dissertation, Fachbereich Informatik, FernUniversität in Hagen, 2006. To appear in: Der andere Verlag, Tönning, Germany, 2006.

[Leveling, 2006b] Johannes Leveling. University of Hagen at CLEF 2006: Reranking documents for the domain-specific task. In Alessandro Nardi, Carol Peters, and José Luis Vicedo, editors, *Results of the CLEF 2006 Cross-Language System Evaluation Campaign, Working Notes for the CLEF 2006 Workshop*, Alicante, Spain, 2006.

[Magnini *et al.*, 2005] Bernardo Magnini, Alessandro Vallin, Christelle Ayache, Gregor Erbach, Anselmo Peñas, Maarten de Rijke, Paulo Rocha, Kiril Simov, and Richard Sutcliffe. Overview of the CLEF 2004 multilingual question answering track. In Carol Peters, Paul Clough, Gareth J. F. Jones, Julio Gonzalo, Michael Kluck, and Bernardo Magnini, editors, *Multilingual Information Access for Text, Speech and Images: 5th Workshop of the Cross-Language Evaluation Forum, CLEF 2004*, volume 3491 of *Lecture Notes in Computer Science (LNCS)*. Springer, Berlin, 2005.

[Magnini *et al.*, 2006] Bernardo Magnini, Danilo Giampiccolo, Pamela Forner, Christelle Ayache, Valentin Jijkoun, Petya Osenova, Anselmo Peñas, Paulo Rocha, Bogdan Sacaleanu, and Richard Sutcliffe. Overview of the CLEF 2006 multilingual question answering track. In *Results of the CLEF 2006 Cross-Language System Evaluation Campaign, Working Notes for the CLEF 2006 Workshop*, Alicante, Spain, 2006.

[Neumann and Sacaleanu, 2004] Günter Neumann and Bogdan Sacaleanu. A cross-language question/answering-system for German and English. In Carol Peters, Julio Gonzalo, Martin Braschler, and Michael Kluck, editors, *Comparative Evaluation of Multilingual Information Access Systems: 4th Workshop of the Cross-Language Evaluation Forum, CLEF 2003*, volume 3237 of *Lecture Notes in Computer Science (LNCS)*, pages 412–424. Springer, Berlin, 2004.

[Neumann and Xu, 2003] Günter Neumann and Feiyu Xu. Mining answers in German web pages. In *Proceedings of the International Conference on Web Intelligence (WI-2003)*, Halifax, Canada, October 2003.

[Ravichandran and Hovy, 2002] Deepak Ravichandran and Eduard Hovy. Learning surface text patterns for a question answering system. In *Proceedings of the 40th ACL conference*, Philadelphia, 2002.

Exploring the Potential of Semantic Relatedness in Information Retrieval

Christof Müller and **Iryna Gurevych**
Ubiquitous Knowledge Processing Group
Telecooperation, Darmstadt University of Technology
Hochschulstr. 10, 64289 Darmstadt, Germany
http://www.cre-elearning.tu-darmstadt.de/elearning/sir/

Abstract

Employing lexical-semantic knowledge in information retrieval (IR) is recognised as a promising way to go beyond bag-of-words approaches to IR. However, it has not yet become a standard component of IR systems due to many difficulties which arise when knowledge-based methods are applied in IR. In this paper, we explore the use of semantic relatedness in IR computed on the basis of GermaNet, a German wordnet [Kunze, 2004]. In particular, we present several experiments on the German IR benchmarks GIRT'2005 (training set) and GIRT'2004 (test set) aimed at investigating the potential of semantic relatedness in IR as opposed to bag-of-words models, as implemented e.g. in Lucene [Gospodnetic and Hatcher, 2005]. These experiments shed some light upon how to combine the strengths of both models in our future work. Our evaluation results show some improvement in IR performance over the bag-of-words model, i.e. a significant increase in mean average precision of about 5 percent points for the training set, but only 1 percent increase for our test set.

1 Introduction

It is often assumed that the use of linguistic, in particular lexical-semantic information, should improve the performance of bag-of-words IR systems, which are based on string matching. The problems with the bag-of-words IR systems arise due to polysemy and synonymy in the natural language. Polysemy of words creates ambiguity and can lead to poor precision due to the words, which are not sense disambiguated. If the synonymy is not taken into account, the recall of the system would be poor, as it does not find relevant documents containing terms, which are synonymous to the search term.

Multiple attempts have been made to address these issues by employing Natural Language Processing (NLP) methods in IR with so far limited success. In many cases, the use of semantic knowledge captured in computer-readable resources like WordNet [Fellbaum, 1998] or FrameNet [Baker *et al.*, 1998] has been explored for the task of disambiguating or selecting related words and thereby improving the performance of IR systems. There exist multiple ways to incorporate lexical-semantic knowledge into IR systems:

- *query expansion* where the query is extended by semantically related terms;
- *indexing concepts* instead of words;
- *document ranking functions* based on lexical-semantic knowledge.

In the paper, we describe a set of experiments aimed to integrate lexical-semantic knowledge into an IR system by using semantic relatedness as the model of relevance between query and documents. Section 2 describes the application domain and corpora used in our experiments. In Section 3, we introduce our baseline system and the newly developed semantic relatedness retrieval model including necessary preprocessing steps of documents and queries. Evaluation results follow in Section 4. After that, we put our work into context by extensively reviewing the state-of-the-art on integrating semantic knowledge in information retrieval in Section 5. Finally, we draw some conclusions in Section 6 and develop some ideas for further research.

2 Corpus Data

In this paper, we conduct experiments with the GIRT corpus, which is a domain-specific corpus devoted to the domain of social science and a standard information retrieval benchmark for German [Kluck, 2004]. It is used in the German domain-specific task at CLEF, which allows to make cross-system comparisons for this task. The corpus consists of abstracts of scientific papers in social science, together with the author and title information and several keywords. The experiments described in Section 4 use the topics and relevance assessments of CLEF'2005. A topic is a natural language statement of information need which is used to create a query for an IR system. For CLEF'2005 there are 25 topics for the GIRT corpus in the German language. Each topic consists of three different parts: a title (keywords), a description (a sentence), and a narration (exact type specification of documents to retrieve). A portion of GIRT documents is annotated with relevance judgements for each topic by using the *pooling technique*. Table 1 shows some statistics about the corpus and topics.

3 System Architecture

In this study, we compare two kinds of IR models on the GIRT corpus: an IR model as implemented by Lucene[1] as the baseline and a model integrating semantic relatedness.

3.1 Document and Query Preprocessing

During the preprocessing of documents and queries we apply several NLP methods which are commonly used in many state-of-the-art IR systems. These include tokenisation, stopword removing, stemming, lemmatisation and compound splitting.

[1] http://lucene.apache.org

	#docs	*#tokens*	*#distinct tokens*	*#tokens/doc (mean)*
GIRT4	151,319	14,312,116	525745	94.58
Topics CLEF2005:	25	–	–	–
Title	–	45	44	1.8
Narration	–	181	104	7.24
Description	–	517	281	20.68

Table 1: Corpus and query statistics (number of tokens counted after removing stopwords and lemmatising).

We perform both, stemming and lemmatisation, but use only lemmas for query and index building. In the future, we will compare the retrieval performance with lemmatised and stemmed indexes. There have already been a number of studies about the usefulness of morphological normalisation in IR. Some of the most recent ones are [Hollink *et al.*, 2004] and [Airio, 2006]. They confirm the positive impact which morphological normalisation has, especially for German. However, they find almost no difference in performance between stemming and lemmatisation. For our system, we use the Snowball Stemmer[2] and the lemmatiser of the TreeTagger [Schmid, 1994].

The third morphological normalisation we perform is the decomposition of compounds. The algorithm we use is based on [Langer, 1998] and uses GermaNet as the lexicon. Decompounding shows significant gain in performance for German in [Hollink *et al.*, 2004]. However, [Airio, 2006] can find almost no difference in performance.

3.2 Lucene-based IR

Lucene is an open source text search library based on an extended boolean (EB) model [Salton *et al.*, 1983]. The preprocessed topics are converted to a Boolean query, whereby separate terms are combined with the operator OR.

3.3 Semantic Relatedness

Semantic relatedness is defined as *any* kind of lexical-semantic or functional association that exists between two words [Gurevych, 2005b]. In order to compute semantic relatedness, lexical-semantic knowledge is required. This knowledge can be derived from a range of resources like computer-readable dictionaries, thesauri, or corpora. The experiments presented in this paper employ the German wordnet GermaNet as the knowledge base. Currently, GermaNet includes about 40000 synsets with more than 60000 word senses modelling nouns, verbs and adjectives. In previous work, the application of different semantic relatedness metrics to GermaNet has been explored [Gurevych and Niederlich, 2005]. The results suggested that the information content based metric introduced by [Lin, 1998] showed better performance than a dictionary-based metric by [Gurevych, 2005b]. Therefore, we integrated the metric by [Lin, 1998] in our information retrieval system. Sometimes, it is called a *universal* semantic similarity metric, as it is supposed to be application-, domain-, and resource independent. However, we should be aware of the fact that semantic similarity takes only synonymy and hyperonymy relations between two concepts into account. Our future work should extend this metric to other types of semantic relations. For computing the information content of concepts, the German newspaper corpus *taz*[3] was used. This corpus covers a wide variety of topics and has about 172 million tokens.

3.4 IR based on Semantic Relatedness

Computing semantic relatedness as described in Section 3.3 allows to quantify the relatedness between two semantic concepts. In order to apply the metric to the task of IR, the relevance of documents to a given query should be computed based on semantic relatedness for the concept pairs. Therefore, we first map all document and query terms except stopwords to concepts in the GermaNet structure receiving two sets of concepts K_d and K_q respectively. As a simple first approach we compute the similarities between a query and a document as the sum of the semantic relatedness values for each pair of query and document terms:

$$sim(d,q) = \sum_{i=1}^{n_d} \sum_{j=1}^{n_q} s(t_{d,i}, t_{q,j}) \quad (1)$$

4 Experimental Work

We conducted several experiments. After each experiment we performed a qualitative analysis of the results in order to understand the strengths and weaknesses of the proposed methods and derive improvements for our method.

Experiment 1 In the first experiment, we used the configuration explained above. Figure 1 depicts *mean average precision*[4] (MAP) for the different runs depending on the query length and the index type (lemmas or lemmas with compounds and decomposed parts) used.

The semantic relatedness (SR) model performs worse than the extended boolean (EB) model in all configurations. SR and EB system work best for short queries, longer queries seem to add noise and the better performance of a combination of *Title* and *Description* over *Description* suggests that some relevant search terms are missing in *Description* or their weighting is changed in the combination of *Title* and *Description*. The combination of lemmas, compounds and compound elements yields the best performance for the EB model, but for the SR model we can observe a decrease when using compound splitting. However follow-up experiments showed the superiority of decomposed compounds also for the SR model. We therefore give only the results for the runs using compound splitting and short queries (*Title*) for the follow-up experiments. Figure 2 shows MAP, the number of retrieved and relevant documents, and the precision after 10 documents have been retrieved (P10) for each experiment and the best EB run.

In order to identify weak points of the semantic relatedness method and to improve it, we examined the results of single topics and searched for possible errors in the relevance judgement of the system. The following shows an example for one topic.

Topic No. 131 (Title): *Zweisprachige Erziehung* (bilingual education) For this topic, the SR model performs better than the EB model. It ranks many relevant documents higher and can even retrieve some documents not found by Lucene. The documents which are not found by

[2] http://snowball.tartarus.org

[3] www.taz.de

[4] Mean average precision is the mean of the average precision for each query.

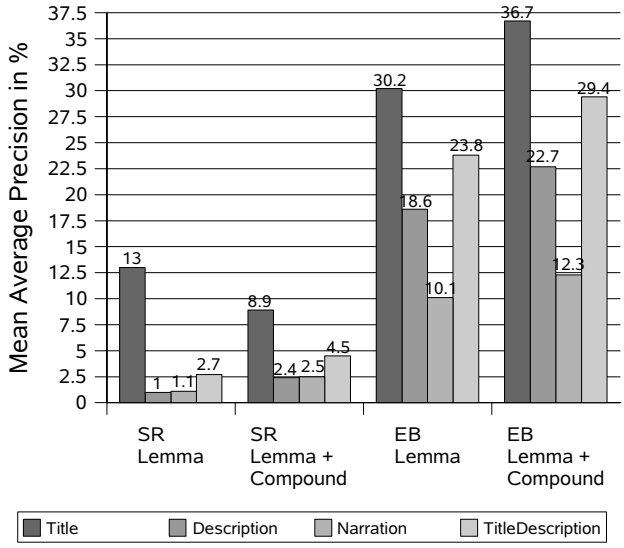


Figure 1: Mean Average Precision for experiment 1 (SR=semantic relatedness model, EB=extended boolean model).

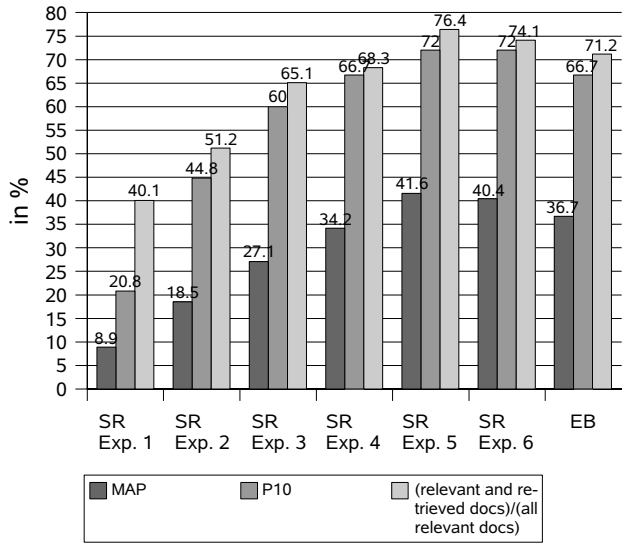


Figure 2: Results for different experiments using lemmas, compound splitting, and short queries (*Title*).

Lucene contain several different terms as a substitute for the query term *Zweisprachige*, which yield a high semantic relatedness score, e.g.:

- *Mehrsprachigkeit* (multilingualism) 0.98
- *sprachlich* (linguistic) 0.83
- *Vielsprachigkeit* (multilingualism) 0.86
- *bilingual* (bilingual) 1.0

Table 2 shows examples of relevant documents for this topic. The EB system retrieves the first two documents as they contain the query term *Erziehung* (education) several times, though they are both ranked low. The third document is not found by the EB system as it contains neither exactly *Zweisprachige* (bilingual) nor *Erziehung*. In this case, only the SR system is able to retrieve the relevant document by using lexical-semantic knowledge. One drawback of our system we observed was that many documents which relate only to one query term, e.g. *Erziehung*, but not to both query terms are ranked very high due to a high frequency of the occurring query term. This causes many relevant documents to be ranked much lower or not to be retrieved at all. To address this issue, we introduced a heuristic in a follow-up experiment.

Experiment 2 We extended the semantic relatedness model in the following way: for the documents which do not contain *all* of the query terms, i.e. not all of the query terms contribute a semantic relatedness score of > 0.8, the similarity score is multiplied by the factor $1/(1 + Number\_of\_not\_related\_query\_terms)$. The following shows the modified Equation 1:

$$sim(d, q) = \frac{\sum_{i=1}^{n_d} \sum_{j=1}^{n_q} s(t_{d,i}, t_{q,j})}{1 + n_{nr}} \qquad (2)$$

where n_{nr} is the number of the query terms which are not semantically related to any of the document terms. This heuristic pushes documents which contain the maximum number of query terms as opposed to those which contain a smaller number of query terms occurring many times.

For the SR system this heuristic yields a precision increase of 9.6 percent points and the number of relevant documents retrieved is increased by 297. This effect can also be seen in Table 2, where the first two documents are ranked much higher after applying this heuristic.

Experiment 3 Our analysis also suggested that the threshold of 0.8 for semantic relatedness might be set too low. We found several pairs of document and query terms with high scores which add noise to the retrieval system, e.g.:

- *Abfallwirtschaft* (waste industry) – *Nutzung* (use) 0.83
- *Werbung* (advertisement) – *Vorschlag* (suggestion) 0.81
- *Politik* (politics) – *Vorgehensweise* (approach) 0.89

We therefore experimented with different values for this threshold and found the optimal value to be 0.98, so that only highly related terms are taken into account for computing the relevance of documents. Increasing the threshold to 0.98 yields a performance improvement of 8.6 percent points and retrieves 372 relevant documents more than in the last experiment. As 0.98 is a very high threshold, we need to perform further analysis on threshold settings.

Experiment 4 In order to motivate this experiment, we will first discuss another example in our data.

Topic No. 147 (Title): *Fußball und Gesellschaft* (soccer and society) Many documents are ranked high which contain the exact query term *Gesellschaft* and the highly *Fußball*-related term Sport (0.84). However, these documents were neither annotated as relevant nor as irrelevant, and judging by title and abstract of some of the documents it can not be concluded if soccer is addressed in

Document ID and Title	*Relevant Terms*		*Relevance Judgement*	*Rank*						
	EB	*SR*		*EB*	*SR*					
					E1	*E2*	*E3*	*E4*	*E5*	*E6*
Ideologie und Realität : interkulturelle Erziehung auf Irrwegen GIRT-DE19980101311	*Erziehung* (education)	*Erziehung*, *Mehrsprachigkeit* (multilingualism), *bilingual*	relevant	117	**50**	**9**	5	5	20	16
Kurzinformation über Modellprojekte : schulische Betreuung der Kinder von Einwanderern und Interkulturelle Erziehung GIRT-DE19910106855	*Erziehung*	*Erziehung*, *Mehrsprachigkeit*, *zweisprachig* (bilingual)	relevant	450	**81**	**13**	8	8	21	23
Sozialpsychologische Grundlagen des schulischen Zweitspracherwerbs bei MigrantenschlerInnen... GIRT-DE19970112872	–	*zweisprachig*, *Mehrsprachigkeit*, *sprachlich* (linguistic)	relevant	—	**21**	**4**	9	9	3	7

Table 2: Topic No. 131 (Title): *Zweisprachige Erziehung* (bilingual education)

these documents or not. Despite that these documents seem to be more relevant than some documents highly ranked by Lucene containing only the query term *Gesellschaft*, they have a disturbing influence on the retrieval performance. Relevant documents which contain both query terms *Gesellschaft* und *Fußball*, but with a smaller frequency, receive a lower score and rank. In order to boost these documents to a higher rank, we 'punish' documents which do not contain the exact string representation of all the query terms. Similar to experiment 2 we therefore multiply the similarity score by the factor $1/(1 + Number\_of\_not\_string\_matched\_query\_terms)$.

$$sim(d,q) = \frac{\sum_{i=1}^{n_d}\sum_{j=1}^{n_q} s(t_{d,i}, t_{q,j})}{(1+n_{nsm})\cdot(1+n_{nr})} \quad (3)$$

This gives us a 7.1 percent points improvement in MAP and lets us retrieve 85 more relevant documents.

Experiment 5 Lucene is using the inverse document frequency *idf* which measures the general importance of a term for predicting the content of a document. We tried to integrate idf into our system and experimented with different measures and approaches and found that the following modification of Equation 3 brought the best improvement:

$$sim(d,q) = \frac{\sum_{i=1}^{n_d}\sum_{j=1}^{n_q} idf(t_{q,j}) \cdot s(t_{d,i}, t_{q,j})}{(1+n_{nsm})\cdot(1+n_{nr})} \quad (4)$$

with $idf(t) = 1/f_t$ where f_t is the number of documents in the collection containing term t. We yield an improvement in MAP of 7.4 percent points and retrieve 219 more relevant documents. With this result we outperform the EB system by 4.9 percent points and retrieve 141 relevant documents more than the best EB run.

Experiment 6 We assumed that by combining the SR approach with the EB model we would be able to improve the retrieval performance. Several methods for combining the similarity scores of different IR systems have been evaluated in the past. We adopted a very simple method which just calculates the sum of the scores of both systems:

$$sim(d,q) = sim_{EB}(d,q) + sim_{SR}(d,q) \quad (5)$$

In the evaluation of [Lee, 1997] this method performed not significantly worse than the best approach. As the semantic relatedness scores are not normalised, we normalise the scores by the minimum and maximum score for each query before applying Equation 5:

$$sim_{SR,norm}(d,q) = \frac{sim_{SR}(d,q) - sim_{SR,min}(q)}{sim_{SR,max}(q) - sim_{SR,min}(q)} \quad (6)$$

For combining the SR system of experiment 5 with the best run of the EB system we found no performance increase, but a slight decrease of MAP compared to experiment 5.

Test Set We used the topics of CLEF'2004 as test set and repeated experiment 5, experiment 6 and the best EB run. Table 3 shows the results and Table 4 shows a comparison of some runs of the SR and EB system on average precision (AP) and P10, using a paired T-Test. Despite the success on our training set we yield an insignificant performance increase of only 1.1 percent MAP for experiment 5 compared with the best EB run. In our future work, we plan to study the impact of semantic relatedness in IR, on multiple datasets to see, under which experimental conditions semantic relatedness is most appropriate.

Run	*MAP*	*P10*	*#Rel.+retr. docs*
SR Experiment 5	34.4	56.0	1074
SR Experiment 6	33.1	57.2	1044
EB	33.3	56.0	1088

Table 3: Results for the test set CLEF'2004.

Paired T-Test(p)		*AP*	*P10*
CLEF'2005	(SR Exp.5,EB)	**0.046**	0.103
	(SR Exp.6,EB)	**0.0040**	0.062
CLEF'2004	(SR Exp.5,EB)	0.755	1.0
	(SR Exp.6,EB)	0.937	0.798

Table 4: Paired T-Test (two-tailed distribution) between Exp.5/Exp.6 and baseline; statistically significant results are highlighted.

5 Related Work

There have been several attempts in the past to integrate lexical-semantic knowledge in IR systems. Table 5 gives an overview.

[Leveling, 2005] has used *Multilayered Extended Semantic Network* (MultiNet) representations of queries and documents in the CLEF domain-specific track for several years with mixed results.

[Smeaton, 1999] reports about several experiments on using WordNet in IR. A large-scale experiment yields a low retrieval performance due to malicious word sense disambiguation and unanalyzed proper nouns, but a small-scale follow-up experiment shows a significant improvement.

[Gurevych, 2005a] uses the German BERUFENet corpus, a collection of descriptions of 5800 professions in Germany [Bundesagentur für Arbeit, 2006], and investigates

Paper	*Queries*	*Documents*	*Method*	*Result*	*Explanation*
[Leveling, 2005]	CLEF2003/ CLEF2004 CLEF2005	GIRT3/ GIRT4/ GIRT4	query expansion/ indexing/ query construction	small improvement/ low performance/ inconclusive	knowledge not sufficient/ spelling and grammatical errors and sentence-based matching
[Smeaton, 1999]	Trec-3	portion of Trec-3, category B, Wall Street Journal	semantic similarity using WordNet	low performance	WSD errors, unanalysed proper nouns
	self-built user queries	captions for 4000 images	semantic similarity	encouraging performance	small scale, manual WSD
[Gurevych, 2005a]	essays about job preferences	BERUFENet elect. job counceling	query expansion/ semantic relatedness	performance increase/ no improvement	only for hyponomy/ no advanced pre-processing GermaNet coverage insufficient
[Arampatzis *et al.*, 2000]	—	—	semantic similarity	—	theoretical
[Fliedner, 2005]	—	Süddeutsche Zeitung, 1700 sentences	semantic similarity	encouraging	no extensive evaluation
[Sanderson, 1994]	subject code of documents	Reuters text categorisation collection	WSD influence on IR	insensitive to ambiguity, very sensitive to WSD errors	—
[Gonzalo *et al.*, 1998]	summaries of documents	derived from SEMCOR	WSD influence on IR	sensitive to ambiguity sensitive to WSD errors	—
[Gonzalo *et al.*, 1998]	summaries of documents	derived from SEMCOR	indexing WordNet synsets	high performance improvement	manual WSD
[Lytinen *et al.*, 2000]	153 test questions	600 frequently asked question files	semantic similarity	good performance	no exclusive evaluation of semantic similarity

Table 5: Summary of related work.

the use of query expansion and semantic relatedness using GermaNet as the underlying knowledge base. Query expansion yields a slightly increased performance. Incorrect analysis resulting from using stemming when mapping words to GermaNet entries and a missing word sense disambiguation (WSD) component are the main reasons for that. The retrieval model using semantic relatedness shows no significant performance gain over the baseline model. However, the system does not use any advanced preprocessing components, such as compound splitting and detection of negative preference statements referring to professions. It is stated that the coverage of the special terminology in GermaNet is still insufficient to be used as a knowledge resource in specialised domains.

A general linguistically motivated retrieval system is proposed by [Arampatzis *et al.*, 2000]. Among others, the model includes semantic expansion of queries and incorporates a semantic similarity measure into the retrieval function which performs a *fuzzy matching* of query and document terms. Unfortunately, no empirical evaluation of the model is reported.

[Fliedner, 2005] develops a question answering system, which incorporates linguistic knowledge from different resources, such as GermaNet and a German FrameNet currently under development in the SALSA project [Burchardt *et al.*, 2006]. The integration of the lexical-semantic knowledge is based on a *Generalised Similarity Measure*. However, no extensive evaluation of the question answering system is reported.

[Sanderson, 1994] takes a closer look at the relationship between word sense disambiguation and information retrieval. He introduces ambiguity into documents by using pseudo-words. The results show that: i.) word sense ambiguity is only a problem for very short queries; ii.) word sense disambiguation with an accuracy of less than 90% has a negative effect on the retrieval performance.[5]

[Gonzalo *et al.*, 1999] on the other hand show with their experiments that word sense disambiguation can be beneficial to IR, even with an accuracy of less than 90%. Additionally, indexing with WordNet synsets is examined by [Gonzalo *et al.*, 1998]. Information retrieval results improve on a manually disambiguated corpus, but also with a disambiguation accuracy of less than 90% an improvement is still observed.

[Lytinen *et al.*, 2000] show that word sense disambiguation of even around 60% accuracy can be helpful in IR. They use a WordNet-based semantic similarity metric for relevance ranking in a question answering system. The similarity metric is combined with a metric based on the *Vector Space Model*. Unfortunately, the impact of the similarity metric on the retrieval performance is not evaluated separately.

Summarising related work, we can see that there is no clear proof for the usefulness of lexical-semantic knowledge in information retrieval. One of the reasons for this is an insufficient coverage of terms by the knowledge bases. They often contain either general vocabulary and thus cannot be effectively applied in specific domains (the case of GermaNet), or model narrow domains and cannot be applied on a broad scale (hand-crafted ontologies). However, if the domain-specific vocabulary is modelled in a knowledge resource and the information retrieval is limited to this particular domain, successful results can be found. A way to overcome the insufficient coverage is by combining several knowledge resources in one system. Unrobust analysis and processing methods also have a negative influence on the performance of IR systems. Finally, word sense disambiguation seems to play an important role when incorporating the lexical-semantic knowledge in IR. Even if word sense disambiguation is not perfect, it seems to be possible to employ information retrieval methods which require word sense disambiguation and achieve positive results.

6 Conclusions and Future Work

In this paper, we explored the potential of lexical-semantic knowledge in IR by using semantic relatedness. Our experiments on the GIRT corpus show that semantic relatedness has the potential to outperform a traditional bag-of-words approach as implemented by Lucene.

Comparing the best run of our system (using *Title*) to IR systems which took part in the domain-specific German monolingual track of CLEF2005 (using *Title* and *Description*), our system would be ranked in the middle-field on the third rank.

The experiments presented in this paper were conducted

[5] The state-of-the art word sense disambiguation systems typically display between 65% and 70% accuracy rates, which is far below 90%.

on the GIRT corpus. In our future work, we would like to study how the models perform for different information retrieval scenarios. The Semantic Information Retrieval (SIR) Project investigates the application of NLP and IR techniques in the domain of electronic career guidance. We collected natural language essays about career preferences of school leavers. Based on these natural language essays, queries for the information retrieval system are generated which use the BERUFENet corpus. Pilot information retrieval experiments with the system based on semantic relatedness have been described by [Gurevych, 2005a].

Another interesting domain for the application of our information retrieval system is eLearning. Educational presentation slides usually contain phrases and keywords rather than complete sentences and feature a complex structure and layout, e.g. figures, tables or diagrams. These experiments can provide useful insights about the applicability of semantically enhanced information retrieval across different domains and different types of information retrieval scenarios.

Acknowledgements

We are grateful to the *Bundesagentur für Arbeit* for providing the BERUFENet corpus. We thank Torsten Zesch for his helpful comments and contributions, and we also thank Niels Ott for providing the compound splitter. This work is carried out as part of the "Semantic Information Retrieval" (SIR) project funded by the German Research Foundation.

References

[Airio, 2006] Eija Airio. Word normalization and decompounding in mono- and bilingual IR. *Information Retrieval*, 9(3):249 – 271, June 2006.

[Arampatzis *et al.*, 2000] Avi Arampatzis, Th.P. van der Weide, P. van Bommel, and C.H.A. Koster. Linguistically motivated information retrieval. *Encyclopedia of Library and Information Science*, 69, 2000.

[Baker *et al.*, 1998] Collin F. Baker, Charles J. Fillmore, and John B. Lowe. The Berkeley FrameNet project. In Christian Boitet and Pete Whitelock, editors, *Proceedings of ACL*, pages 86–90, San Francisco, California, 1998. Morgan Kaufmann Publishers.

[Bundesagentur für Arbeit, 2006] Bundesagentur für Arbeit. BERUFENet. http://infobub.arbeitsagentur.de/berufe/index.jsp, July 2006.

[Burchardt *et al.*, 2006] A. Burchardt, K. Erk, A. Frank, A. Kowalski, S. Pado, and M. Pinkal. The salsa corpus: a german corpus resource for lexical semantics. In *Proceedings of LREC 2006*, Genoa, Italy, 2006.

[Fellbaum, 1998] Christiane Fellbaum, editor. *WordNet: An Electronic Lexical Database*. MIT Press, Cambridge, MA, 1998.

[Fliedner, 2005] Gerhard Fliedner. A generalised similarity measure for question answering. In *Proceedings of NLDB 2005*, volume 3513 of *LNCS*, pages 380–383, Alicante, 2005.

[Gonzalo *et al.*, 1998] Julio Gonzalo, Felisa Verdejo, Irina Chugur, and Juan Cigarran. Indexing with wordnet synsets can improve text retrieval. In *Proceedings of the COLING/ACL '98 Workshop on Usage of WordNet for NLP*, pages 38–44, Montreal, Canada, 1998.

[Gonzalo *et al.*, 1999] J. Gonzalo, A. Pefias, and F. Verdejo. Lexical ambiguity and information retrieval revisited. In *Proceedings of EMNLP/VLC*, 1999.

[Gospodnetic and Hatcher, 2005] Otis Gospodnetic and Erik Hatcher. *Lucene in Action*. Manning Publications Co., 2005.

[Gurevych and Niederlich, 2005] Iryna Gurevych and Hendrik Niederlich. Computing semantic relatedness in german with revised information content metrics. In *Proceedings of "OntoLex 2005 - Ontologies and Lexical Resources" IJCNLP'05 Workshop*, Jeju Island, Republic of Korea, 2005.

[Gurevych, 2005a] Iryna Gurevych. Anwendungen des semantischen Wissens über Konzepte im Information Retrieval. In *Proceedings of Knowledge eXtended: Die Kooperation von Wissenschaftlern, Bibliothekaren und IT-Spezialisten*, Jülich, Germany, 2. - 4. November 2005.

[Gurevych, 2005b] Iryna Gurevych. Using the structure of a conceptual network in computing semantic relatedness. In *Proceedings of IJCNLP'05*, Jeju Island, Republic of Korea, 2005.

[Hollink *et al.*, 2004] Vera Hollink, Jaap Kamps, Christof Monz, and Maarten de Rijke. Monolingual document retrieval for european languages. *Information Retrieval*, 7(1 - 2):33 – 52, Jan 2004.

[Kluck, 2004] Michael Kluck. The girt data in the evaluation of clir systems from 1997 until 2003. In C. Peters, J. Gonzalo, M. Braschler, and M. Kluck, editors, *Comparative Evaluation of Multilingual Information Access Systems. CLEF 2003 Trondheim, Norway, Revised Selected Papers*, volume 3237 of *Lecture Notes in Computer Science*, pages 379–393. Springer, Trondheim, Norway, 2004.

[Kunze, 2004] Claudia Kunze. *Computerlinguistik und Sprachtechnologie. Eine Einführung*, chapter Lexikalisch-semantische Wortnetze, pages 423–431. Spektrum Akademsicher Verlag, second edition, 2004.

[Langer, 1998] Stefan Langer. Zur Morphologie und Semantik von Nominalkomposita. In *Proceedings of KONVENS*, page 8397, 1998.

[Lee, 1997] John Ho Lee. Analyses of multiple evidence combination. In *Proceedings of ACM-SIGIR*, pages 267–276, 1997.

[Leveling, 2005] Johannes Leveling. University of hagen at clef 2005: Towards a better baseline for nlp methods in domain-specific information retrieval. In *Results of CLEF 2005, Working Notes for the CLEF 2005 Workshop*, Wien, Österreich, 2005.

[Lin, 1998] Dekang Lin. An information-theoretic definition of similarity. In *Proceedings of the International Conference on Machine Learning*, pages 296–304. Morgan Kaufmann, San Francisco, CA, 1998.

[Lytinen *et al.*, 2000] S. Lytinen, N. Tomuro, and T. Repede. The use of wordnet sense tagging in faqfinder. In *Proceedings of the AAAI-2000 workshop on AI and Web Search*, Austin, TX, July 2000.

[Salton *et al.*, 1983] G. Salton, E. Fox, and H. Wu. Extended boolean information retrieval. *Communications of the ACM*, 26(11):1022–1036, 1983.

[Sanderson, 1994] Mark Sanderson. Word sense disambiguation and information retrieval. In *Proceedings of ACM-SIGIR*, pages 49–57, Dublin, IE, 1994.

[Schmid, 1994] H. Schmid. Probabilistic part-of-speech tagging using decision trees. In *Proceedings of International Conference on New Methods in Language Processing*, 1994.

[Smeaton, 1999] A.F. Smeaton. *Natural Language Information Retrieval*, chapter Using NLP or NLP Resources for Information Retrieval Tasks, pages 99–111. Kluwer Academic Publishers, 1999.

The Effects of Topic Familiarity on User Search Behavior in Question Answering Systems

Azzah Al-Maskari
Dept. of Information Studies
University of Sheffield
Sheffield, S1 4DP, UK
Lip05aaa@shef.ac.uk

Mark Sanderson
Dept. of Information Studies
University of Sheffield
Sheffield, S1 4DP, UK
m.sanderson@shef.ac.uk

Abstract

This paper reports on experiments that attempt to characterize the relationship between users and their knowledge of the search topic in a Question Answering (QA) system. It also investigates user search behavior with respect to the length of answers presented by a QA system. Two lengths of answers were compared; snippets (one to two sentences of text) and exact answers. A user test was conducted, 92 factoid questions were judged by 44 participants, to explore the participants' preferences, feelings and opinions about QA system tasks. The conclusions drawn from the results were that participants preferred and obtained higher accuracy in finding answers from the snippets set. However, accuracy varied according to users' topic familiarity; users were only substantially helped by the wider context of a snippet if they were already familiar with the topic of the question, without such familiarity, users were about as accurate at locating answers from the snippets as they were in exact set.

1 Introduction

Over the past few years, the question answering tracks at the Text Retrieval Conferences (TREC) Vorheese (2002, 2003, 2004) have brought formal and rigorous evaluation methodologies to bear on the question answering task. Although the importance of these evaluations cannot be denied, they measure performance of complex systems that typically involve a combination of information retrieval, information extraction, and natural language processing technologies. Järvelin and Ingwersen (2004) assert that the real issue in information retrieval systems design is not whether recall/precision goes up by a statistically significant percentage, but rather whether it helps the user solve the search task more effectively or efficiently. Therefore, the issue in Information Retrieval (IR) shifts from maximizing the retrieval performance by refining IR techniques and methods to maximizing the understanding of users' behaviors and information need representation during retrieval. Thus there is a great demand to consider the knowledge of users' behavior to be the key solution to successful retrieval.
In QA systems, the users' decision about the correctness and usefulness of an answer mainly depends on the context in which possible answers appear in addition to the users' previous knowledge of the topic of question.

This paper explores if there is a measurable difference in task performance using *exact* or *snippet* sets as an answer sets in a QA system; judges if there is a preference for exact over snippets retrieval or vice versa; and assesses if there is a difference in users' search behavior when they are searching exact or snippet sets (i.e., accuracy, level of confidence and number of answers to be displayed). This paper is divided as follows: in section 2 relevant previous works on answer-on-context is described, section 3 explains the experimental approach followed to conduct this user test and the results found, section 4 draws conclusions and future research.

2 Related Research

Previous studies have examined users' preferences for the length of answers from a QA system. Lin and Mitamura (2004) explored the roles of context in QA systems in four different options; exact answer, answer-in-sentence, answer-in-paragraph, and answer-in-document. They reported that users like answer-in-paragraph condition best and the exact answer condition the least. López-Ostenero et al. (2004) and Peinado et al. (2005) found that users prefer passage retrieval to document retrieval when searching standard document and passage retrieval engines in an interactive Cross-Language QA system. In contrast, the finding of Figuerola et al. (2004) demonstrated that users appraised document retrieval over passages for QA systems. A more recent study by Navarro et al. (2005) also showed that users considered larger context are better than shorter ones by comparing a passage retrieval system versus a clause retrieval system in an interactive cross language QA system.

Previous research has also examined user topic familiarity. Kelly and Cool (2002) reported on users searching an IR system, showing that as one's familiarity with a topic increases, searching efficiency increases and reading time decreases. In addition, Shiri and Revie (2003) investigated the effects of topic familiarity and topic complexity on cognitive and physical moves in a thesaurus-enhanced online IR system. They defined cognitive moves as user's conceptual analysis of terms or documents; and physical moves – those associated with the use of system features. Their findings indicated that an increased number of cognitive and physical search moves were associated with more complex topics. It was also observed that users searching moderately familiar and very familiar topics used more cognitive and physical moves than users searching for unfamiliar topics, though this difference was not statically significant. It was suggested that contextual factors, such as topic familiarity and task, affected the rate of occurrence of these behaviors. While is has been acknowledged in these studies that topic familiarity is an important factor influencing information seeking, no one appears to have considered QA systems and user topic familiarity. The work in this paper focuses on identifying information search behaviors that might be related to topic familiarity.

3 Experimental Approach

To start the experiments, 92 factoid questions were randomly selected from the TREC QA track and issued to an in-house QA system, AnswerFinder[1]. AnswerFinder was first tested by issuing 199 TREC questions; 92 factoid, 51 definition and 56 list. As shown in Table 1, AnswerFinder performed best at answering factoid questions, thus for this experiment only factoid questions were used.

	Factoid	Definition	List
Correct	63%	6%	15%
Not exact	1%	6%	9%
Wrong	22%	15%	13%
No answer	6%	23%	18%

Table 1: AnswerFinder Performance

The answers were manually assessed following an assessment scheme similar to the answer categories in iCLEF 2004:

- *Correct:* answer string is valid and supported by the snippets.
- *Non-exact:* answer string is missing some information, but the full answer is found in the snippets.
- *Wrong:* answer string and the snippets are missing important information or both the answer string and the snippets are wrong compared with the answer key.
- *No answer:* system returns no answer.

Table 1 illustrates AnswerFinder's effectiveness.

Answer Type		
Correct	65	68.5%
Non exact	1	1.1%
Wrong	19	6.5%
No Answer	7	23.9%
Total	**92**	**100.0%**

Table2: Overall View of AnswerFinder Performance.

[1] A web-based question answering tool Greenwood (2004). http://www.dcs.shef.ac.uk/~mark/phd/software/

Procedure and Design of the User's Tests

To conduct the user test 44 master students participated in judging the TREC questions. These users were of differing nationalities and backgrounds. Each participant was asked to judge AnswerFinder's answers to 20 different questions; 10 questions of snippets set and 10 of the exact set, each user was allotted 15 minutes to complete their task. Figure1[2] and Figure2[3] typify snippets and exact sets.

When did the royal wedding of Prince Andrew and Sarah take place?

1. 1986
   It was great fun when it did take place and you can have a look by clicking ... to the summer of **1986** and the nuptials of Prince Andrew and Sarah Ferguson.
2. July 23
   Finally, on November 20, 1947, the long-awaited Royal wedding took place . ... Prince Andrew and Sarah were married at Westminster Abbey on **July 23**, ...
3. 1992
   ... the royal marriage at Windsor's Guildhall, the ceremony will now take place on ... The Duchess of York, who was divorced from Prince Andrew in **1992**, ...
4. November 20, 1947
   Finally, on **November 20, 1947**, the long-awaited Royal wedding took place .
5. April 8th
   The wedding will take place on **April 8th** .

Figure 1: Answer Type: Snippet set

When did the royal wedding of Prince Andrew and Sarah take place

1. 1986
2. July 23
3. 1992
4. November 20, 1947
5. April 8th

Figure 2: Answer Type: Exact answer

A Latin-square matrix design was adopted to minimize the effects of user-specific, question specific and order-related factors on the tasks. To prevent any learning effect masking any system effect, every question was introduced to the participants in one set only, half of the participants were given the exact set first while the other half were given the snippets set first. Figure 3 shows a sample presentation order.

User	Search Order (condition: A|B, question 1.....100)
1	1 18 4 21 17 6 30 44 11 3 49 68 58 78 52 73 79 67 60 55
2	51 82 63 60 80 45 76 54 65 27 2 7 21 29 44 17 30 19 15 5
3	2 5 19 18 38 37 31 28 15 35 62 69 73 87 81 77 80 88 70 59
4	74 52 46 79 65 48 78 83 72 64 10 35 31 20 4 36 29 87 44 8
5	12 41 23 36 2 26 32 24 22 5 70 71 82 73 47 66 87 85 83 62
6	59 75 84 51 47 61 85 82 46 58 9 14 36 32 37 22 33 23 18 7
.	
.	
42	69 56 88 65 72 46 79 83 97 50 19 28 23 4 14 7 92 21 41 10
43	93 54 73 85 48 61 66 58 82 92 16 3 42 5 36 26 43 39 13 97
44	11 25 15 8 29 20 40 33 94 99 86 74 65 80 50 70 54 58 63 81

Figure 3: Questions Distribution: the shaded area means the snippets set and the white area means the exact set

The Test Measures

The following measures were used to examine users' search behavior and topic familiarity in QA system; these measures are explained in the succeeding sections:

- Participants' accuracy in identifying the correct answers in each set.
- The effect of question familiarity on participants.
- The effect of answer sets, snippets and exact, on participants.
- Confidence of participants in their judgments in each set.
- Number of answers preferred by the participants in QA systems.

The Accuracy of the Participants in Identifying the Correct Answer

Users' effectiveness was measured by the "correct answer" identified by each participant that was checked against an answer sheet (created by the evaluator). Table 3 illustrates the participants' overall accuracy in judging the exact and the snippets sets; accuracy was measured as the ratio of correct answers identified to the total number of questions. According to t-test (significant test), there is no significant difference between judging the snippets and the exact sets.

[2] This shows how AnswerFinder originally produces the answers

[3] This is the modified exact answer set taken from Figure 1.

	Correct	Wrong
Exact Set	53.3%	46.7%
Snippets Set	59.3%	40.7%

Table 3: Users' accuracy in identifying "correct answer"

An examination of failure cases was made: in some cases, the answers were clearly presented in the snippets but the participants did not choose the right answer this could be due to their understanding or lack of knowledge of the topic. Therefore, judgments for some questions have entailed human errors and there exists legitimate differences of opinions and perceptions among the participants. In addition, in some cases the QA system provided an incomplete or incoherent piece of text which did not help the users to deduce accurate answers. Thus, the low accuracies are due to both the system and the participants.

The Effect of Questions' Familiarity on the Performance of Participants

Participants completed a post search questionnaire, which determined their familiarity with each question on three scales:

1) Personal knowledge: if users know the answer to the question before looking at the given choices.
2) No idea: if they have no idea about the answer.
3) Topic is familiar but they don't remember the answer.

Tables 4, 5 and 6 depict how the participants' knowledge affected their choices of answers. The general results showed that accuracy varied with respect to topic familiarity; when users were more familiar with a topic their accuracy in identifying answers was high. Perhaps unsurprisingly when users knew the answer, their accuracy in identifying it correctly was similar in the exact and snippet. Table 5 shows users were better able to locate correct answers in the snippets if they had previous knowledge of the topic. Table 6 illustrates similarity in users' accuracy in both sets when users lacked knowledge about the topics. This was surprising as it was presumed such a question was one where users needed the most help.

Set	Correct	Wrong
Exact	75.2%	24.8%
Snippet	73.6%	26.4%

Table 4: User knows answer

Set	Correct	Wrong
Exact	53.3%	46.7%
Snippet	62.7%	37.3%

Table 5: User is familiar with topic

Set	Correct	Wrong
Exact	44.9%	55.1%
Snippet	47.7%	52.3%

Table 6: User lack the familiarity and knowledge

The Effect of the Snippets Set in Identifying the Correct Answers

Participants completed a post search questionnaire, which assessed if the snippets had helped them in identifying the correct answer. More than half of them agreed that the snippets frequently assisted them in identifying the correct answers. However 22.2% asserted it did not help and 20% thought that it only helped occasionally.

Participants were also asked about their preference of the exact and the snippets sets; 24.4% of them favor the exact set while 75.6% prefer the snippets set. The participants justified the reason for their choices; the group who chose the snippets explained it gave more detail and insight about the answers, which resulted in users feeling more confident. On the other hand, the group who chose the exact said the snippets did not help them to pick an answer, but it rather confused them for the same reason identified earlier. They further confirm their contentment with the exact answers for factoid questions because factoid questions do not require explanation. To summarize, the snippets set helped participants to answer the questions in many cases albeit in some cases it failed.

Apparently this shows different preferences among participants as Schamber (1994) states that for text retrieval, different people have different opinions about whether or not a given document should be retrieved for a query.

Confidence of the Participants on their Judgments

Participants' confidence was considered an important facet of their assessment; different judgments indicated different confidence. Participants were required to rate their confidence after judging each question. The users' confidence ranged from 1 (not at all confident) to 5 (completely confident). Table 7 shows users' confidence with snippets was higher than with exact. The majority of participants complained about the difficulty of choosing an answer from the exact set because they lacked the knowledge about the questions' topics.

Confidence Level	Snippets	Exact
completely confident	28.4%	15.8%
Very confident	15.6%	8.4%
fairly confident	15.1%	16.4%
Slightly confident	14.4%	20.0%
Not Confident	26.4%	39.3%

Table 7: participants' confidence in both sets

Number of Answers Preferred by the Participants

In the final test, participants were asked about the number of answers they deemed suitable for QA systems. According to Table 8, five answers are reasonably enough; fewer answers are also acceptable by some users.

No. of Answers	
1	6.7%
2	6.7%
3	20%
4	20%
5	35.6%
more than 5	8.9%
10	2.2%

Table 8: No. of answers preferred by the participants

Participants who chose one or two answers claimed that for factoid questions only one or two answers were sufficient and more answers may lead to confusion. The users who preferred more than five answers wanted a chance to compare and verify the results especially when they did not know the topic. Thus, the less knowledgeable they are about the topic, the more answers to be displayed.

4 Conclusion

A user test was conducted to investigate a QA system. It was concluded that participants preferred and obtained higher accuracy in finding answers from the snippets than from exact, although there was no statistically significant difference in accuracy between the two. The general trend suggests that the longer context, the better the users' accuracy. Some information search behaviors were also discussed such as accuracy and confidence which vary with respect to topic familiarity. Accuracy was found to increase with topic familiarity; the more familiar participants were with a topic, the more accurate their answers. Participants' familiarity with the topic was boosted by the context provided by the snippets. Nevertheless, with no previous familiarity of the topic, users' ability to locate correct answers was similarly poor for both exact and snippet sets.

QA systems are built to fulfill the goal of fact-finding by providing users with short answers quickly and concisely. However, according to this study and the previous studies, users are not always satisfied with short answers and they often prefer longer context to verify the accuracy of the answers. Thus, it can be concluded that it is difficult to establish a fixed context for QA systems.

For future research, it is recommended to build an interface with an answer space that enables users to navigate information clusters i.e., to view the exact answer, the passage or the full document. This is to assess how far each user searches in the information clusters to detect how much context users really want in inclusive of their knowledge level.

A further study could combine information context with cultural and physical contexts by taking a broader viewpoint, not only the analysis and evaluation of the system performance, but also include contextual and situational factors such as users and their knowledge levels, information needs; work-tasks, characteristics and types.

References

FIGUEROLA, C. G., ANGEL F. ZAZO, BERROCAL, J. L. A. & ALDANA, E. R. V. D. (2004) REINA at iCLEF 2004. *iCLEF*.

JÄRVELIN, K. & INGWERSEN, P. (2004) Information seeking research needs extension towards tasks and technology. *Information Research,* 10(1).

KELLY, D. & COOL, C. (2002) The Effects of Topic Familiarity on Information Search Behavior. *Joint Conference on Digital Libraries (JCDL)***,** 74-77.

LIN, F. & MITAMURA, T. (2004) Keyword Translation from English to Chinese for Multilingual QA. *Association for Machine Translation in the Americas, AMTA* Washington, USA.

LÓPEZ-OSTENERO, F., GONZALO, J., PEINADO, V. & VERDEJO, F. (2004) Interactive Cross-Language Question Answering: Searching Passages versus Searching Documents. *iCLEF*.

NAVARRO, B., MORENO-MONTEAGUDO, L., NOGUERA, E., VÁZQUEZ, S., LLOPIS, F. & MONTOYO, A. (2005) "How much context do you need?" An Experiment about Context Size in Interactive Cross-language Question Answering. . *iCLEF*.

PEINADO, V., LÓPEZ-OSTENERO, F., GONZALO, J. & VERDEJO, F. (2005) UNED at iCLEF 2005: Automatic Highlighting of Potential Answers. *iCLEF*.

SCHAMBER, L. (1994) Relevance and information behavior. *ARIST***,** 3-48.

SHIRI, A. A. & REVIE, C. (2003) The effects of topic complexity and familiarity on cognitive and physical moves in a thesaurus-enhanced search environment. *Journal of Information Science,* 29**,** 517.

Dedicated Backing-Off Distributions for Language Model Based Passage Retrieval

Munawar Hussain, Andreas Merkel and Dietrich Klakow
Spoken Language Systems
Saarland University, Saarbrücken, 66123, Germany
{Munawar.Hussain|Andreas.Merkel|Dietrich.Klakow}@lsv.uni-saarland.de

Abstract

Passage retrieval is an essential part of question answering systems. In this paper we use statistical language models to perform this task. Previous work has shown that language modeling techniques provide better results for both, document and passage retrieval.

The motivation behind this paper is to define new smoothing methods for passage retrieval in question answering systems. The long term objective is to improve the quality of question answering systems to isolate the correct answer by choosing and evaluating the appropriate section of a document.

In this work we use a three step approach. The first two steps are standard document and passage retrieval using the Lemur toolkit. As a novel contribution we propose as the third step a re-ranking using dedicated backing-off distributions. In particular backing-off from the passage-based language model to a language model trained on the document from which the passage is taken shows a significant improvement.

For a TREC question answering task we can increase the mean average precision from 0.127 to 0.176.

1 Introduction

Recently lot of work has been carried out on open-domain Question Answering Systems. These QA Systems include an initial document and/or passage retrieval step. Retrieved passages are then further processed using a variety of techniques to extract the final answers. The passage retrieval method strongly influences the performance of QA System. This is especially true for real systems where computational resources are limited. A good passage retrieval system will mean that only small number of top ranked passages needs to be analyzed to find the answer. In this paper[1] we compare the existing retrieval methods, both traditional and language modeling based, for document and passage retrieval. We have used the AQUAINT document collection as training and test corpus. Out of all methods tested, by choosing the best passage retrieval method as our baseline, we define and test new language models to improve retrieval performance. These language models are defined on different data collections (passage collection, document collection, corpus) and are interpolation based unigram language models.

The rest of this paper is organized as follows. Related work is discussed in Section 2. Section 3 discusses document retrieval. Section 4 presents the passaging of documents and passage retrieval performed. Section 5 explains the process of re-ranking. We conclude the paper by discussing our results and future work in Section 6.

2 Related Work

This section discusses the state of the art in the field of passage retrieval.

Passage retrieval is an important component of QA Systems and it directly influences overall performance.

C. L. A. Clarke et. al. [Clarke *et al.*, 2000] developed the MultiText system, which implements a technique to efficiently locate high-scoring passages.

A language modeling based approach was used by Andres Corrada-Emmanuel et. al. [Corrada-Emmanuel *et al.*, 2003]. They examined the effectiveness of language models in passage ranking for a question answering system.

Dell Zhang et. al. [Zhang and Lee, 2003] also developed a language modeling approach to passage question answering. Their system consists of a question classification component and a passage retrieval component.

Stefanie Tellex et. al. [Tellex *et al.*, 2003] carried out a Quantitative Evaluation of Passage Retrieval Algorithms for Question Answering. They evaluated a number of passage retrieval algorithms, and one new algorithm of their own called Voting. They implemented a voting scheme that scored each passage based on its initial rank and also based on the number of answers the other algorithms returned from the same document.

Some work has been done to improve the document retrieval by performing passage retrieval.

James P. Callan [Callan, 1994] examined passage level evidence in document retrieval.

Use of language modeling for passage retrieval and comparison with document-based retrieval was done by Xiaoyong Liu et. al. [Liu and Croft, 2002].

Deng Cai et. al. [Cai1 *et al.*, 2004] explored the use of page segmentation algorithms to partition web pages into blocks and investigated how to take advantage of block-level evidence to improve retrieval performance in the web context.

3 Document Retrieval

This section explains our experimental setup for document retrieval. The retrieved document set will be later used for passage retrieval.

[1]This work was partially supported by the BMBF project Smartweb under contract 01 IMD01 M

3.1 Dataset

The training document set or corpus for evaluation is the AQUAINT collection that consists of 1,033,461 documents taken from the New York Times, the Associated Press, and the Xinhua News Agency newswires. We selected AQUAINT as some well established standard task, which is helpful to compare our work with the state of the art. Our question set for evaluation contains 50 factoid questions, from TREC topic 1394 to 1443. In all our experiments, stemming is applied. No stop word removal is performed. Relevance judgments are obtained from the judged pool of top retrieved documents by various TREC participating retrieval systems.

3.2 System Architecture for Document Retrieval

The inputs to the system are the corpus and a set of questions. The output is a ranked list of documents for each question. Bellow is an explanation of each of the system components.

KeyFileIndexer & Stemmer: This component builds a key file index of AQUAINT corpus. Stemming is done along with indexing, using the Krovetz stemmer. The generated index is used by each retrieval method.

Question Stemmer: It is responsible for converting questions into queries by stemming them. Again the Krovetz stemmer is used for stemming.

Retriever: This component is responsible for the actual retrieval of documents. Retrieval methods are explained in following section.

3.3 Experimental Methods

A number of popular retrieval techniques exist, which include both traditional and language modeling techniques. We evaluate the performance of some of these techniques on our test data. The retrieval methods evaluated in this section are standard TFIDF, OKAPI, and the language modeling framework. The Dirichlet Prior, Jelinek-Mercer, and Absolute Discounting smoothing methods are the three methods that we have tested. They belong, in general sense, to the category of interpolation-based methods, in which we discount the counts of the seen words and the extra counts are shared by both the seen words and unseen words. The Lemur toolkit is used to run the experiments, because it is efficient and is optimized for fast retrieval. It provides both traditional and language modeling based retrieval algorithms and has been used by many research groups in the IR community.

3.4 Evaluation Methodology

Our goal is to study the behavior of individual retrieval methods and smoothing techniques as well as to compare different methods. Unlike traditional retrieval techniques, in case of language modeling retrieval technique, for each smoothing method we experiment with a wide range of parameter values. In each run, the smoothing parameter is set to the same value across all queries and documents. (While it is certainly possible to set the parameters differently for individual queries and documents through some kind of training procedure, it is beyond the scope of our work.) In order to study the behavior of a particular smoothing method, we examine the sensitivity of non-interpolated average precision to variations in a set of selected parameter values. Along with finding the optimal value of smoothing parameters, we also need to find the optimal number of retrieved documents N. Therefore we first fix the number of retrieved documents by comparing the non-interpolated average precision for varying number of documents retrieved, using each retrieval method. For the purpose of comparing smoothing methods, we first optimize the performance of each method using the non-interpolated average precision as the optimization criterion, and then compare the best runs from each method. The optimal parameter is determined by searching over the entire parameter space.

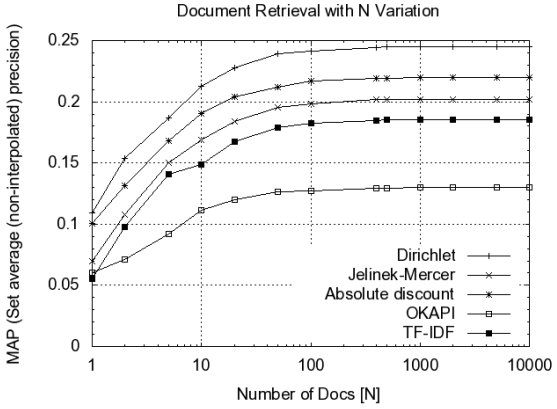


Figure 1: Document Retrieval with varying number of documents retrieved. For Dirichlet Prior the value of prior is set to 2000, for Jelinek-Mercer the value of λ is set to 0.8 and for Absolute Discounting the value of δ is set also to 0.8.

3.5 Experimental Results

This section explains results obtained from different retrieval methods. We first derive the expected influence of number of documents retrieved by plotting the non-interpolated average precision against document number for each retrieval method. We examine the sensitivity of retrieval performance by plotting the non-interpolated average precision at N documents against different values of the smoothing parameter. Following section explains the reason for retrieving a finite number of N documents per query.

Document size tuning

In this section, we study the behavior of each retrieval technique for different numbers of documents retrieved. We examine the sensitivity of retrieval performance by plotting the non-interpolated average precision, with fixed smoothing parameter for this experiment where required, against different number of documents retrieved. The smoothing parameter values are taken from previous work [Zhai and Lafferty, 2001]. The plot in Fig 1 displays the non-interpolated average precision for different number of documents retrieved. It can be seen that with increase in document number, performance also increases. It can also be seen that the increase in performance after 500 documents is relatively marginal. For number of retrieved documents N greater than 500 the cost of computing is significantly larger compared to the gain in performance. Therefore N is fixed at 500. Overall the Dirichlet Prior performed best by far. One reason for this could be that our queries on average are not verbose. Our experiments support the claim that language modeling techniques perform better than traditional ones, as TF-IDF and OKAPI performed worse.

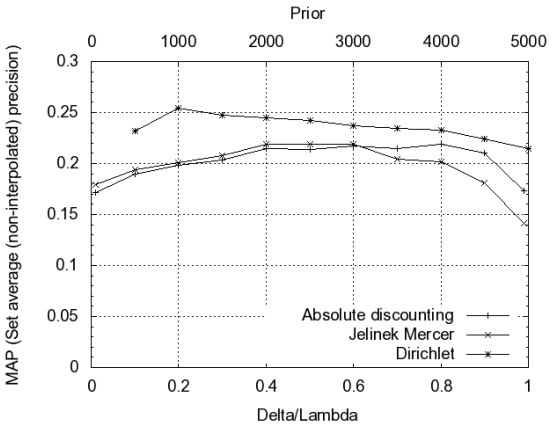


Figure 2: Plot of non-interpolated average precision against smoothing parameter, with delta/lambda varying from 0.01 to 0.99 and prior varying from 500 to 5000. Number of retrieved documents fixed at 500.

Parameter tuning for language modeling techniques
In this section, we examine the sensitivity of retrieval performance by plotting the non-interpolated average precision at 500 retrieved documents against different values of the smoothing parameter. Following is the analysis of our results.

For Jelinek-Mercer, the value to λ is varied between zero and one. The plot in Fig 2 shows non-interpolated average precision for different settings of λ. As depicted in plot, optimal value of λ is near 0.5, which indicates that our queries are of mixed length. According to [Zhai and Lafferty, 2001], for short queries optimal point is around 0.1 and for long queries optimal point is generally around 0.7. This is because long queries need more smoothing and less emphasis is placed on the relative weighting of terms.

Dirichlet Prior: For Dirichlet Prior, the value of prior μ is varied between 500 and 5000 with intervals of 500. The plot in Fig 2 illustrates the non-interpolated average precision for different settings of the prior sample size. As mentioned in [Zhai and Lafferty, 2001], the optimal prior μ vary from collection to collection and depends on query lengths. For our dataset and questions it is around 1000.

Absolute Discounting: For Absolute Discounting, the value to δ is varied between zero and one. The plot in Fig 2 shows non-interpolated average precision for different settings of δ. The optimal value of δ is near 0.8, which fortifies the claim by [Zhai and Lafferty, 2001] that the optimal value for δ tends to be around 0.7.

Overall the Dirichlet Prior performed best using prior of 1000 and 500 retrieved documents. Then came Absolute Discounting, which is better than Jelinek-Mercer. The good performance of Dirichlet Prior is relatively insensitive to the choice of μ. Indeed, many non-optimal Dirichlet runs are also significantly better than the optimal runs for Jelinek-Mercer and Absolute Discounting. This is because our queries are not long. As for long queries, Jelinek-Mercer is supposed to perform the best. As displayed by Table 1, TF-IDF performed slightly worse than Jelinek-Mercer, while OKAPI performed even worse.

4 First Pass Passage Retrieval

Passage retrieval is mainly used for three purposes. Firstly, passage retrieval techniques have been extensively used in standard IR settings, and have proven effective for document retrieval when documents are long or when there are topic changes within a document. Secondly, from an IR system user's standpoint, it may be more desirable that the relevant section of a document is presented to the user than the entire document. Thirdly, passage retrieval is an integral part of many question answering systems. We are performing passage retrieval for question answering systems. This section explains our methodology to establish a baseline using existing techniques developed for passage retrieval. For our experiments, we first retrieve documents (Section 3), then split these documents into passages.

Method	Parameter	MAP
Dirichlet Prior	$\mu = 1000$	0.254
Jelinek-Mercer	$\lambda = 0.5$	0.219
Absolute Discounting	$\delta = 0.8$	0.219
TFIDF	-	0.185
OKAPI	-	0.130

Table 1: Non-interpolated average precision for best run of each retrieval methods. With μ of 1000, δ of 0.8, and λ of 0.5

4.1 Experimental Setup

This section explains our setup for passage retrieval.

Passage Making
Passages are created using the following procedure. The top 500 retrieved documents are selected, see Section 3 for details of the document retrieval. The selected documents are then split into passages by a "passage maker". Our passage making technique is based on document structure [Berger and Lafferty, 1999] [Agichtein and Gravano, 2000] [Clarke*et al.*, 2000]. This entails using author-provided marking (e.g. period, indentation, empty line, etc.) as passage boundaries. Examples of such passages include paragraphs, sections, or sentences. Since our corpus is nicely structured (SGML form), we used paragraphs as passages.

Dataset
The query topics are the same as used for document retrieval (Section 3). For each query we have a distinct corpus consisting of passages created from the top 500 retrieved documents.

Experimental Methods
For passage retrieval we used the same set of retrieval methods as for document retrieval explained in Section 3. Likewise, the evaluation methodology is the same as for document retrieval (Section 3).

4.2 Experimental Results

Following subsections discuss results.

Passage document size tuning
In this section, we study the behavior of each retrieval technique for different number of retrieved passages, which is similar to what we did for document retrieval in Section 3. The plot in Fig 3 shows the non-interpolated average precision for different number of retrieved passages. It can be seen that with increase in number of retrieved passage documents, performance also increases. But the increase in performance after 500 passages is relatively marginal. Therefore the passage document number N is fixed at 500. Overall Dirichlet Prior performed best. Our experiments also show that there is no significant performance difference between retrieval methods, i.e. the curves are pretty

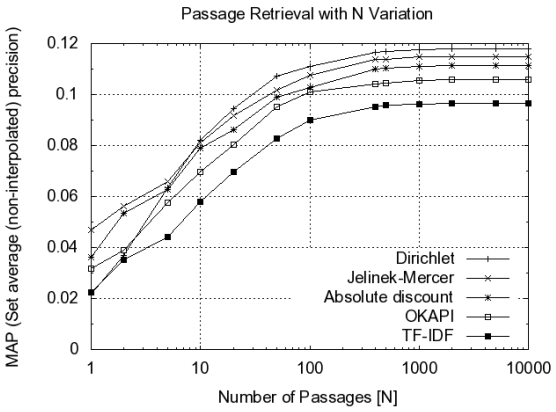


Figure 3: Passage Retrieval with varying number of retrieved passages. For Dirichlet Prior the value of prior is set to 1000, for Jelinek-Mercer the value of λ is set to 0.4 and for Absolute Discounting the value of δ is set also to 0.4.

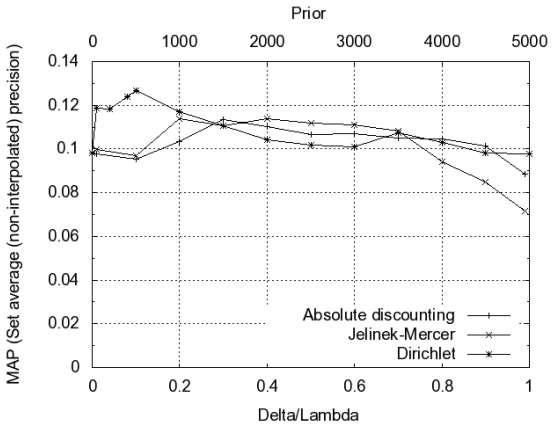


Figure 4: Plot of non-interpolated average precision against smoothing parameter,, with delta/lambda varying from 0.01 to 0.99 and prior varying from 500 to 5000. Number of retrieved passages fixed at 500.

close to each other. Performance of OKAPI is slightly worse than language modeling techniques. TF-IDF showed worse performance. Another noticeable fact is that Dirichlet Prior performance improves significantly for N between 1 and 10.

Parameter tuning for language modeling techniques
In this section, we study the behavior of individual smoothing methods, as we did for document retrieval in Section 3. Below is an analysis of our results.

Jelinek-Mercer smoothing: For Jelinek-Mercer, the value of λ is varied between zero and one. The plot in Fig 4 shows non-interpolated average precision for different settings of λ. As depicted in plot, optimal value of λ is near 0.4, which indicates that our queries are of mixed length. According to [Zhai and Lafferty, 2001], for short queries optimal point is around 0.1 and for long queries optimal point is generally around 0.7. As long queries need more smoothing and less emphasis is placed on the relative weighting of terms.

Dirichlet Prior: The value of Dirichlet Prior μ is varied between 1 and 5000 with intervals of 500. The plot in Fig 4 illustrates the non-interpolated average precision for different settings of the prior sample size. As mentioned in [Zhai and Lafferty, 2001], the optimal priors μ vary from collection to collection and depends on query lengths. For our dataset and questions it is around 500.

Absolute Discounting: The value of δ is varied between zero and one. The plot in Fig 4 shows the non-interpolated average precision for different settings of δ. The optimal value of δ is near 0.3.

Overall the Dirichlet Prior performed best using μ of 500 and 500 retrieved passage documents. Then came Jelinek-Mercer, which is slightly better than Absolute Discounting. But the performance difference is not very significant. OKAPI performed slightly worse than Absolute Discounting while TF-IDF performed even worse.

Table 2 gives a comparison of the best run by each technique.

Method	Parameter	MAP
Dirichlet Prior	$\mu = 500$	0.127
Jelinek-Mercer	$\lambda = 0.4$	0.114
Absolute Discounting	$\delta = 0.3$	0.113
TFIDF	-	0.105
OKAPI	-	0.096

Table 2: Non-interpolated average precision for best run of each retrieval methods

5 Passage Re-Ranking

This section explains our language models, which are based on an interpolation smoothing scheme. Since Lemur is not flexible enough to implement such custom models, we shifted to our own language modeling toolkit. This toolkit is very flexible in generating custom language models. It uses perplexity to rank the documents. To check the similarity between the two toolkits, an experiment was carried out using Jelinek-Mercer smoothing technique to regenerate the results produced by Lemur. These results confirmed the validity of results generated by our toolkit.

5.1 Experimental Setup

Our experimental setup consists of document collections generated by experiments explained in previous sections. Following sections explain our datasets, experimental methods, and the system architecture.

Dataset
The query topics are the same as used for document retrieval (Section 3). The corpus C for evaluation is the AQUAINT collection that consists of documents taken from the New York Times, the Associated Press, and the Xinhua News Agency newswires. Also, we have the document collection dc and the passage collection pc obtained from our previous experiments. All these collections are stemmed and no stop word removal is performed.

Evaluation Methodology
Our toolkit uses perplexity to rank the documents. For the purpose of studying the behavior of an individual language model, we select a set of representative parameter values and examine the sensitivity of non-interpolated average precision MAP to the variation in these values. In question answering mean reciprocal rank (MRR) is also widely used. We checked the correlation of MRR and MAP on question answering tasks. For consistency with the document retrieval, we report MAP throughout the paper.

Experimental Methods
Our experimental methods are language modeling based. We have defined a number of language models using Jelinek-Mercer smoothing techniques.

5.2 System Architecture for Passage Re-Ranking

This section explains complete architecture of our experimental setup. Language models explained in this section utilize the vocabulary closed with the query and the value of interpolation parameter is varied between zero and one. The main difference between these models is the background collection.

Language Model I (pdclm)
This language model is defined as linear interpolation between unigram language models defined on passage and related document collection. Where each passage is taken from related retrieved passages (Section 4) and related document collection consists of 500 top ranked documents retrieved (Section 3), for a given query. As perplexity is given by the formula

$$PP = exp[-\sum_{w} f(w) \log P(w)]$$

where *f(w)* is the relative frequency of words in the query and the probability is

$$P(w) = (1 - \lambda)P_{ml}(w|p) + \lambda P(w|dc),$$

where P_{ml} is maximum likelihood of word *w* in passage *p* and *dc* is document collection.

Language Model II (ppclm)
This language model is similar to pdclm explained above with the related passage collection consisting of 500 top ranked passages retrieved as the background collection. For this language model the probability is

$$P(w) = (1 - \lambda)P_{ml}(w|p) + \lambda P(w|pc),$$

where *pc* is passage collection.

Language Model III (pdlm)
Here again the language model differs from pdclm in the background collection. The background collection is the single document from witch the passage was extracted i.e. the document containing the passage being ranked. For this language model, the probability for calculating the perplexity is

$$P(w) = (1 - \lambda)P_{ml}(w|p) + \lambda P(w|d),$$

where *d* is single document. Fig 5 explains complete setup to re-rank passages using this language model.

Re-ranker: It is responsible to re-rank the collection of 500 related passages per query. It utilizes standard tree and background tree containing statistical information required by language models.

Vocabulary: Our word list consists of all the words in the single document containing the passage being ranked, closed with words from query.

Standard Tree: It contains statistical information for given passage being ranked. We build one standard tree per passage.

Background Standard Tree: It consists of statistical information for the single document containing the passage being ranked. We build one standard tree per document.

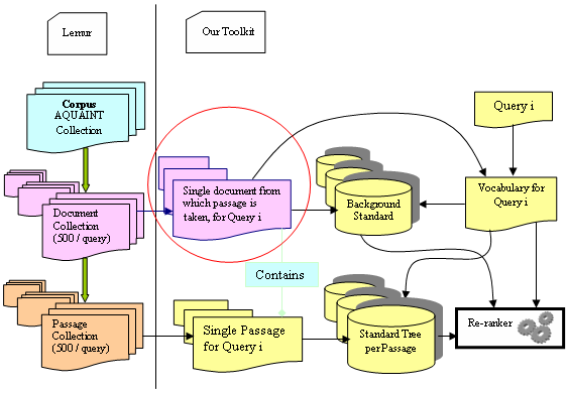


Figure 5: Dataset flow diagram for the pdlm language model.

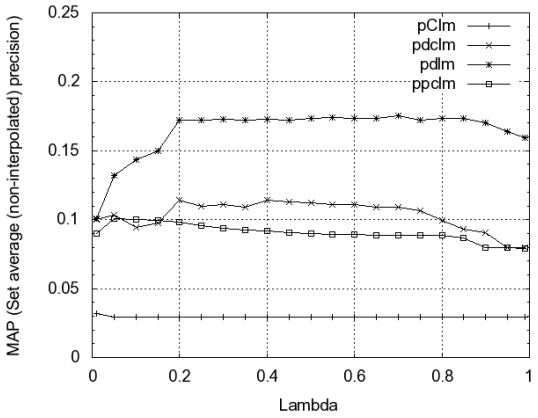


Figure 6: Plot of non-interpolated average precision against λ. Jelinek-Mercer b/w passage and different background collections with λ varying from 0.01 to 0.99.

Language Model IV (pClm)
For this language model the background collection is the complete corpus (AQUAINT document collection *C*). The probability for calculating the perplexity is

$$P(w) = (1 - \lambda)P_{ml}(w|p) + \lambda P(w|C),$$

5.3 Experimental Results

This section discusses the results of our experiments.

Language Model I (pdclm)
This language model is a reproduction of the language model with Jelinek-Mercer smoothing used in Section 4. It reproduces our previous results, which confirmed the validity of results generated by our language modeling toolkit. The plot in Fig 6 shows non-interpolated average precision for different settings of λ. It illustrates that the optimal value of λ is near 0.4.

Language Model II (ppclm)
Fig 6 shows the results using this language model by line with cross as points. The optimal value for λ is 0.05. According to [Zhai and Lafferty, 2001] small λ means more emphasis on relative term weighting, which means that the passage collection has a smaller role in ranking than passage itself. This might be due to small size of passages and variety in topics they discuss. With this language model we observe a 20% decrease over the baseline.

Method	Lambda	MAP
pdclm	0.40	0.114
ppclm	0.05	0.101
pdlm	0.70	0.176
pClm	0.01	0.032

Table 3: Non-interpolated average precisions for the best run of each language model. Passage re-ranking using the document language model for smoothing improves MAP by 39% over the best result from Lemur.

Language Model III (pdlm)

The line with squares in Fig 6 shows the results using this language model. The optimal value for λ is 0.70. The value of λ near middle of the parameter space suggests that both passage and document collection are equally important for ranking. The document is given a bit more importance than the passage, which is quite understandable as passages are of small size and sometimes they miss some related terms from query. With this language model we have more then 38% improvement over the baseline, which is quite a significant improvement. This is no surprise as both document and passage being used discuss the same topic. The related document size is relatively small compared to the document or passage collection, which also contributes to the improvement in results.

Language Model IV (pClm)

Fig 6 shows, using line with diamonds, the results using this language model. The optimal value of λ is 0.01. A small λ means more emphasis on relative term weighting, which means that corpus have nearly no role in ranking the passages. This is because of large size of corpus, with lots of irrelevant terms. It is also clear from Fig 6 that this language model performed worse than all our proposed models.

Table 3 display best results by each language model.

6 Conclusion and Future Work

We have studied the problem of language model smoothing in the context of passage retrieval for QA Systems and compared it with traditional models, including TF-IDF and OKAPI. We then examined three popular interpolation-based smoothing methods (Jelinek-Mercer, Dirichlet Prior, and Absolute Discounting), and evaluated them using the AQUAINT retrieval testing collection.

First we performed document retrieval. Our experiments showed that the Dirichlet Prior performed the best with prior of 1000. Then we carried out passage retrieval and observed that again the Dirichlet Prior performed the best with a prior of 500. With these experiments we established a baseline value. We have defined a number of language models based on the Jelinek-Mercer smoothing technique, and found out that interpolation between language model for passage and single document from which passage is extracted provided more then 38% improvement, which is quite significant for QA Systems.

Table 4 gives list of best runs for document retrieval, passage retrieval and re-ranking experiments. Our best performing language model can be used for real QA Systems. We have used one of the basic approaches to passage generation. One problem with our approach is that it does not take care of the topic shift within a passage. It also does not consider topics which spread over multiple passages. Other more sophisticated passaging techniques could further improve our proposed language model. The language models we have proposed and tested are all unigram models. As previous work depicts, higher order language models will improve retrieval performance.

Step	Method	Parameter	MAP
Document Retrieval	Dirichlet Prior	$\mu = 1000$	0.254
Passage Retrieval	Dirichlet Prior	$\mu = 500$	0.127
Re-ranking	pdlm	$\lambda = 0.70$	0.176

Table 4: Summary of results.

References

[Clarke *et al.*, 2000] C. Clarke, G. Cormack, D. Kisman and T. Lynam. Question answering by passage selection (Multitext experiments for TREC-9). In *Proceedings of the Ninth Text REtrieval Conference (TREC-9)*, 2000.

[Corrada-Emmanuel *et al.*, 2003] A. Corrada-Emmanuel, W. Bruce Croft and Vanessa Murdock. Answer Passage Retrieval for Question Answering. In *CIIR Technical Report IR-283*, 2003.

[Berger and Lafferty, 1999] A. Berger and J. Lafferty. Information Retrieval as statistical translation. In *Proceedings of ACM SIGIR*, pp. 275-281, 1999.

[Zhang and Lee, 2003] A. Berger and J. Lafferty. A Language Modeling Approach to Passage Question Answering. In *Proceedings of the Text REtrieval Conference (TREC 2003)*, pp. 489, 2003.

[Tellex *et al.*, 2003] Stefanie Tellex, Boris Katz, Jimmy Lin, Aaron Fernandes and Gregory Marton. Quantitative Evaluation of Passage Retrieval Algorithms for Question Answering. In *Proceedings of the 26th annual ACM SIGIR conference*, pp. 41 - 47, 2003.

[Callan, 1994] James P. Callan. Passage-level evidence in document retrieval. In *Proceedings of the 17th annual ACM-SIGIR conference*, pp. 302-310, 1994.

[Liu and Croft, 2002] Xiaoyong Liu and W. Bruce Croft. Passage Retrieval Based On Language Models. In *CIKM conference*, pp. 375-382, 2002.

[Cai1 *et al.*, 2004] Deng Cai1, Shipeng Yu2, Ji-Rong Wen and Wei-Ying Ma. Block-based Web Search. In *Proceedings of the 27th annual ACM SIGIR conference*, pp. 456-463, 2004.

[Zhai and Lafferty, 2001] Chengxiang Zhai and John Lafferty. A Study of Smoothing Methods for Language Models Applied to Ad Hoc Information Retrieval. In *Proceedings of the 24th annual ACM SIGIR conference*, pp. 334-442, 2001.

[Agichtein and Gravano, 2000] E. Agichtein and L. Gravano. Snowball: Extracting Relations from Large Plain-Text Collections. In *Proceedings of the 5th ACM Conference on Digital Libraries*, pp. 85-94, 2000.

[Clarke*et al.*, 2000] C. Clarke, G. Cormack and E. Tudhope. Relevance ranking for one to three term queries. In *Information Processing and Management*, pp. 291-311, 2000.

A Word Sense-Oriented User Interface for Interactive Multilingual Text Retrieval

Ernesto William De Luca and Andreas Nürnberger
Faculty of Computer Science
University of Magdeburg
39106 Magdeburg, Germany
{deluca, nuernb}@iws.cs.uni-magdeburg.de

Abstract

In this paper we present an interface for supporting a user in an interactive cross-language search process using semantic classes. In order to enable users to access multilingual information, different problems have to be solved: disambiguating and translating the query words, as well as categorizing and presenting the results appropriately. Therefore, we first give a brief introduction to word sense disambiguation, cross-language text retrieval and document categorization and finally describe recent achievements of our research towards an interactive multilingual retrieval system. We focus especially on the problem of browsing and navigation of the different word senses in one source and possibly several target languages. In the last part of the paper, we discuss the developed user interface and its functionalities in more detail.

1 Introduction

The internet comprises of mainly English documents, but the amount of documents in other languages grows daily. Therefore, the internet is likely to change very quickly from an English language medium to a multilingual information and communication service. Most people have a good passive understanding of a foreign language, but are not usually in the situation to formulate search queries in this foreign language as good as in their mother tongue. Considering that people want to access multilingual information, the importance of their ability of language understanding increases rapidly. At the moment the support provided to navigate multilingual information is not yet so sophisticated that users can access documents over the internet in the seamless and transparent way as they do in their mother tongue.

In order to enable users to access multilingual information, different problems have to be solved: disambiguating the query words, translating the query words, as well as categorizing and presenting the results appropriately. In the following, we briefly discuss these aspects.

1.1 Disambiguating the Query Words

Humans often use polysemous words for searching for documents. Unfortunately, a distinction of the related word senses is difficult [Miller, 2001]. A word is polysemous if it has different meanings (polysemy from Greek poly = *many* and semy = *meanings*). When people search for documents related, e.g., to the word *bank*, they will find different documents related to different meanings of this word (bank as a financial institution, bank as a seat, etc.). Humans are able to disambiguate these polysemous words using their knowledge about the related context, but mostly they can do this using their linguistic context knowledge related strictly to the language [Miller, 2001]. Reading the documents retrieved, they can assign the word sense to its linguistic context. In order to identify the meaning of a polysemous word in an automatic word sense disambiguation task, this linguistic context has to be considered. Working in a multilingual context, words have to be disambiguated both in the native and in other languages (see Section 2.1).

1.2 Translating the Query Words

Retrieving documents in other languages, we have to translate the concepts of the search keywords. Machine translation should help in processing and delivering this information. But as discussed in, e.g., [Peters and Sheridan, 2000], this approach cannot be viewed as a realistic answer to the problem of query translations right now.

The problem of automatically matching documents and queries over languages is not properly solved yet, and therefore it has to be done manually to a great extent. In Section 2 the use of query-related word senses retrieved from the lexical resources and their translation as an alternative solution to this problem is discussed.

1.3 Categorizing and Visualizing the Results

User studies have shown that categorized information can improve the retrieval performance for a user. Thus, interfaces providing category information are more effective than pure list interfaces for presenting and browsing information as shown, for example, in [Dumais *et al.*, 2001], where the effectiveness of different interfaces for organizing search results was evaluated. Users were 50% faster in finding information organized into categories. Similar results based on categories used by Yahoo were presented in [Labrou and Finin 1999].

Motivated by these evaluations, we developed methods in order to provide additional disambiguating information to the documents of a result set retrieved from a search engine in order to enable categorization, restructuring or filtering of the retrieved document result set. Since we cannot expect a perfect word sense disambiguation or categorization of results, an adaptive and error tolerant visualization is required. Thus, the retrieval of information should be supported by an appropriate interactive visualization of results and categories.

2 Word Sense Disambiguation (WSD) and Translation (WST)

The automatic disambiguation of word senses is still a very interesting and challenging research task. Since the 1950's different researchers try to disambiguate words, sentences or documents for different purposes as machine translation, information retrieval and hypertext navigation, content and thematic analysis, grammatical analysis (Part-Of-Speech Tagging), speech or text processing [Ide and Véronis, 1998]. In general terms a word sense disambiguation (WSD) process can be described by two steps:

1. All the senses of the word relevant (at least) to the text or discourse are extracted/found (through a list, categories, ontologies, dictionaries, etc…).
2. Every occurrence of the word is assigned to the appropriate word sense (considering the context and the external knowledge resources).

For disambiguating word senses a variety of association methods (knowledge-driven, data-driven or corpus-based WSD) can be used [Ide and Véronis, 1998]. So far, we only used the knowledge-driven WSD approach, i.e. we make use of linguistic information contained in lexical resources [Peters, 2001], like machine readable dictionaries, thesauri or computational lexicons, in order to obtain a linguistic context description of the word senses. Therefore, lexical resources have to be (automatically) explored using the query words, selecting the concepts based on the linguistic relations that define the different word senses and their linguistic context.

In order to identify the meaning of a polysemous word in a WSD task, we need to recognize also its linguistic context. For this purpose the linguistic context is used in two ways:

1. Bag of words (as in some window surrounding the searched word, as in a bag).
2. Relational information (including information about distance from searched word, syntactic relations, semantic categories, etc.).

The linguistic context knowledge can be accessed from an information retrieval system using the knowledge-driven WSD approach mentioned above.

In order to use linguistic resources for a multilingual approach we have to retrieve not only the concept (word sense) with its linguistic relations, but also its related translations. Some of the linguistic information and the related translations required to disambiguate word senses, as we discussed above, are provided in lexical resources like EuroWordNet [Vossen, 1997]. Besides, this resource can be used for text analysis, computational linguistics and many related areas [Morato *et al.*, 2004]. In the following, we briefly describe the use of EuroWordNet for document retrieval in a multilingual Framework.

2.1 The use of EuroWordNet

Given that we want to retrieve from the web different documents in different languages, we have to analyze the different linguistic contexts of a word in these languages. Therefore, we decided to use the EuroWordNet multilingual lexical database. Its basic structure is the same as the Princeton WordNet [Miller *et al.*, 1993] in terms of SynSets with different semantic relations between them.

EuroWordNet consists of a set of language specific WordNets. Each individual WordNet represents a unique language-internal system of lexicalizations. The Inter-Lingual-Index (ILI) was introduced in order to connect the WordNets of the different languages. Thus, it is possible to access the concepts (SynSets) of a word sense in different languages. It means that we can retrieve one and the same concept in different languages with its related translations and linguistic relations.

In addition to the Inter-Lingual-index, there is also a Domain-Ontology and a Top-Concept-Ontology related to this lexical database. The shared Top-Ontology is a superordinate hierarchy of 63 semantic distinctions for the most important language independent concepts (e.g. Artifact, Natural, Cause, Building) and is interconnected with the ILI through the WordNet-Offsets. Hereby, a common semantic framework for all the languages is given, while language specific properties are maintained individually. The Domain-Ontology was created for use in information retrieval settings in order to obtain specific concepts (only implemented exemplary for the computer terminology). Figure 1 gives an overview over the architecture of the EuroWordNet whereby the single components and its relations are represented.

However, different problems related to the use of (Euro)WordNet for information retrieval have been encountered as discussed in more detail, e.g., in [Mihalcea and Moldovan 2001; Morato *et al.*, 2004; De Luca and Nürnberger, 2006c]. One main problem is that the differentiation of word senses is very often too fine grained for typical information retrieval tasks. One way to obtain a higher granularity is to merge SynSets if they describe a very similar meaning of the same word [De Luca and Nürnberger, 2006a]. For web search, such methods could be used for creating a reduced structure of the ontology hierarchy, having fewer word senses that are carrier of a more distinctive meaning, in order to categorize the documents retrieved [De Luca and Nürnberger, 2006c]. We described a first approach to solve this problem in [De Luca and Nürnberger, 2006a] in a monolingual task.

When we deal with EuroWordNet, these problems persist, and other problems come along. In general, the problem of automatically finding translation of word senses can be solved using such a resource. The use of the Inter-Lingual-Index helps for this purpose. However, the coverage of language-dependent word senses varies from language to language, i.e. from ~20.000 (german) to 150.000 (english) Synsets. Using this lexical resource, we have to take into account the missing (or incomplete) translations contained in the lexical resource, apart from the lexical gaps (word senses that exist in a language and not in another).

2.2 The use of the CARSA Search Engine Framework

The document search in our approach is done using our search engine framework CARSA [Bade *et al.*, 2005]. CARSA is a web services based architecture, which supports the development of context based information retrieval systems. The idea of these systems is to support a user in his search process by, e.g., adapting the search results as well as the interface itself to user specific needs and interests. We decided to divide query results set processing (the information to be presented) from the interface design (information presentation) in order to simplify the

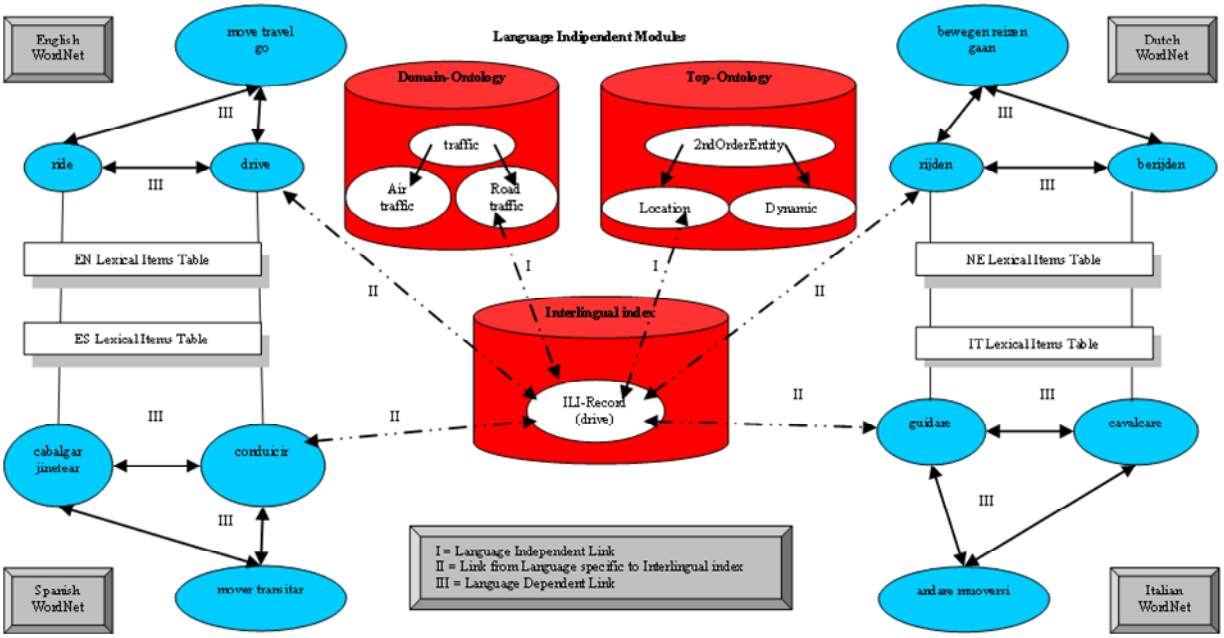


Figure 1. EuroWordNet Architecture (see [Vossen, 1997]).

development of retrieval systems for, e.g., different desktop as well as mobile devices. The central component of the retrieval system is a meta search engine providing methods to restructure and annotate result sets of user queries. Search engines (e.g. Google, local searchers) and the user interfaces are connected to the system by Web services. Using this modular implementation, it is easily possible to extend the system by additional search engines or to integrate different interfaces. An overview of the system architecture of CARSA is given in [Bade *et al.*, 2005].

3 Cross-Language Text Retrieval

After having described the problem of disambiguating word senses in general, we focus in the following on the disambiguation of the words in documents retrieved from an information retrieval system, and particularly in the multilingual framework we want to deal with.

In general an information retrieval system tries to find and retrieve relevant documents related to a user query, with documents and query being in the same language [Abdelali *et al.*, 2003]. Dealing with a multilingual document collection naturally brings up new questions. Being able to read a document in a foreign language does not always imply that a user can formulate appropriate queries in that language as well. Users find cross-language text retrieval particularly useful when they can express their information needs effectively in their mother tongue, while handling with languages they are less confident with [Oard, 1997].

In [Peters and Sheridan, 2000] different methods for multilingual information access are described, addressing the problems of accessing, querying and retrieving useful documents contained in different collections in several languages at any level of specificity, including different computational linguistic processing. The authors distinguish three main approaches for multilingual information access:

1. Machine translation techniques
2. Corpus-based techniques
3. Knowledge-based techniques

They argue that full machine translation (MT) can not be seen as a realistic answer to the problem of matching documents and queries over languages. One weakness of present fully automatic machine translation systems is the limitation of producing high quality translations only in specific domains. Such approaches could substitute every possible translation for a polysemous word, thus increasing recall at the expense of precision. In addition, it does not represent a cost-effective solution for query translation either [Oard and Dorr 1996].

The corpus-based approaches analyze automatically large collections of text with statistical methods. Here, the semantic is given only by translated sentences related across the languages and these approaches are applicable only in a restricted domain.

Since we want to avoid the use of large corpora and translation methods that are not yet providing sufficient quality, our focus is on the use of knowledge-based approaches to enable multilingual information access. These approaches use ontologies, dictionaries (bi- or multilingual) or thesauri in order to enable cross-language text retrieval. Thus, we first try to find all word senses, then retrieve the appropriate translation from the lexical resource and finally categorize documents using the (most likely) proper word sense. Finally, we have to visualize the results according to the user needs as described in more detail in Sect. 5.

For a more detailed description of the three fundamental approaches for multilingual information access, we refer to [Peters and Sheridan, 2000], where a detailed explanation and several references are given.

4 Combining Word Sense Disambiguation within Cross-Language Text Retrieval

In order to combine the word sense disambiguation process within a cross-language retrieval system, we have developed, so far, several tools, e.g., [De Luca and Nürnberger, 2005] and evaluated different disambiguation approaches [De Luca and Nürnberger, 2006a and 2006c]. In the following, we discuss some of the most important aspects. For more details see the referenced publications.

4.1 Tools

The first visualization interface for multilingual search, MultiLexExplorer [De Luca *et al.*, 2006], was developed with a focus on multilingual *explorative* search. MultiLexExplorer combines word sense disambiguation with a text retrieval approach in an interactive framework. It uses lexical resources to support a user in disambiguating documents (retrieved from the web or a local document collection) given the different meanings (retrieved from lexical resources, in our case EuroWordNet) of a search term having unambiguous description in different languages. By visualizing search results grouped by keyword combinations and word senses, the user can discover languages using lexical resources for disambiguating meanings, combining words and their translation. The translations of all possible source language senses are provided in the target language based on the ILI entries of EuroWordNet (see Section 2.1).

The LexiRes tool [De Luca and Nürnberger, 2006b] provides the possibility of restructuring the word senses provided by a lexical resource for information retrieval purposes. Users are usually interested in a small list of meanings with very distinctive features. Since many lexical resources, especially WordNet, provide frequently too fine grained word sense distinctions, we implemented the tool LexiRes that gives the possibility to navigate lexical information, helping authors of already available lexical resources in deleting or restructuring concepts using automatic or manual merging methods, e.g., as described in [De Luca and Nürnberger, 2006a].

These tools were first steps before implementing the interface presented in this paper. Both tools were used to deal with multilingual queries and documents and helped in finding an appropriate visualization of the results and word senses.

4.2 Document Disambiguation and Classification Using the Sense Folder Approach

In this section we briefly introduce the functionality of the Sense Folder Disambiguation. This approach is used to classify documents in *Sense Folders*, which are defined based on context descriptions obtained by merging information from word senses (retrieved from WordNet) with associated linguistic relations as proposed in [De Luca and Nürnberger, 2006c].

First of all we want to semantically disambiguate the query terms (used in the retrieved documents) using WordNet. Therefore, we categorize documents with respect to the meaning of a query term using different linguistic relations retrieved from EuroWordNet. These relations provides us with words defining the context of the query term in order to create *Sense Folders* for its different meanings. Thus, for each (EuroWordNet-) sense of a query term, a *Sense Folder* (prototypical word vector) is created containing:

- all synonyms (the SynSet)
- all hypernyms (the superordinate word), i.e. dividing senses/categories where hypernyms intersect,
- all hyponyms (the subordinate word),
- the belonging glosses (description of the SynSet elements by words that are frequently used in this specific semantic context),
- and the belonging word domain (word context).

These Sense Folders are compared within the words contained in the documents and are used in order to categorize and annotate retrieved documents with their best matching Sense Folder. Every document is first assigned to its most similar Sense Folder and afterwards this classification is revised by a clustering process in order to improve the disambiguation performance [De Luca and Nürnberger, 2006c]. Labels defining the disambiguating classes are then added to each document of the result set. The visualization of such additional information (Fig. 2) should enable a simple navigation through the huge number of documents and, if possible, should restrict information only to the relevant query-related results.

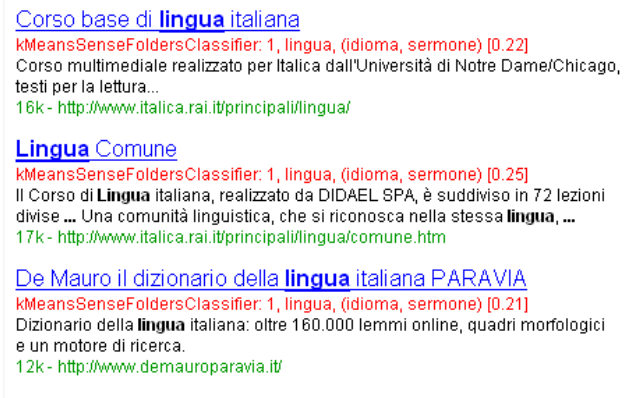


Figure 2 Annotation/classification example searching for the term 'lingua'

Figure 2 shows the implemented categorization techniques combining the knowledge-driven WSD with the knowledge-based text retrieval approach integrated in the developed user interface. The lexical resources are used in order to disambiguate documents (retrieved from the web) given the different meanings (retrieved from lexical resources, in this case EuroWordNet) of a search term. These techniques were combined with clustering processes that strongly improved the overall classification performance. While the pure Sense Folder based approach correctly classified 42% of the documents of a small benchmark dataset, the clustering process was able to assign approximately 70% of the documents to the correct class [De Luca and Nürnberger, 2006c]. More details about these approaches can also be found in [De Luca and Nürnberger, 2005 and 2006c].

5 The User Interface

In the following, we describe an approach for combining cross-language text retrieval, word sense disambiguation and document classification to provide a user-oriented presentation of the search results. We first briefly discuss related work, then we present the implemented user interface that gives the possibility of an interactive multilingual search, and finally we discuss a first evaluation of the automatic merging methods in this setting.

5.1 Related Work

Different work has already been done in dealing with word senses, clustering and multilingual queries. For example, in [Mihalcea and Moldovan, 2001 and Peters *et al.*, 1998] approaches for automatic sense clustering with EuroWordNet were presented. Methods for collapsing similar meanings for query expansion have been discussed in [Moldovan and Mihalcea, 2000].

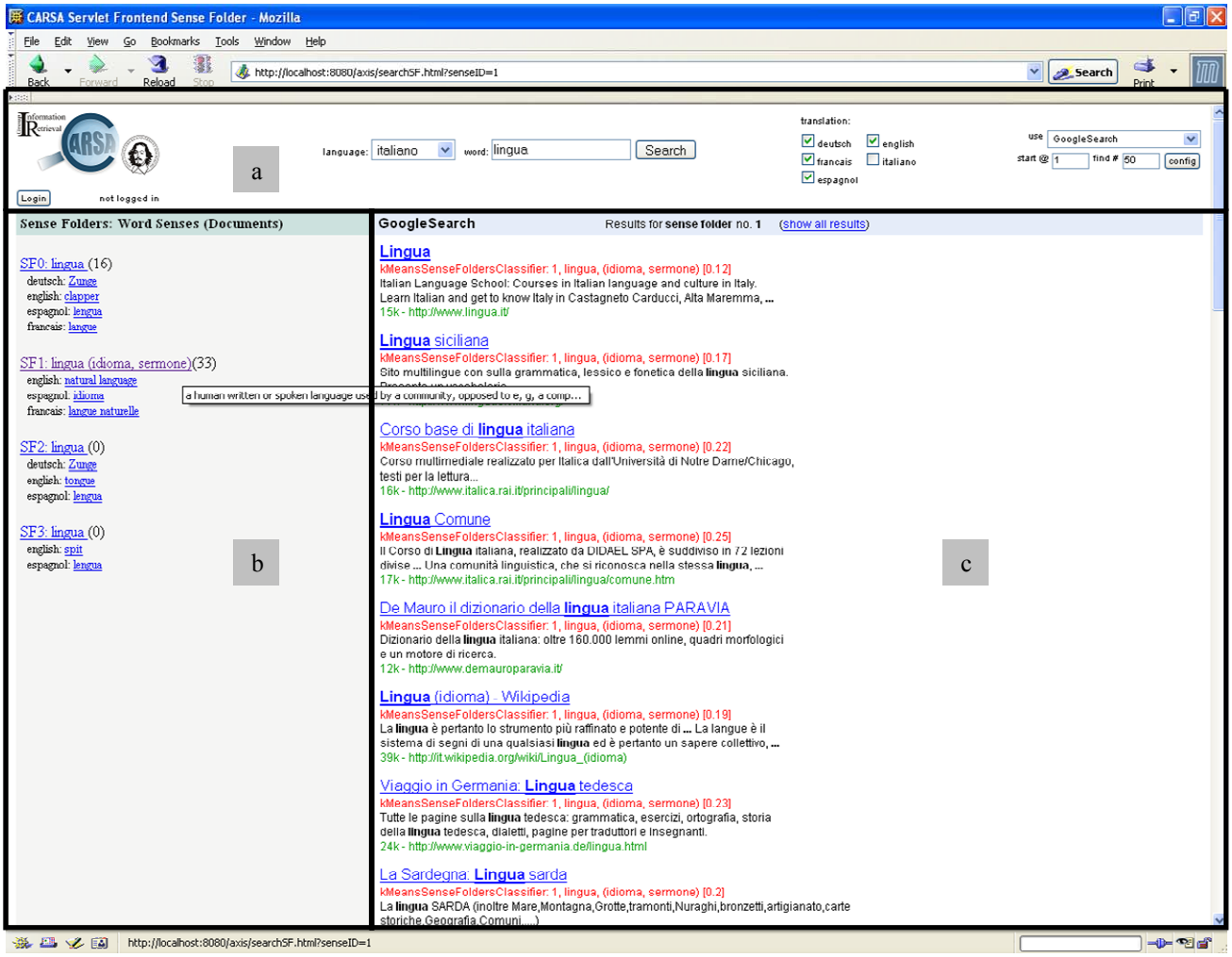


Figure 3. Multilingual User Interface

Peters *et al.*, [1998] developed automatic methods for grouping senses into more coarse-grained sense groups. They started clustering, for example, word senses sharing the same hypernym (calling them sisters) that occurs between two or more senses hyperonyms.

In [Mihalcea and Moldovan, 2001] it was also motivated that there is no need of a fine grain distinction between concepts n a retrieval setting. In this work the authors propose an approach to collapse synsets having very similar meaning or deleting synsets that are rarely used. The similar meanings are collapsed here for query expansion. Also approaches for semantic indexing, e.g. [Gonzalo *et al.*, 1998], show that there is no need for such fine distinctions of word senses.

Different to the approaches mentioned above, we are not working on methods for query expansion, since we think that these approaches usually restrict the result set to much (and thus reduce the recall) and furthermore frequently change the original meaning of the query and thus do not reflect the users intention any longer [Gonzalo *et al.*, 1998]. Our goal is to support the user in the retrieval process by semantic annotation and structuring of result sets without modification of the initial query.

5.2 Interacting with the Multilingual User Interface

Figure 3 shows the user interface that is divided in three main parts:

a) The Search and Configuration Dialogue
b) The Word Sense Presentation
c) The Result List Presentation

A user that interacts with this user interface has to first configure the search before starting (label a). First of all the user chooses the language he wants to search with (as a "starting" source language). Afterwards, other languages (target languages that the user usually is able to speak and understand) can be selected for initializing the multilingual search. Furthermore, a user can also configure which search engines should be used. Of course, default settings are provided.

The configuration dialogue for the linguistic parameters for classification (see Section 5.3) can be started clicking on the button *config* (see Figure 4). However, this dialogue is recommended only for expert users. Here, the user can choose not only the linguistic parameters, but also the classification methods that should be used by the system. Presets for classification are implemented as a default.

After having configured the search parameters, the user can type in the query terms that he is interested in. These keywords are sent to the CARSA meta-search engine and to the Ontology Engine. The meta-search engine retrieves the documents. The documents are retrieved implicitly language-dependently. It means that when we start a search in Italian, we only retrieve Italian documents since the selected search terms are in italian. The Ontology En-

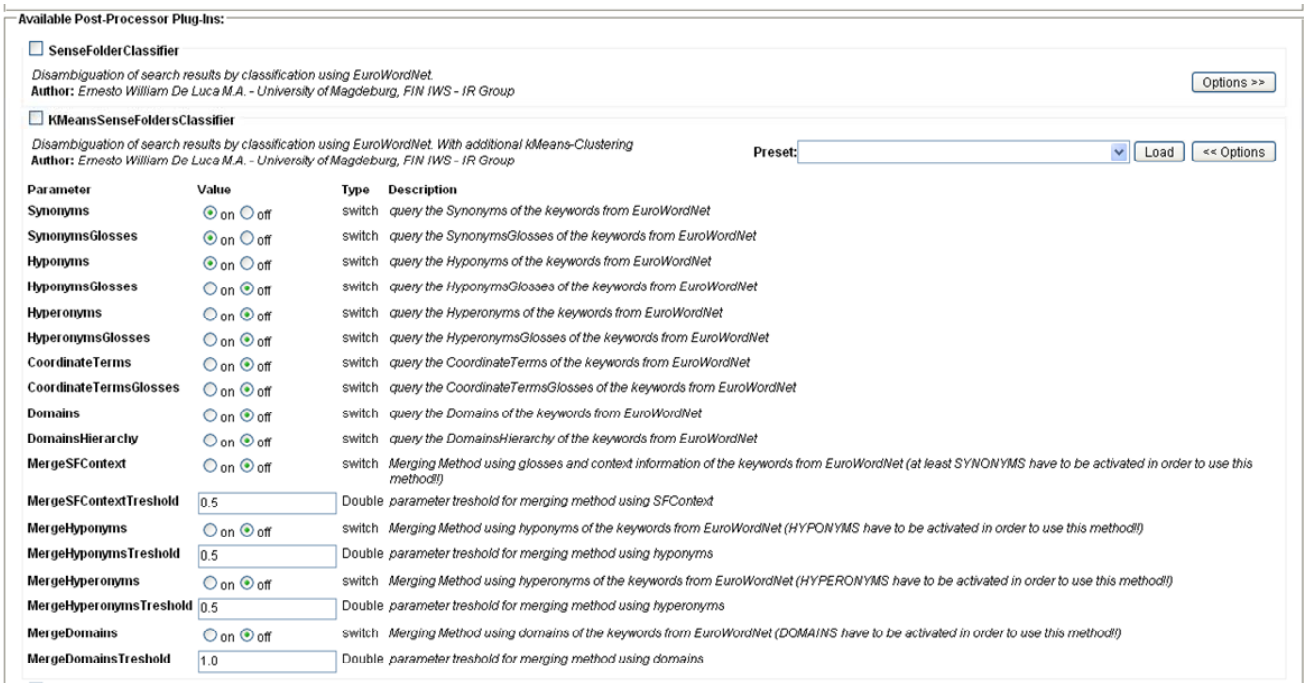


Figure 4. Parameter Configuration

gine is concerned with the process of retrieving the word senses related to the query and the related translations used for the Sense Folder Disambiguation as described in Sect. 4.2. The retrieved word senses are used to filter the results and present them annotated by their meaning. Therefore, every document is labeled with the best matching Sense Folder (see Sect. 4.2). Every Sense Folder is used as class containing the related retrieved documents (label b). A glossary entry is shown in order to help a user in understanding what the word sense means. Such a glossary entry is activated on the mouse rollover event. It is always in English and retrieved from the EuroWordNet ontology.

If we click for example on the Sense Folder 1 (SF1) on the left side of the user interface, the system will show to the user only the documents that are classified by the system as belonging to that word sense, in this case 33 documents that are presented on the right side of the interface (label c), if the user clicks on the word sense. It means that if we are only interested in the documents related to the word sense "natural language" in Italian, we do not have to scan all results in order to retrieve the documents we are interested in. We can just browse the documents related to this word sense; in our case only 33 of 50. However, we like to emphasize that the word sense categorization is not perfect as already mentioned in Sect. 4.2. Therefore, we are still working on visualization methods that are better able to deal with this uncertain classification.

As we can see from Figure 3 not all senses are covered from the documents. It means that when we were looking for documents related to the word sense "lingua" (SF3) in the sense of "spit", we wouldn't find with the first search any related documents.

This interface gives the possibility of a multilingual search. As we can see, every Sense Folder has a translation related to the word senses retrieved for the languages chosen at the beginning of the search process. As we said before, the use of EuroWordNet implies missing (or incomplete) translations and lexical gaps. It means that not all word senses have a 1:1 translation in all foreign languages selected. However, considering our example above, where we are looking for the word sense "lingua" (SF3) in the sense of "spit", we can just click, for example, on the English translation of the word sense, to start automatically a new search with "spit" as a new search word in the English document web collection. The user interface presents then the new word senses related to the query word "spit" and filters the new retrieved documents to the correspondent Sense Folder. Obviously, here a new word sense disambiguation and retrieval process is started.

5.3 Search Configuration

Given that we want to use the word senses for filtering the documents with respect to their meaning, we have to configure the search with the document classification. The user can configure the Sense Folder Classification (or the classification supported by the clustering methods, as in the example with k-Means Clustering) choosing the parameters that characterize the word sense classes used as described in Sect. 4.2. Here the user can choose to activate any linguistic relation and merging method. Choosing the merging methods, thresholds can also be defined. Figure 4 shows the parameter configuration dialog that can be interactively be modified from the user. Depending on which linguistic parameters have been activated, the system classifies the documents. A first evaluation of the combination of the merging parameters has been described in [De Luca and Nürnberger, 2006a].

5.4 Evaluation of the Linguistic Parameters

In the following, we show a first evaluation of the combination of the linguistic parameters. Table 1 shows the results of this evaluation.

For our experimental studies we chose the pre-classified BankSearch web page collection [Sinka and Corne, 2002] consisting of 10,000 web documents classified into 10 equally-sized categories each containing 1,000 web

	SynHyperHypoGlo	SynHyperGlo	SynHypoGlo	SynGlo	SynHyperHypo
SF (Single SynSet „operation“)	**0.42**	0.38	0.40	0.32	0.29
CL (Single SynSet „operation“)	0.55	0.47	**0.54**	0.46	0.37
SF (merged SynSet „operation“)	**0.42**	0.39	0.40	0.30	0.22
CL (merged SynSet „operation“)	0.67	0.66	**0.82**	0.47	0.10
SF(Single SynSet „rule“)	0.36	0.28	**0.43**	0.33	0.26
CL (Single SynSet „rule“)	0.58	0.28	**0.68**	0.52	0.21
SF (merged SynSet „rule“)	0.40	0.31	**0.45**	0.36	0.27
CL (merged SynSet „rule“)	0.79	0.26	**0.87**	0.60	0.19

SF =Sense Folder Classification CL= k-Means Clustering with Sense Folders
Syn=Synonyms Hyper= Hyperonyms Hypo= Hyponyms Glo=Human descriptions

Table 1 Evaluation of linguistic parameters

documents. To each category one of four distinct themes, namely Banking and Finance, Programming Languages, Science, and Sport was assigned.

For the evaluation, we selected the subset of documents that contain the words “rule” or “operation”. The obtained documents were categorized using the pure Sense Folder classification (SF) approach and the clustering (CL) approach. We compared these two different automatic classification with the classification contained in the dataset (based on themes). Since these themes match nicely with the possible meanings of the term “rule” or “operation” described in WordNet (see Table 2 and Table 3), we first run the evaluation using all the SynSet available (related to the themes, but used as “Single SynSets”) and then merging them, mapping one or more SynSets to one Theme (“Exact Match, merged SynSets”). It means that we had first 6 SynSets (not merged) of the two word senses and 4 SynSets after merging semantically very similar word senses (For details on merging SynSets see [De Luca and Nürnberger, 2006a]). We consider in the following SynSet #0 as correctly classifying documents assigned to the banking and finance theme, SynSet #1 for the programming theme, SynSet #2 for the science and SynSet #3 for the sport theme. The SynSets that are considered not belonging to any of the themes have been removed. If the term “rule” or the term “operation” occurs in a document of this dataset it is usually used in the sense of the assigned theme. We can notice here that the best combination is almost always when we use only the combination of the linguistic relations (synonymy, hyponymy and gloss-description) with the merged form of the word senses.

6 Conclusions and Future Work

In this paper we presented a multilingual user interface that helps users in the search process considering the languages they can speak and the word senses they want to navigate in order to retrieve the documents they are looking for. Therefore, we integrated different word sense disambiguation methods in order to automatically categorize retrieved documents with respect to the sense in which a query term is used within the document. The results are presented in groups that can be accessed interactively. Even though, the performance of the word sense disambiguation methods is not yet sufficient and has to be improved, the interface already provides additional information that can help a user in browsing multilingual search results.

In future work, the usability of the current user interface has to be evaluated in order to better understand the needs of users working in a multilingual environment.

Furthermore, the use of EuroWordNet is very helpful, but we are thinking of implementing methods to extend this ontology, because only the English language has more or less acceptable coverage of the language.

The merging methods applied to the word senses can be helpful for a better document classification, but a deeper evaluation should be done and a more detailed analysis of the disambiguation performance is still necessary.

Wordnet SynSet	**Single SynSet**	**Exact Mapping (merged SynSet)**
#0 rule ruler (Metrology)	#1(2)	#1(2) Program
#1 rule formula (Sociology)	#0 (1)	#0(1) Banking
#2 rule regulation (behavior)	none	none
#3 rule formula (Mathematics)	#1 (3)	#1(2) Program
#4 principle rule (rule, law)	#2 (4)	#2 (3) Science
#5 principle rule (generalization)	#2 (5)	#2 (3) Science
#6 rule (religion)	none	none
#7 rule prescript (guide)	none	none
#8 rule (game, sport)	#3 (6)	#3(4) Sport
#9 rule linguistic rule (Linguistics)	none	none
#10 rule (legal authority)	none	none
#11 rule (History Time_Period)	none	none

Table 2. Comparison of WordNet SynSets and restructured SynSets for clustering for the word “rule”.

Wordnet SynSet	**Single SynSets**	**Exact Mapping (merged SynSet)**
#0. operation (being operative)	none	#3 (4) Sport
#1. operation (Commerce)	#0 (1)	#0 (1) Banking
#2. operation, functioning	none	#3 (4) Sport
#3. operation activity	#3 (5)	#3 (4) Sport
#4. operation (Computer Science)	#1 (2)	#1 (2) Program
#5. operation (Military)	none	none
#6. operation (Medicine)	#2 (3)	#2 (3) Science
#7. operation, procedure	#3 (6)	#3 (4) Sport
#8. process, operation, cognitive operation (Psychology)	#2 (4)	#2 (3) Science
#9. operation (Mathematics)	none	#1 (2) Program

Table 3. Comparison of WordNet SynSets and restructured SynSets for clustering for the word “operation”.

References

[Abdelali *et al.*, 2003] Abdelali, J. Cowie, D. Farwell, B. Ogden and S. Helmreich. Cross-Language Information Retrieval using Ontology In: *Proc. of the Conference TALN 2003*, France, 2003.

[Bade *et al.*, 2005] K. Bade, E. W. De Luca, A. Nürnberger and S. Stober. CARSA - An Architecture for the Development of Context Adaptive Retrieval Systems, In: *Proceedings of the 3rd International Workshop on Adaptive Multimedia Retrieval (AMR05).*

[De Luca and Nürnberger, 2005] E. W. De Luca and A. Nürnberger. A Meta Search Engine for User Adaptive Information Retrieval Interfaces for Desktop and Mobile Devices In: *Proc. of the Workshop on Personalized Information Access (PIA 2005)*, In Conj. with the Int. Conference on User Modelling (UM'05), UK, 2005.

[De Luca and Nürnberger, 2006a] E. W. De Luca and A. Nürnberger. Rebuilding Lexical Resources for Information Retrieval using Sense Folder Detection and Merging Methods. In: *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC 2006)*. Genova, Italy, 2006.

[De Luca and Nürnberger, 2006b] E. W. De Luca and A. Nürnberger. LexiRes: A Tool for Exploring and Restructuring EuroWordNet for Information Retrieval". In: *Proceedings of the Workshop on Text-based Information Retrieval (TIR-06)*. In conjunction with the 17th European Conference on Artificial Intelligence (ECAI'06). Riva del Garda, Italy / Aug 29th, 2006.

[De Luca and Nürnberger, 2006c] E. W. De Luca and A. Nürnberger. Using Clustering Methods to Improve Ontology-Based Query Term Disambiguation. In: *International Journal of Intelligent Systems, Volume 21, 693-709, John Wiley & Sons*, 2006.

[De Luca *et al.*, 2006] E. W. De Luca, S. Hauke, A. Nurnberger and S. Schlechtweg. Using Multilingual Ontologies for Adaptive Web-based Language Exploration. In: *Proceedings of the International Workshop on Applications of Semantic Web Technologies for E-Learning (SW-EL06)*. In Conjunction with the International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems (AH2006). Dublin, Ireland, 2006.

[Dumais *et al.*, 2001] S. T. Dumais, E. Cutrell and H. Chen. Bringing order to the web: Optimizing search by showing results in context. In: *Proc. of the CHI'01*, 2001, 277-283.

[Gonzalo *et al.*, 1998] J. Gonzalo, F. Verdejo, I. Chugur, and J. Cigarran. Indexing with WordNet synsets can improve text retrieval. In: *Proceedings ACL/COLING Workshop on Usage of WordNet for Natural Language Processing*, 1998.

[Ide and Véronis, 1998] N. Ide and J. Véronis. Word Sense Disambiguation: The State of the Art. In: *Computational Linguistics*, Volume 14, Part 1, 1998.

[Morato *et al.*, 2004] J. Morato, M. Marzal, J. Lloréns and J. Moreiro. WordNet Applications. In: *Proc. of the 2nd Int. Conf. Global WordNet*, Brno, Czech Rep. 2004.

[Labrou and Finin 1999] Y. Labrou and T. Finin. Yahoo! as an ontology: using Yahoo! categories to describe documents. In: *Proc. of 8th Int. Conf. on Information and Knowledge Management*, 1999.

[Mihalcea and Moldovan, 2001] Rada Mihalcea and Dan Moldovan, Automatic Generation of a Coarse Grained WordNet. In: *Proceedings of NAACL Workshop on WordNet and Other Lexical Resources*, Pittsburgh, PA, June 2001.

[Miller, 2001] G. A. Miller. Ambiguous Words. In: *Impacts Magazine*. Publ. on KurzweilAI.net, 2001.

[Miller *et al.*, 1993] G. A. Miller, R. Beckwith, C. Fellbaum, D. Gross and K. Miller. Five papers on WordNet. ftp.cogsci.princeton.edu/pub/wordnet/5paper s.ps. 1993.

[Moldovan and Mihalcea, 2000] Dan Moldovan and Rada Mihalcea, Using WordNet and Lexical Operators to improve Internet Searches. In: *IEEE Internet Computing, vol. 4 no. 1*, January 2000.

[Oard, 1997] D. W. Oard. Alternative Approaches for Cross-Language Text Retrieval, College of Library and Information Services, University of Maryland, http://www.ee.umd.edu/medlab/filter/sss/papers/oard/paper.html, 1997.

[Oard, and Dorr 1996] D. W. Oard & B. J. Dorr, "A Survey of Multilingual Text Retrieval, UMIACS TR-96-19, University of Maryland, College Park, MD, 1996.

[Peters and Sheridan, 2000] C. Peters and P. Sheridan. Multilingual Information Access. In: *Lectures on Information Retrieval, Third European Summer-School*, ESSIR 2000, Varenna, Italy, 2000.

[Peters, 2001] W. Peters. Lexical Resources, In: *NLP group Department of Computer Science*, University of Sheffield, http://phobos.cs.unibuc.ro/roric/lexintroduction.html, 2001.

[Peters *et al.*, 1998] Peters, W., Peters, I., Vossen, P. Automatic sense clustering in EuroWordNet. In: *Proceedings of the 1st international conference on Language Resources and Evaluation*. Spain, 1998.

[Sinka and Corne, 2002] Sinka, M.P., Corne, D.W. A large benchmark dataset for web document clustering, in Abraham, A., Ruiz-del-Solar, J., Koeppen, M. (eds.), *Soft Computing Systems: Design, Management and Applications, Volume 87 of Frontiers in Artificial Intelligence and Applications* (2002) 881-890.

[Vossen, 1997] P. Vossen. EuroWordNet: a multilingual database for information retrieval. In: *Proceedings of the DELOS workshop on Cross-language Information Retrieval*, Zurich, 1997.

A network model approach to document retrieval taking into account domain knowledge

Peter Scheir, Stefanie N. Lindstaedt
Know-Center
Inffeldgasse 21a, 8010 Graz, Austria
{pscheir, slind}@know-center.at

Abstract

We preset a network model for context-based retrieval allowing for integrating domain knowledge into document retrieval. Based on the premise that the results provided by a network model employing spreading activation are equivalent to the results of a vector space model, we create a network representation of a document collection for retrieval. We extended this well explored approach by blending it with techniques from knowledge representation. This leaves us with a network model for finding similarities in a document collection by content-based as well as knowledge-based similarities.

1 Introduction

The work presented here originated in the context of the project APOSDLE[1]. One of the objectives of this project is contextualized delivery of information to knowledge workers. For this task we create formal models (task model, domain model, competence model) of the different aspects of the work context to represent the environment the knowledge worker operates in. We use this information, besides content-based analysis of resources, for retrieving information relevant to the current work situation.

When defining a model[2] of the context of a knowledge worker (cf. [Ulbrich *et al.*, 2006]) we noticed that there are three classes[3] of objects that could be used for querying an information retrieval system:

- A *set of concepts* of a knowledge representation that describes the situation of the knowledge worker, for example the current actions a person performs or the competencies he or she acquires.

  This concepts stem from the formal models that are used to represent the context of the knowledge worker.

- A *set of documents* that are related to the current situation of the knowledge worker, for example the document template he or she is currently interacting with, or the process documentation the person is reading.

  In our approach documents are related to concepts from the task and the domain model. This enables us to infer which documents are associated with the current task and vice versa.

- A *set of terms* which are related to his or hers current situation, examples for such terms would be parts of documents the person currently views or a text he or she currently types.

  The set of terms is not related to any of the models that span our context model. Nevertheless we think of it as a vital addition to our approach to retrieval.

To increase the chances for successfully supporting the worker with resources, a model taking all three classes of objects (concepts, terms, documents) as query items into account was needed. In this contribution we present our suggestion for such a model and discuss the technical feasibility of a system implementing the model. Additionally we will present related work in the field of network models in information retrieval.

Our contribution is structured as follows: First (in Section 2) we give an overview on network models in information retrieval, explain underlying concepts such as spreading activation and present systems operating on knowledge representations as well as on document collections. Then in Section 3 we introduce our model and the challenges related to our approach. We will discuss the technical realization of the model, present the lessons we have learned so far and point out the benefits of our approach. We conclude this contribution (Section 4) with a brief discussion and future tasks on our research agenda.

2 Network models in information retrieval

Network models have a long tradition in information retrieval and experienced great popularity in the 1980s, inspired by the rise of neural networks. Systems using network representations often employ a processing technique called spreading activation. Spreading activation originates from cognitive psychology where it servers as mechanism for explaining how knowledge is represented and processed in the human brain (cf. [Anderson, 1983]). The human mind is modelled as a network of nodes, which represent concepts and are connected by edges. Starting from a set of initially activated nodes, activation spreads over the edges to their neighbours. Those nodes with the highest level of energy are seen to be the most similar to the set of nodes activated initially. A detailed introduction to spreading activation in information retrieval can be found in [Crestani, 1997].

As in this section we will give an overview on network models in the field of information retrieval, we find

[1] Advanced Process Oriented and Self-Directed Learning Environment - http://www.aposdle.org/

[2] The context-model used in APOSDLE is based on a meta-model that defines mappings between a task model, a domain model and a competence model. These models are created according to the current application domain of the system.

[3] Currently the deduction of these classes from the meta-model is not described in a formalized way.

it important to note that our view on information retrieval is inspired by Raphael [Raphael, 1968], who sees information retrieval as document retrieval as well as fact retrieval. Therefore we see document-content-based retrieval as well as knowledge-representation-based retrieval as components of information retrieval. In the work presented here we introduce a network model covering both of these aspects for finding resources to support knowledge workers.

Two *divergent* applications of network model in information retrieval exist which we aim to unify in our approach: (1) retrieval in knowledge bases and (2) retrieval in document bases. We will now give a short overview about those applications.

2.1 Knowledge retrieval using network models

There exist several classical and contemporary systems that model a knowledge base as a network of nodes that is searched by spreading activation. Examples are documented in [Cohen and Kjeldsen, 1987] or [Berger *et al.*, 2004]. While knowledge representations are a current issue since the early days of artificial intelligence research, it was the ongoing effort of the Semantic Web community that lead to sound mechanisms for creating knowledge representations in the form of ontologies (see [Guarino, 1998] for an exact definitions of an ontology). New standards for knowledge representation where defined, new methodologies and tools for developing ontologies where developed. Examples of recent systems, employing a network model and spreading activation for identifying similarity in ontologies, exist [Alani *et al.*, 2003] [Rocha *et al.*, 2004].

2.2 Document retrieval using network models

With the rise of neural networks researchers tried applying this paradigm to information retrieval, resulting in a neural network like representation of document collections. One of the first systems employing a network representation for document retrieval was AIR [Belew, 1989]. In AIR the document collection is modelled as a network of nodes. Two classed of nodes exist, one for representing documents and one for representing terms contained in the documents. Some systems for document retrieval also focused on a central aspect of neural network: learning. So incorporate [Belew, 1989] and [Kwok and Grunfeld, 1996] relevance feedback of the user by reweighing connections between documents and terms into the network representation.

3 A network model for integrating domain knowledge into document retrieval

The approach presented here combines two sources for retrieving resources for the knowledge worker. On the one hand content-based similarity of documents is used, while on the other hand similarities stem from a knowledge representation. For integrating the two sources, a network representation is used as this form of model is well suited for both aspects of our approach (cf. Sections 2.1 and 2.2) and allows for the integration of the two aspects. The resulting model can be represented as three layers architecture: (1) A layer for documents, (2) a layer for terms extracted from documents and (3) a layer for concepts originating from the ontology(s) used (see Figure 1).

The document layer and the term layer stem from the document base present in the system. For every document

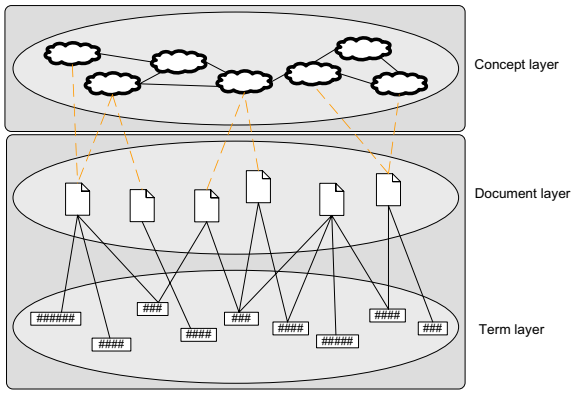


Figure 1: The network model consisting of concept layer, document layer and term layer

present in the document base a document node in the document layer is created. Document nodes feature a unique identifier to be precisely distinguishable. The term nodes result from the terms contained in the documents. Classical text indexing techniques are used to extract the terms from the documents, additional pre-processing steps as stemming can be added to the pre-processing queue, before adding term nodes to the term layer. If a term is contained in a document the term node is connected to the document node by an undirected edge in the network. The edge is weighted by term-frequency - inverse-document-frequency (tf-idf). This data structure can be represented as a matrix similar to the term-document-matrix used in other information retrieval approaches. The document and term layer of the model presented here are comparable to the approaches reviewed in 2.2.

The concept layer is created by transforming a knowledge representation into a network structure. This is done by using the underlying graph structure of the knowledge representation, in which nodes symbolizing concepts are connected by edges. The edges already present in the graph data structure can be weighted using various approaches, one of them would be to take the paths between two nodes into account (cf. [Rocha *et al.*, 2004]). The concept layer is comparable to the systems described in Section 2.1.

Items from every layer can be used to initiate a query. Therefore this layer concept fulfils our requirements of a retrieval system for contextualized information delivery (cf. Section 1), which should be able to build a query from a set of concepts, a set of terms and a set of documents. The network is searched using spreading activation (cf. Section 2). Depending on the current context of a user, a set of concepts nodes, term nodes and documents nodes is activated, when the network is queried. From these nodes energy spreads over the network, leaving those document nodes with the most energy that are closest related to the set of query nodes. Theses nodes will be presented as result of the contextualized retrieval process. As the edges connecting node are undirected activation can spread in both direction over an edge.

3.1 Challenges resulting from this approach

After the creation of the three layers the challenge of combining the already well integrated document and term layer with the concept layer exists (symbolised by the dashed lines in Figure 1). While the document and term layers stem from resources present in the document base of the

system (cf. Section 2.2), the concept layer originates from the knowledge representation(s) used (cf. Section 2.1). Per se those two tiers are not connected, as the domain models are created by experts for a special purpose, while the resources in the document base of the system are created by the workers in the company during their daily job. As manually relating concepts from formal models to documents is a burdensome task an interesting challenge is raised that is similar to one existing when trying to establish a Semantic Web: A lot of resources already exist in the current form of the web, but most of them are not annotated semantically. Therefore our research effort will precisely observe and contribute to current and future developments in fields such as ontology learning [Buitelaar *et al.*, 2005] and the (semi-)automatic semantic annotation of resources with semantic metadata [Handschuh and Staab, 2003].

Additionally, we find it important to integrate support for reasoning over the semantics of edges that stem from a knowledge representation. This is similar to the approach presented in [Wolverton, 1995] where the spread of activation is guided by a reasoning engine that decides which edges to follow depending on the retrieval task.

3.2 Related work

Currently no approaches to information retrieval are known to the authors, which provide information based on content-based similarities of resources and a knowledge representation by combining them into a network model. An effort into this direction was I3R [Croft and Thompson, 1987] where a network structure was introduced allowing for connecting documents, extracted terms and concepts from a semantic network. In the presented prototype only content-based similarity of text documents and similarities based on the same author and co-referenced works were used. In [Agosti and Crestani, 1993] a similar network structure is suggested. It remains unclear in both approaches how content-based similarity and the semantic layer are connected.

3.3 Technical realization

To explore the functionality of our model two core components of our retrieval engine need to be built: (1) An implementation of the spreading activation algorithm is needed and (2) an efficient way of storing the network data-structure used for the retrieval task. We have already implemented two research prototypes for testing purposes (one for the concept layer and one for the term and document layer). While we initially started with an in-memory representation of the whole network structure, formed by objects representing nodes that reference other objects, we have dropped this idea in favour of an search-index-based representation (on hard-disk) of the network. In the following we will explain why:

In our model three types of nodes exist: One for concepts from a knowledge representation, one for document and one for terms. As in our model a document node can be connected to term nodes and concept nodes only, and no connections between terms nodes and terms nodes or documents nodes and documents nodes exist, the graph representation of our model is sparse. For this reason we store our network as an adjacency list as this is the general accepted procedure in managing sparse graphs (cf. [Sedgewick and Schidlowsky, 2003]). This approach equals a large hash table, with source nodes as keys and edge, destination node pairs as values. The index of a standard text search engine like Lucene[4] is well suited to perform the task of storing this adjacency list representation of the network. For example in [Lux and Granitzer, 2005] our colleagues demonstrated how to efficiently use Lucene's index for the task of graph retrieval.

Using the search-index-based approach presented here we are still able to use our existing implementation of spreading activation algorithms, gain the benefit of easy serializability of the network structure and avoid the problem of high memory usage.

3.4 Lessons learned

To assuring the quality of our model at an early stage we did a prototypical implementation of a system, consisting of document nodes and term nodes, employing spreading activation for search. The built system can be compared to the ones described in Section 2.2. Our initial goal was to evaluate our system against the vector space model, similar to the work done by [Salton and Buckley, 1988]. We started building a system with the same indexing back-end as our spreading-activation-based system but with a vector-space-model-based search. During the implementation of the vector space model we were surprised by the similarity of the code for a basic vector space model implementation to the one for a basic the network model using spreading activation. This leads us to further research which we will now briefly summarize:

When [Salton and Buckley, 1988] did the first comparison between the vector space model and network models using spreading activation it ended in favour of the vector space model. We (as [Crestani, 1997]) believe that these results come from the fact that in this comparison a mature version of the vector space model was compared to a rather basic form of the network model. [Wilkinson and Hingston, 1991] present a two layer network model (term and document nodes) which employed the cosine measure from the vector space model by weighting the edges between term and document nodes using tf-idf. This already allowed for speculations on the similarity between the two models. As noted by [Mandl, 2000] in 1994 [Mothe, 1994] demonstrated theoretically and empirically that the results from a network model using spreading activation in the first wave of spreading are identical to those returned by the vector space model, if the same weighting functions are used for both models.

We see this finding as an important quality assurance measure for our work as we can assume that the results of the network model built from documents and terms is equivalent to that of a vector space model if the same weighting techniques are used. We can now extend this model by blending it with a knowledge representation.

3.5 Benefits of our approach

Our approach to context-based retrieval using a network model allows for the integration of domain knowledge in the form of ontologies into document retrieval. In existing systems knowledge representations are statically implemented and are not easily changeable, as it is in our case.

Advantages of the employed approach of search based on a network representation and spreading activation are (cf. [Alves and Jorge, 2005]): The independence of the content type of objects present in the network (but of course different similarities measures have to be defined) and the robustness to missing information. A system built on our

[4] http://lucene.apache.org/

model allows for ostensive [Campbell and van Rijsbergen, 1996] retrieval (a form of retrieval where a user not explicitly formulates a query for a search, but selects material that currently is useful for him), as needed for the user context-based approach.

As spreading activation can be mapped to a distributed web environment using a messaging approach and our approach incorporates knowledge representations with document collections we think that the presented method for retrieval should fit well for searching the Semantic Web.

4 Conclusion and Future Work

We have presented our effort on integrating content-based similarity of documents with a knowledge representation by using a network structure for the task of context-based retrieval. While we are satisfied with our results so far, several issues remain to address: On the theoretical side more research on the integration of the conceptual layer and the document layer (cf. Section 3.1) has to be done, here we want to employ methods from the fields of ontology learning and (semi-)automatic semantic annotation of resources. The more technical part of our future work will focus on the implementation of the retrieval system itself, following the approach presented in the technical realization part of this contribution (Section 3.3).

Acknowledgments

We thank Thomas Mandl and Josiane Mothe for their support on the equivalency of the vector space model and network models using spreading activation. We also thank our colleague Armin Ulbrich for his founding work on the context-model and his helpful comments.

This work has been partially funded under grant 027023 in the IST work programme of the European Community. The Know-Center is funded by the Austrian Competence Center program Kplus under the auspices of the Austrian Ministry of Transport, Innovation and Technology (www.ffg.at/index.php?cid=95) and by the State of Styria.

References

[Agosti and Crestani, 1993] Maristella Agosti and Fabio Crestani. A methodology for the automatic construction of a hypertext for information retrieval. In *SAC '93: Proceedings of the 1993 ACM/SIGAPP symposium on Applied computing*. ACM Press, 1993.

[Alani *et al.*, 2003] Harith Alani, Srinandan Dasmahapatra, Kieron O'Hara, and Nigel Shadbolt. Identifying communities of practice through ontology network analysis. *IEEE Intelligent Systems*, 18(2):18–25, 2003.

[Alves and Jorge, 2005] Mario A. Alves and Alipio M. Jorge. Minibrain: a generic model of spreading activation in computers, and example specialisations. In *ECML/PKDD 2005 workshop 'Subsymbolic paradigms for learning in structured domains'*, 2005.

[Anderson, 1983] John R. Anderson. A spreading activation theory of memory. *Journal of Verbal Leaning and Verbal Behaviour*, 22:261–295, 1983.

[Belew, 1989] Richard K. Belew. Adaptive information retrieval: using a connectionist representation to retrieve and learn about documents. In *Proceedings of the 12th annual international ACM SIGIR conference on Research and development in information retrieval*, 1989.

[Berger *et al.*, 2004] Helmut Berger, Michael Dittenbach, and Dieter Merkl. An adaptive information retrieval system based on associative networks. In *Proceedings of the 1st Asia-Pacific Conference on Conceptual Modelling*, 2004.

[Buitelaar *et al.*, 2005] Paul Buitelaar, Philipp Cimiano, and Bernardo Magnini, editors. *Ontology Learning from Text: Methods, Evaluation and Applications*. IOS Press, 2005.

[Campbell and van Rijsbergen, 1996] Iain Campbell and Cornelis J. van Rijsbergen. The ostensive model of developing information needs. In *of 2nd International Conference on Conceptions of Library Science*, 1996.

[Cohen and Kjeldsen, 1987] Paul R. Cohen and Rick Kjeldsen. Information retrieval by constrained spreading activation in semantic networks. *Inf. Process. Manage.*, 23:255–268, 1987.

[Crestani, 1997] Fabio Crestani. Application of spreading activation techniques in information retrieval. *Artif. Intell. Rev.*, 11:453–482, 1997.

[Croft and Thompson, 1987] W. B. Croft and R. H. Thompson. I3R: a new approach to the design of document retrieval systems. *J. Am. Soc. Inf. Sci.*, 38:389–404, 1987.

[Guarino, 1998] Nicola Guarino. Formal Ontology and Information Systems. In *International Conference On Formal Ontology In Information Systems*, 1998.

[Handschuh and Staab, 2003] Siegfried Handschuh and Steffen Staab, editors. *Annotation for the Semantic Web*. IOS Press, 2003.

[Kwok and Grunfeld, 1996] K.L. Kwok and L. Grunfeld. TREC-4 Ad-Hoc, Routing Retrieval and Filtering Experiments using PIRCS. In *The Fourth Text REtrieval Conference (TREC-4)*, 1996.

[Lux and Granitzer, 2005] Mathias Lux and Michael Granitzer. A fast and simple path index based retrieval approach for graph based semantic descriptions. In *Proceedings of the Second International Workshop on Text-Based Information Retrieval*, 2005.

[Mandl, 2000] Thomas Mandl. Tolerant and adaptive information retrieval with neural networks. In *Global Dialogue. Science and Technology Thinking the Future at EXPO 2000 Hannover*. 2000.

[Mothe, 1994] Josiane Mothe. Search mechanisms using a neural network model. In *Proceedings of the RIAO 94 (Recherche dInformation assiste par Ordinateur)*, 1994.

[Raphael, 1968] Bertram Raphael. SIR : Semantic Information Retrieval. In *Semantic Information Processing*. MIT Press, Cambridge, MA, 1968.

[Rocha *et al.*, 2004] Cristiano Rocha, Daniel Schwabe, and Marcus Poggi Aragao. A hybrid approach for searching in the semantic web. In *Proceedings of the 13th international conference on World Wide Web*, 2004.

[Salton and Buckley, 1988] G. Salton and C. Buckley. On the use of spreading activation methods in automatic information. In *SIGIR '88: Proceedings of the 11th annual international ACM SIGIR conference on Research and development in information retrieval*, 1988.

[Sedgewick and Schidlowsky, 2003] Robert Sedgewick and Michael Schidlowsky. *Algorithms in Java, Part 5: Graph Algorithms*. Addison-Wesley, 2003.

[Ulbrich *et al.*, 2006] Armin Ulbrich, Peter Scheir, Stefanie N. Lindstaedt, and Manuel Goertz. A context-model for supporting work-integrated learning. In *Innovative Approaches for Learning and Knowledge Sharing - First European Conference on Technology Enhanced Learning*, 2006.

[Wilkinson and Hingston, 1991] Ross Wilkinson and Philip Hingston. Using the cosine measure in a neural network for document retrieval. In *Proceedings of the 14th annual international ACM SIGIR conference on Research and development in information retrieval*, 1991.

[Wolverton, 1995] Michael Wolverton. An investigation of marker-passing algorithms for analogue retrieval. In *Case-Based Reasoning Research and Development*, 1995.

Hashing-basierte Indizierung: Anwendungsszenarien, Theorie und Methoden

Benno Stein und **Martin Potthast**

Fakultät Medien, Mediensysteme
Bauhaus-Universität Weimar, 99421 Weimar, Germany
benno.stein@medien.uni-weimar.de
martin.potthast@medien.uni-weimar.de

Abstract

Hashing-basierte Indizierung ist eine mächtige Technologie für die Ähnlichkeitssuche in großen Dokumentkollektionen [Stein 2005]. Sie basiert auf der Idee, Hashkollisionen als Ähnlichkeitsindikator aufzufassen – vorausgesetzt, dass eine entsprechend konstruierte Hashfunktion vorliegt. In diesem Papier wird erörtert, unter welchen Voraussetzungen grundlegende Retrieval-Aufgaben von dieser neuen Technologie profitieren können.

Weiterhin werden zwei aktuelle, hashing-basierte Indizierungsansätze präsentiert und die mit ihnen erzielbaren Verbesserungen bei der Lösung realer Retrieval-Aufgaben verglichen. Eine Analyse dieser Art ist neu; sie zeigt das enorme Potenzial maßgeschneiderter hashing-basierter Indizierungsmethoden wie zum Beispiel dem Fuzzy-Fingerprinting.

1 Hintergrund

Vereinfachend gesagt behandelt das textbasierte Information-Retrieval die zielgerichtete Suche in einer große Menge D von Dokumenten. In diesem Zusammenhang unterscheiden wir zwischen dem „realen“ Dokument $d \in D$ in Form eines Papiers, eines Buchs oder einer Web-Seite, und seiner (Computer)repräsentation $\mathbf{d}$ in Form eines Wortvektors, eines Suffixbaums oder einer Signaturdatei. Sei $\mathbf{D}$ die Menge der Repräsentationen aller realen Dokumente aus D.

In vielen Anwendungen ist die (Computer)repräsentation $\mathbf{d}$ eines Dokuments ein m-dimensionaler Vektor, so dass sich die Objekte in $\mathbf{D}$ als Elemente eines m-dimensionalen Vektorraums auffassen lassen. Die Ähnlichkeit zwischen zwei Dokumenten d_1, d_2 sei als umgekehrt proportional zur Distanz zwischen den Vektoren $\mathbf{d_1}, \mathbf{d_2} \in \mathbf{D}$ vereinbart. Zur Messung der Ähnlichkeit dient eine Funktion $\varphi(\mathbf{d_1}, \mathbf{d_2})$, die auf das Intervall $[0; 1]$ abbildet, wobei 0 keine Ähnlichkeit und 1 maximale Ähnlichkeit bedeutet. φ kann zum Beispiel auf der l_1-Norm, der l_2-Norm oder auf dem Winkel zwischen zwei Vektoren beruhen.

Offensichtlich maximiert das zu d ähnlichste Dokument $d' \in D$ die Funktion $\varphi(\mathbf{d}, \mathbf{d}')$, und offensichtlich kann d' durch eine lineare Suche in $\mathbf{D}$ gefunden werden. Weniger bekannt ist, dass sich die Bestimmung von d' nicht schneller als in $O(|\mathbf{D}|)$ bewerkstelligen lässt, falls die Dimensionalität m des Vektorraums etwa 10 oder mehr beträgt [Weber *et al.* 1998]. An diesem Punkt setzt die Idee der hashing-basierten Indizierung an: Mithilfe von Hashing lässt sich in quasi konstanter Zeit feststellen, ob $\mathbf{d}$ ein Element in $\mathbf{D}$ ist. Dieses Konzept ist auf die Ähnlichkeitssuche übertragbar, falls eine Hashfunktionen $h_\varphi : \mathbf{D} \to U$ existiert, die eine Menge $\mathbf{D}$ von Dokumentrepräsentationen auf ein Universum $U, U \subset \mathbf{N}$ von Hashwerten abbildet und die folgende Eigenschaft besitzt [Stein 2005]:

$$h_\varphi(\mathbf{d}) = h_\varphi(\mathbf{d}') \;\Rightarrow\; \varphi(\mathbf{d}, \mathbf{d}') \geq 1 - \varepsilon, \qquad (1)$$

mit $\mathbf{d}, \mathbf{d}' \in \mathbf{D}$ und $0 < \varepsilon \ll 1$. Anders ausgedrückt, eine Hashkollision zwischen zwei Elementen aus $\mathbf{D}$ kann als Indiz für eine hohe Ähnlichkeit zwischen ihnen gewertet werden.

Aufbauend auf dieser Idee werden in Abschnitt 2 Anwendungsszenarien diskutiert, bei denen sich durch hashing-basierte Indizierung die Retrieval-Performanz und die Ergebnisqualität signifikant verbessern lässt. Abschnitt 3 stellt zwei hashing-basierte Indizierungsverfahren vor, die im textbasierten Information-Retrieval anwendbar sind, und Abschnitt 4 präsentiert Ergebnisse einer vergleichenden Analyse der beiden Ansätze für ausgewählte Retrieval-Aufgaben.

2 Aufgaben und Anwendungsszenarien im textbasierten Information-Retrieval

Sei T die Menge aller Worte, die in den Dokumentrepräsentationen $\mathbf{d} \in \mathbf{D}$ verwendet werden. Die wichtigste Datenstruktur zur Indizierung von D ist die invertierte Liste [Witten *et al.* 1999; Baeza-Yates und Ribeiro-Neto 1999]. Jedes Wort $t \in T$ wird auf eine sogenannte Vorkommensliste abgebildet, die für jedes Vorkommen von t in den jeweiligen Dokumenten d_i, $i = 1, \ldots, o$, eine eindeutige Referenz auf die entsprechende Position innerhalb von d_i speichert.

Sei eine – möglicherweise sehr große – Dokumentkollektion D gegeben, die sowohl durch eine invertierte Liste, μ_i, als auch durch einen Hashindex, μ_h, indiziert ist:

$$\mu_i : T \to \mathcal{D}$$
$$\mu_h : \mathbf{D} \to \mathcal{D}$$

$\mathcal{D}$ bezeichnet die Potenzmenge von D. Während die invertierte Liste μ_i verwendet wird, um Wortanfragen zu beantworten, ist der Hashindex μ_h besonders dazu geeignet, Anfragen zu behandeln, die in Form eines Beispieldokuments formuliert sind. Bei dieser Art von Anfragen werden alle zu dem Beispieldokument ähnlichen Dokumente gesucht. Abschnitt 3 zeigt, wie ein entsprechender Hashindex zusammen mit einer passenden Hashfunktion h_φ konstruiert werden kann.

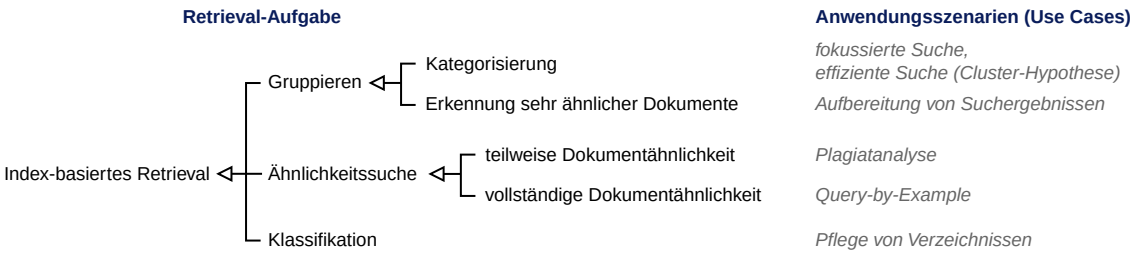


Abbildung 1: Taxonomie von Aufgaben und Beispiele für Anwendungsszenarien im textbasierten Information-Retrieval, die sich durch hashing-basierte Indizierung verbessern lassen.

Wir haben drei Arten von Retrieval-Aufgaben identifiziert, bei denen hashing-basierte Indizierung Verbesserungspotenzial birgt: (*i*) Gruppierung, insbesondere die Dokumentkategorisierung und die Identifikation sehr ähnlicher Dokumente (near duplicate identification [Broder 2000]), (*i*) Ähnlichkeitssuche, wobei zwischen vollständigen und partiellen Vergleichsmethoden unterschieden werden sollte, und (*iii*) Klassifikation. Abbildung 1 organisiert diese Aufgaben und nennt Beispiele für Anwendungsszenarien; die folgenden Unterabschnitte erläutern das Potenzial der hashing-basierten Indizierung aus technischer Sicht.

2.1 Retrieval-Aufgabe: Gruppierung

Die Gruppierung von Dokumenten spielt eine wichtige Rolle bei der Benutzerinteraktion und bei Schnittstellen von Informationssystemen: Viele Retrieval-Aufgaben liefern eine große Ergebnismenge mit Dokumenten, die sortiert, visuell aufbereitet oder von Duplikaten befreit werden soll. Eine Sortierung oder eine visuelle Aufbereitung erfordern die Identifikation von geeigneten Kategorien, also die Lösung eines unüberwachten Klassifikationsproblems. Kategorisierende Suchmaschinen wie Vivísimo und AIsearch lösen diese Aufgabe mittels einer Cluster-Analyse [Zamir und Etzioni 1998; Meyer zu Eißen und Stein 2002]. Die Erkennung von annähernd identischen Dokumenten ist nützlich u. a. bei der Produktsuche im World Wide Web oder zur Eliminierung von Dokumenten, die – bedingt durch die Spiegelung von Web-Seiten – mehrfach in einer Ergebnismenge auftauchen.

Anmerkungen zur Laufzeit. Durch hashing-basierte Indizierung lässt sich die Laufzeit solcher Retrieval-Aufgaben deutlich verkleinern. Ein Informationsbedarf wird hier als Wortanfrage formuliert, für die zunächst mit einer invertierten Liste μ_i die Ergebnismenge $D' \subseteq D$ ermittelt wird. Anschließend kommt ein Hashindex μ_h zur Gruppierung zum Einsatz. Abbildung 2 illustriert die Strategie. Der Hashindex μ_h erlaubt die Gruppierung der Dokumente in Linearzeit in der Größe der Ergebnismenge $|D'|$. Die Verwendung eines vektorbasierten Dokumentmodells hätte eine Laufzeit von $O(|D'|^2)$ zur Folge, da Duplikaterkennung oder Kategorisierung eine paarweise Ähnlichkeitsberechnung zwischen allen Elementen in D' erfordert.

2.2 Retrieval-Aufgabe: Ähnlichkeitssuche

Die am weitesten verbreitete Retrieval-Aufgabe ist diejenige Ähnlichkeitssuche, bei der Anwender ihren Informationsbedarf als Wortanfrage formulieren. Falls alle zur Wortanfrage passenden Dokumente zu ermitteln sind, ist eine invertierte Liste μ_i der optimale Index für die zu durchsuchende Kollektion. Bekannte Suchmaschinen wie Google, Yahoo oder AltaVista sind darauf spezialisiert, diese Art

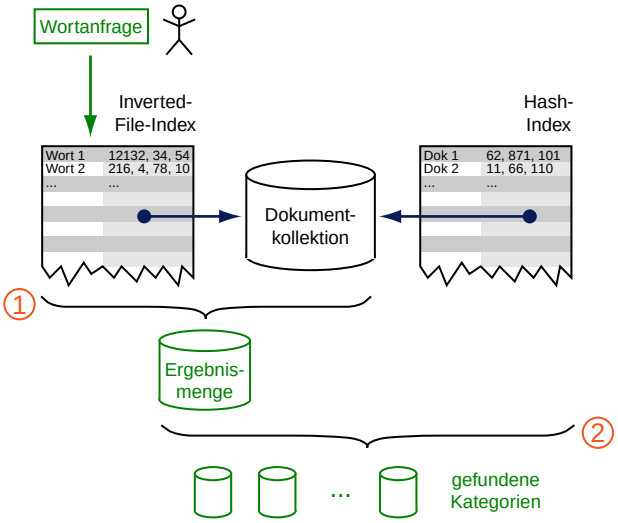


Abbildung 2: Illustration der Retrieval-Aufgabe Gruppierung. Ausgangspunkt ist ein als Wortanfrage formulierter Informationsbedarf. Mit einer invertierten Liste wird die Ergebnismenge derjenigen Dokumente ermittelt, die zu der Wortanfrage passen (Schritt ①). Anschließend wird mit einem Hashindex die Ergebnismenge in Linearzeit kategorisiert (Schritt ②).

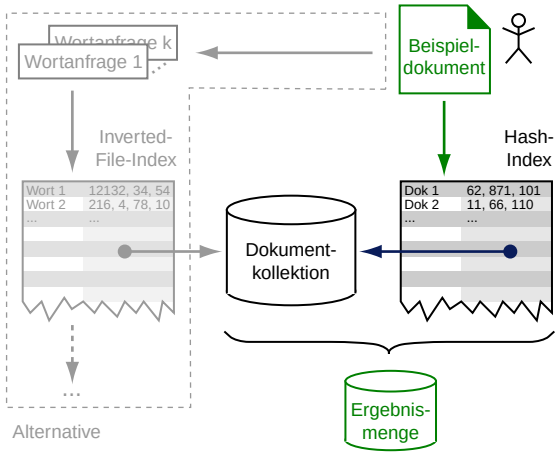


Abbildung 3: Illustration der Retrieval-Aufgabe Ähnlichkeitssuche. Ausgangspunkt ist ein in Form eines Beispieldokuments formulierter Informationsbedarf, der sich mit einem Hashindexes in konstanter Zeit beantworten lässt. Ein alternativer Ansatz (links angedeutet) erfordert die Extraktion von Schlüsselworten aus dem Beispieldokument sowie die Konstruktion geeigneter Wortanfragen.

von Anfragen extrem effizient zu bedienen.

Beschreibt ein Anwender seinen Informationsbedarf mit einem Beispieldokument („Suche ähnliche Dokumente“), kann ein hashing-basierter Index μ_h zum Einsatz kommen. Voraussetzung hierfür ist, dass eine Hashfunktion mit der Eigenschaft (1) konstruiert werden kann. Dies wiederum hängt von dem akzeptierten ε-Intervall des Anwenders für seinen speziellen Informationsbedarf ab.

Anmerkungen zur Laufzeit. Verglichen mit einer invertierten Liste μ_i ist mit einem hashing-basierten Index μ_h die Retrieval-Aufgabe um Größenordnungen schneller lösbar. Um die zu einem Beispieldokument ähnlichen Dokumente unter Verwendung von μ_i zu finden, wären zunächst geeignete Schlüsselworte aus dem Beispieldokument zu extrahieren, eine Reihe von k Wortanfragen zu formulieren und die Ergebnismengen $D'_1, \ldots, D'_k$ mit dem Beispieldokument zu vergleichen. Abbildung 3 (links angedeutet) illustriert diese Strategie; auf der rechten Seite ist die Strategie unter Verwendung von μ_h gezeigt. Unter der Annahme, dass die Zugriffszeit für beide Indizierungskonzepte $O(1)$ beträgt, benötigt die Konstruktion der Ergebnismenge bei Verwendung von μ_i eine Laufzeit von $O(|D'_1|+\ldots+|D'_k|)$, bei Verwendung von μ_h aber nur eine Laufzeit von $O(1)$.

Der tatsächliche erzielte Laufzeitunterschied hängt von der Qualität der extrahierten Schlüsselworte ab, sowie dem Geschick, hieraus Wortanfragen zu formulieren. Die Praxis zeigt, dass beträchtliche Verbesserungen zu erwarten sind. Dieses Ergebnis wird bei Retrieval-Aufgaben wie der Plagiatanalyse weiterhin verbessert: Hier ist die Segmentierung des Eingabedokuments – und damit eine Vervielfachung der Anfragen – notwendig, da eine Ähnlichkeitssuche für jeden einzelnen Abschnitt zu erfolgen hat [Stein und Meyer zu Eißen 2006].

2.3 Retrieval-Aufgabe: Klassifikation

Klassifikation spielt eine wichtige Rolle bei vielen textbasierten Retrieval-Aufgaben wie der Genre-Analyse, dem Filtern von Spam-Mails, der Kategoriezuordnung oder der Authoridentifikation. Es handelt sich hierbei um überwachte Klassifikationsaufgaben, also dem Pendant zur unüberwachten Kategorie*bildung*; sie lässt sich – eine kleine Anzahl von Klassen vorausgesetzt – erfolgreich mit Bayes, Diskriminanzanalyse, Support-Vector-Machines oder Neuronalen Netzen lösen. Bei einer großen Anzahl Klassen ist die Konstruktion eines Klassifikators mit garantierten statistischen Eigenschaften nahezu unmöglich.

Mit hashing-basierter Indizierung kann für eine Menge von Klassen $C_1, \ldots, C_p$ ein robuster Klassifikator konstruiert werden, selbst wenn p groß ist oder wenn nur eine kleine oder unregelmäßig verteilte Menge von Trainingsdokumenten vorliegt. Der Klassifikationsansatz folgt dem Prinzip der Nächsten-Nachbar-Suche und geht davon aus, dass die Trainingsdokumente der Klassen C_i in einem Hashindex μ_h indiziert sind. Für ein neu zu klassifizierendes Dokument d' wird der Hashwert berechnet und die Menge D' derjenigen Dokumente ermittelt, die demselben Hash-Bucket wie d' zugeordnet sind. Diese Dokumente werden bezüglich ihrer Verteilung in $C_1, \ldots, C_p$ untersucht und d' wird derjenigen Klasse C_j zugeordnet, der die meisten der Dokumente aus D' angehören:

$$C_j = \underset{i=1,\ldots,p}{\operatorname{argmax}} |C_i \cap D'|$$

3 Hashing-basierte Indizierungsverfahren

Ein hashing-basierter Index μ_h kann auf Basis einer Hashfunktion $h_\varphi : \mathbf{D} \rightarrow U$ direkt mit einer Hashtabelle $\mathcal{T}$ und einer Standardhashfunktion $h : U \rightarrow \{1, \ldots, |\mathcal{T}|\}$ konstruiert werden. Dabei bildet h das Universum U von Hashwerten gleichmäßig auf die $|\mathcal{T}|$ Speicherstellen der Hashtabelle ab.

Um eine Menge $\mathbf{D}$ von Dokumentrepräsentationen zu indizieren, wird der Hashwert $h_\varphi(\mathbf{d})$ aller Dokumente $\mathbf{d} \in \mathbf{D}$ berechnet und in $\mathcal{T}$ an Speicherstelle $h(h_\varphi(\mathbf{d}))$ eine Referenz auf d gespeichert. $\mathcal{T}$ enthält also für jeden Hashwert von h_φ einen Hash-Bucket $D' \subset D$ mit der folgenden Eigenschaft:

$$d_1, d_2 \in D' \;\Rightarrow\; h_\varphi(\mathbf{d_1}) = h_\varphi(\mathbf{d_2}),$$

wobei $\mathbf{d_1}, \mathbf{d_2}$ die Dokumentrepräsentationen der Dokumente d_1, d_2 bezeichnen. Mit $\mathcal{T}$ und h_φ kann eine als Beispieldokument formulierte Anfrage durch einmaliges Nachschlagen in der Hashtabelle in $O(1)$ beantwortet werden. Erfüllt h_φ weiterhin die Eigenschaft (1), so entspricht der für $\mathbf{d}$ ermittelte Hash-Bucket der Menge $\mathbf{D}'$ von Dokumenten, die sich bezüglich φ in der ε-Umgebung von $\mathbf{d}$ befinden:

$$\mathbf{d}' \in \mathbf{D}' \;\Rightarrow\; \varphi(\mathbf{d}, \mathbf{d}') \geq 1 - \varepsilon$$

Die wesentliche Herausforderung besteht in der Wahl und der Parametrisierung einer geeigneten Ähnlichkeitshashfunktion h_φ für Textdokumente. Zwei kürzlich vorgeschlagene Ansätze sind für diese Aufgabe geeignet: Fuzzy-Fingerprinting und Locality-Sensitive-Hashing.

3.1 Fuzzy-Fingerprinting

Bei Fuzzy-Fingerprinting handelt es sich um einen Hashing-Ansatz, der speziell für das textbasierte Information-Retrieval entwickelt wurde – jedoch nicht darauf beschränkt ist [Stein 2005]. Es basiert auf der Definition einer kleinen Anzahl k, $k \in [10, 40]$, von Präfixäquivalenzklassen. Eine Präfixäquivalenzklasse enthält alle Worte, die mit demselben Präfix beginnen. Die Berechnung von $h_\varphi(\mathbf{d})$ geschieht in den folgenden Schritten: (*i*) Bestimmung von $\mathbf{pf}$, einem k-dimensionalen Vektor, der die Verteilung der Indexterme in $\mathbf{d}$ auf die k Präfixäquivalenzklassen quantifiziert. (*ii*) Normalisierung von $\mathbf{pf}$ auf Basis eines repräsentativen Korpus und Berechnung von $\Delta_{\mathbf{pf}} = (\delta_1, \ldots, \delta_k)^T$, dem Vektor der Abweichungen von der erwarteten Verteilung.[1] (*iii*) Fuzzifizierung von $\Delta_{\mathbf{pf}}$ durch die Projektion der exakten Abweichungen unter Verwendung verschiedener Fuzzifizierungsschemata. Abbildung 4 illustriert die Berechnungsvorschrift.

In der Praxis kommen zwei bis drei Fuzzifizierungsschemata zum Einsatz, wobei jedes Schema (= linguistische Variable) bis zu vier Intervalle umfasst. Gleichung 2 definiert, wie sich aus dem normalisierten Abweichungsvektor $\Delta_{\mathbf{pf}}$ eines Dokuments unter Verwendung eines Fuzzifizierungsschemas ρ, das r Intervalle besitzt, ein Hashwert berechnet:

$$h_\varphi^{(\rho)}(\mathbf{d}) = \sum_{i=0}^{k-1} \delta_i^{(\rho)} \cdot r^i, \quad \text{mit } \delta_i^{(\rho)} \in \{0, \ldots, r-1\} \quad (2)$$

$\delta_i^{(\rho)}$ ist ein dokumentspezifischer Wert und beschreibt die fuzzifizierte Abweichung von $\delta_i \in \Delta_{\mathbf{pf}}$ vom Erwartungswert unter Verwendung des Fuzzifizierungsschemas ρ.

[1] Als Referenzkorpus dient uns der British National Corpus. Er enthält ca. 100 Millionen Worte und repräsentiert einen aktuellen Querschnitt der geschriebenen und gesprochenen englischen Sprache [Aston und Burnard 1998].

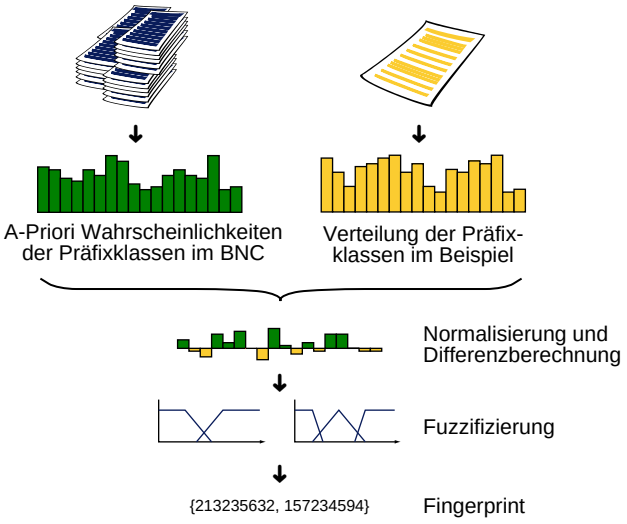


Abbildung 4: Die Berechnung eines Hashwerts durch Fuzzy-Fingerprinting.

Dokumentrepräsentationen auf Basis von Präfixäquivalenzklassen lassen sich als Abstraktionen des Vektorraummodells auffassen. Einerseits schneidet ein auf diese Art abstrahiertes Dokumentmodell bei den Retrieval-Aufgaben Gruppierung, Ähnlichkeitssuche oder Klassifikation tendenziell schlechter ab als das Vektorraummodell; andererseits sind die entsprechenden Vektoren um Größenordnungen kleiner und nicht dünn besetzt.

3.2 Locality-Sensitive-Hashing

Locality-Sensitive-Hashing (LSH) stellt einen allgemeinen Rahmen für die Konstruktion von Hashfunktionen dar [Indyk und Motwani 1998]. Eine lokalitätssensitive Hashfunktion h_φ ist eine Kombination von k einfachen Hashfunktionen h, $h : \mathbf{D} \rightarrow U$, die zufällig und unabhängig voneinander aus einer Familie H_φ von Hashfunktionen gezogen sind. Wählt man die Addition als Verknüpfungsoperator, so berechnet sich der Hashwert $h_\varphi(\mathbf{d})$ durch Addition der Hashwerte der einfachen Hashfunktionen:

$$h_\varphi(\mathbf{d}) = \sum_{i=1}^{k} h_i(\mathbf{d}), \qquad \text{mit } \{h_1, \ldots, h_k\} \subset_{rand} H_\varphi$$

In der letzten Zeit sind verschiedene Familien H_φ von Hashfunktionen entwickelt worden, die sich im textbasierten Information-Retrieval anwenden lassen [Charikar 2002; Datar *et al.* 2004; Bawa *et al.* 2005]; hier konzentrieren wir uns auf den Ansatz von Datar *et al.* Die Idee dieser Hashfamilie ist, eine Dokumentrepräsentation $\mathbf{d}$ durch Berechnung des Skalarprodukts $\mathbf{a}^T \cdot \mathbf{d}$ auf eine reelle Zahl abzubilden. $\mathbf{a}$ ist ein Zufallsvektor, dessen Vektorkomponenten unabhängig voneinander aus einer bestimmten Wahrscheinlichkeitsverteilung gezogen sind. Der reelle Zahlenstrahl wird in äquidistante Intervalle der Breite r unterteilt und jedem Intervall eine eindeutige natürliche Zahl zugewiesen. Das Skalarprodukt wird mit der Zahl desjenigen Intervalls assoziiert, in das der berechnete Wert des Skalarprodukts fällt. Die Berechnung von h_φ für k Zufallsvektoren $\mathbf{a}_1, \ldots, \mathbf{a}_k$ geschieht wie folgt:

$$h_\varphi^{(\rho)}(\mathbf{d}) = \sum_{i=1}^{k} \left\lfloor \frac{\mathbf{a}_i^T \cdot \mathbf{d} + c}{r} \right\rfloor$$

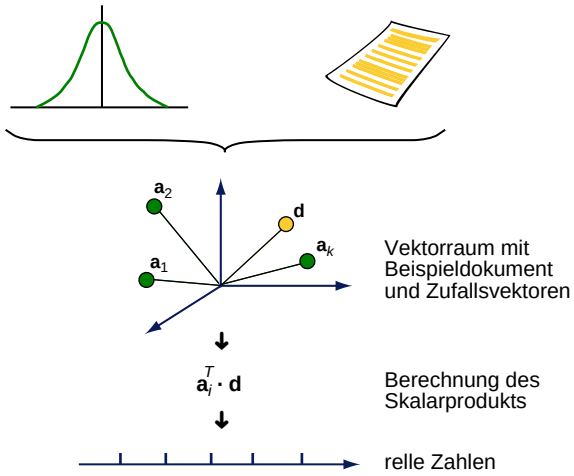


Abbildung 5: Die Berechnung eines Hashwerts durch Locality-Sensitive-Hashing.

$c \in [0, r]$ wird zufällig gewählt, um alle Segmentierungen des reellen Zahlenstrahls zu ermöglichen. Abbildung 5 illustriert die Berechnungsvorschrift.

Eine besondere Eigenschaft von Locality-Sensitive-Hashing ist, dass eine untere Schranke für die Retrieval-Qualität angegeben werden kann: Ist die durchschnittliche Distanz eines Dokuments zu seinem nächsten Nachbarn a-Priori bekannt, lässt sich h_φ so parametrisieren, dass die Wahrscheinlichkeit, den nächsten Nachbarn zu finden, oberhalb eines bestimmten Grenzwerts liegt [Gionis *et al.* 1999]. Diese Eigenschaft folgt aus der lokalen Sensitivität der zugrundeliegenden Hashfamilie H_φ und schreibt vor, dass für alle $h \in H_\varphi$ die Wahrscheinlichkeit einer Kollision der Hashwerte zweier Dokumente mit deren Ähnlichkeit steigt.[2]

3.3 Retrieval-Eigenschaften von Ähnlichkeitshashfunktionen

Auffälligstes Merkmal hashing-basierter Indizierungsverfahren ist die Abstraktion eines feingranularen Ähnlichkeitskonzeptes – quantifiziert durch eine Ähnlichkeitsfunktion φ – auf das binäre Konzept „ähnlich oder nicht ähnlich“: Zwei Dokumentrepräsentationen werden als ähnlich angesehen, wenn ihre Hashwerte gleich sind; andernfalls wird angenommen, dass sie nicht ähnlich sind. Diese als Eigenschaft (1) formalisierte Implikation steht in direkter Beziehung zu dem statistischen Konzept der *Precision*. Die Umkehrung der Implikation steht in direkter Beziehung zu dem statistischen Konzept des *Recall*: Ist die Ähnlichkeit zweier Dokumentrepräsentationen größer als ein bestimmter Schwellwert $1 - \varepsilon$, so wird angenommen, dass ihre Hashwerte gleich sind.

Es ist zu bemerken, dass Letzteres für eine Hashfunktion h_φ nicht in der Allgemeinheit gelten kann. h_φ berechnet für jede Dokumentrepräsentation genau einen Hashwert und definiert dadurch eine absolute Partitionierung des Raums der Dokumentrepräsentationen.[3] Zwangsläufig muss der

[2] Die Hashfamilie von Datar *et al.* ist lokalitätssensitiv, wenn die eingesetzte Wahrscheinlichkeitsverteilung α-stabil ist; das bekannteste Beispiel einer solchen Verteilung ist die Gauss-Verteilung. Grundlagen hierzu und weitere Details sind in Indyk [2000] und Nolan [2005] beschrieben.

[3] Im Gegensatz dazu definiert der vollständige Ähnlichkeitsgraph einer Menge $\mathbf{D}$ von Dokumentrepräsentationen für jedes Element in $\mathbf{D}$ eine spezifische Partitionierung.

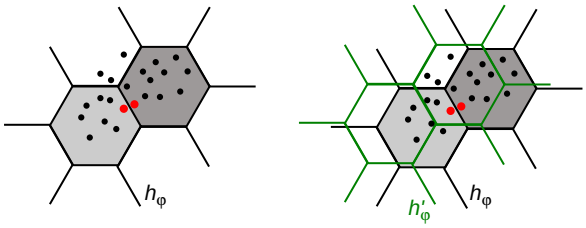


Abbildung 6: Abbildung einer Menge von Dokumentrepräsentationen in die Ebene. Eine Hashfunktion h_φ unterteilt die Ebene in Regionen, wobei jede Region durch genau einem Hashwert charakterisiert ist. Auch zwei sehr ähnliche Dokumentrepräsentationen (rot markiert) können auf verschiedene Hashwerte abgebildet sein (links dargestellt). Diese Schwellwertcharakteristik lässt sich durch die Verwendung mehrerer Hashfunktionen h_φ und h'_φ abmildern (rechts dargestellt).

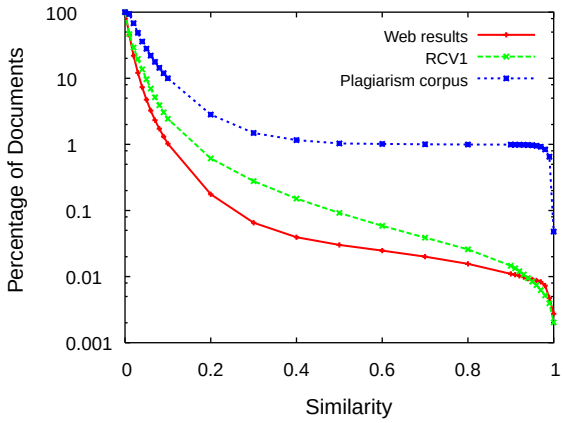


Abbildung 7: Das Diagramm zeigt das Verhältnis der Dokumente, deren paarweise Ähnlichkeit über einem bestimmten Schwellwert liegt.

durchschnittliche Recall zu einer Suchanfrage kleiner als 1 sein. Abbildung 6 illustriert diesen Zusammenhang: Trotz ihrer hohen Ähnlichkeit (= geringe Distanz) bildet die Hashfunktion h_φ einige der Dokumentrepräsentationen auf verschiedene Hashwerte ab. Wird zusätzlich eine zweite Hashfunktion h'_φ verwendet, die den Raum leicht unterschiedlich partitioniert, kann eine Suchanfrage durch die disjunktive Verknüpfung der beiden Hashfunktionen beantwortet werden. In der Praxis entspricht das der Konstruktion von zwei Hashindizes μ_h, μ'_h und der Rückgabe der Vereinigungsmenge beider Ergebnismengen als Gesamtergebnismenge einer Suchanfrage. Tatsächlich lässt sich ein monotoner Zusammenhang zwischen der Anzahl der Hashfunktionen und dem erzielten Recall beobachten. Ein so verbesserter Recall geht auf Kosten der Precision.

Es ist aufschlussreich, die verschiedenen Konzepte zu vergleichen, mit denen Varianz in die Hashwertberechnung eingebracht wird: Fuzzy-Fingerprinting verwendet hierfür unterschiedliche Fuzzifizierungsschemata ρ_i, Locality-Sensitive-Hashing verwendet hierfür unterschiedliche Mengen von Zufallsvektoren ρ_i. In beiden Fällen wird eine Dokumentrepräsentation $\mathbf{d}$ durch eine *Menge* von l einzelnen Hashwerten $\{h_\varphi^{(\rho_i)}(\mathbf{d}) \mid i = 1, \ldots, l\}$ kodiert. Diese Menge bezeichnen wir als Fingerabdruck.

4 Fuzzy-Fingerprinting versus LSH: Fallstudien

Dieses Kapitel präsentiert einige Ergebnisse umfassender Experimente, in denen die beiden Ansätze zur hashingbasierten Indizierung für die Retrieval-Aufgaben der Duplikateliminierung bzw. der Identifikation fast gleicher Dokumente (Abschnitt 2.1) und der Ähnlichkeitssuche (Abschnitt 2.2) eingesetzt wurden. Die Experimente zeigen die Alltagstauglichkeit dieser Technologie und erlauben Aussagen dahingehend, welcher der Ansätze besser für die jeweilige Retrieval-Aufgabe bzw. für das textbasierte Information-Retrieval im Allgemeinen geeignet ist.

4.1 Aufbau der Experimente

Die Experimente wurden auf Grundlage von drei Testkollektionen durchgeführt; zwei der Kollektionen enthalten jeweils 100.000 Dokumente, die dritte enthält 3.000 Dokumente.[4]

[4]Die Metadateien zur Beschreibung der Kollektionen stellen wir anderen Wissenschaftlern auf Anfrage zur Verfügung.

Die erste Kollektion (Web) wurde mit den Suchmaschinen Yahoo, Google und AltaVista erstellt und enthält die Ergebnisse einer fokussierten Suche. Hierzu wurde zunächst eine kleine Menge von Dokumenten über ein bestimmtes Thema ausgewählt und hieraus etwa 100 Schlüsselworte mittels einer Kookkurenzanalyse extrahiert (vgl. Matsuo und Ishizuka [2004]). Diese Vorgehensweise soll eine unverzerrte Auswahl von Schlüsselworten für ein Thema sicherstellen. Auf Basis der Schlüsselwortmenge wurden aus bis zu fünf Worten bestehende Anfragen generiert und den Suchmaschinen übergeben. Für jede Anfrage wurden die am höchsten eingestuften Suchergebnisse geladen und der Textinhalt extrahiert. Diese Kollektion dient zur Nachbildung von Ergebnismengen, wie sie von typischen Web-Retrieval-Systemen geliefert werden.

Die zweite Kollektion ist eine Auswahl von Dokumenten aus dem „Reuters Corpus Volume 1“ (RCV1), der von der Reuters Corporation für Forschungszwecke veröffentlicht wurde [Rose *et al.* 2002]. Der Korpus enthält mehr als 800.000 Dokumente, von denen jedes zwischen einigen hundert bis zu mehreren tausend Worten umfasst. Die Dokumente sind mit Metainformationen wie Kategorie, geographische Region oder Industriesektor angereichert. Insgesamt gibt es 103 verschiedene Kategorien, die hierarchisch unter den vier Hauptkategorien „Government, Social“, „Economics“, „Markets“ und „Corporate, Industrial“ einsortiert sind. Jede der Hauptkategorien ist die Wurzel eines Baums von Unterkategorien, so dass jede Unterkategorie die Informationen seiner Elternkategorie verfeinert. Diese Kollektion dient zur Nachbildung von Retrieval-Situationen in Unternehmen, die ihre Dokumente in vordefinierten Verzeichnishierarchien organisieren.

Die dritte Kollektion ist ein speziell angefertigter Korpus, um verschiedene Arten von Plagiatvergehen zu simulieren.[5] Die darin enthaltenen Dokumente wurden mit einem Algorithmus zur Synthese von Plagiatinstanzen erzeugt; Dokumente der ACM Digital Library bildeten die Eingabe. Diese Kollektion dient zur Nachbildung von Retrieval-Situationen, in denen es gilt, sehr ähnliche Dokumente zu entdecken.

[5]„Ein Plagiat ist die Vorlage fremden geistigen Eigentums bzw. eines fremden Werkes als eigenes Werk oder Teil eines eigenen Werkes“ Wikipedia [2006].

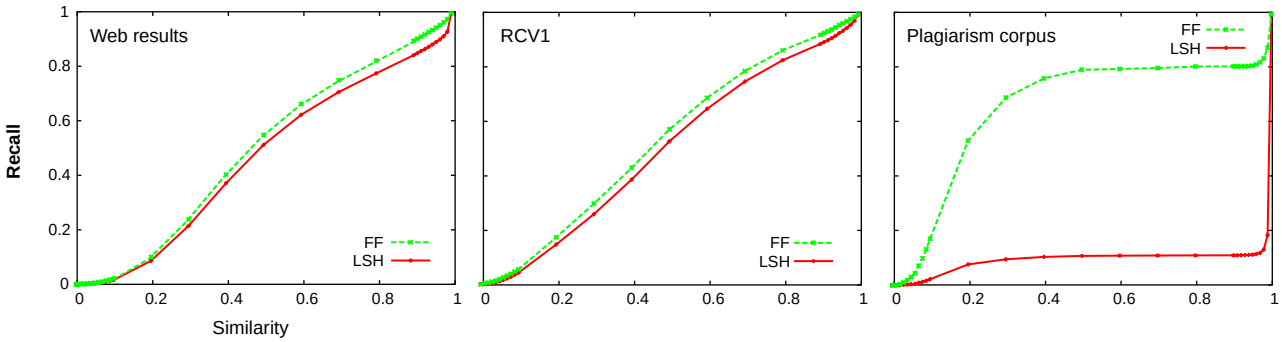


Abbildung 8: Der mit Fuzzy-Fingerprinting (FF) und Locality-Sensitive-Hashing (LSH) auf den drei Testkollektionen erzielte Recall in Abhängigkeit von der Ähnlichkeit.

4.2 Ergebnisse

Um die Retrieval-Performanz der Ähnlichkeitshashfunktionen zu messen, wurden für jede Testkollektion die Hashindizes gemäß Fuzzy-Fingerprinting und Locality-Sensitive-Hashing konstruiert. Für jeden Hashindex wurden für die Ähnlichkeitsschwellwerte $0.1 \cdot i,\ i \in \{0, \ldots, 10\}$, die durchschnittliche Precision und der durchschnittliche Recall ermittelt. Hierzu wurden Anfragen ausgewertet, bei denen jedes Dokument einer Kollektion als Beispieldokument diente. Die Referenzwerte für Precision und Recall basierten auf dem Vektorraummodell, dem $tf \cdot idf$-Schema und der Anwendung des Kosinusähnlichkeitsmaßes.

Die Analyse der drei Testkollektionen zeigt, dass der Anteil der Dokumente einer Kollektion, deren paarweise Ähnlichkeit über einer bestimmten Ähnlichkeitsschwelle liegt, exponentiell abnimmt. Abbildung 7 illustriert diesen Sachverhalt. Es ist zu beobachten, dass in den beiden großen Kollektionen der Prozentsatz sehr ähnlicher Dokumente klein ist, während im Plagiatkorpus ein deutlich größerer Anteil vorliegt.

Um dieser Verteilung Rechnung zu tragen und um die Ähnlichkeitshashfunktionen vergleichbar zu machen, wurden diese so parametrisiert, dass die durchschnittliche Anzahl zurückgegebener Dokumente pro Anfrage nicht zu groß und etwa gleich war. Hierfür wurde die Anzahl der verwendeten Fuzzifizierungsschemata bzw. Zufallsvektormengen angepasst (siehe Abschnitt 3.3): zwei bis drei Fuzzifizierungsschemata für Fuzzy-Fingerprinting, zwischen 10 und 20 Zufallsvektormengen für Locality-Sensitive-Hashing.

Abbildung 8 stellt den Recall beider Hashing-Ansätze in Abhängigkeit der Ähnlichkeitsschwellen für die drei Testkollektionen gegenüber. Bei hohen Ähnlichkeitsschwellen (> 0.8) ist der Recall bei beiden Hashing-Ansätzen ausgezeichnet. Ein hoher Recall für *niedrige* Ähnlichkeitsschwellen ist hingegen nur zufällig erreichbar, was an der linken und mittleren Kurve für die RCV1- und die Web-Kollektion zu beobachten ist. Dieses Verhalten lässt sich durch die Verteilung der Ähnlichkeiten in Abbildung 7 erklären. Beide Hashing-Ansätze verhalten sich ähnlich, wobei Fuzzy-Fingerprinting einen leicht besseren Recall erzielt. Bei dem Plagiatkorpus hingegen zeigt sich ein anderes Bild: Fuzzy-Fingerprinting übertrifft Locality-Sensitive-Hashing deutlich bei der Erkennung sehr ähnlicher Dokumente.

Abbildung 9 stellt die Precision beider Hashing-Ansätze in Abhängigkeit von den Ähnlichkeitsschwellen für die drei Testkollektionen gegenüber. Offensichtlich ist – unabhängig von der Testkollektion – die Precision von Fuzzy-Fingerprinting signifikant höher als die von Locality-Sensitive-Hashing. Das heißt, eine von Fuzzy-Fingerprinting für eine Anfrage zurückgegebene Ergebnismenge D' enthält entweder mehr relevante Dokumente oder sie ist kleiner als die von Locality-Sensitive-Hashing gelieferte Ergebnismenge. Beides hat direkten Einfluss auf die Suchzeit pro Anfrage und die Ergebnisqualität.

Die Laufzeit exakter Retrieval-Ansätze, die beispielsweise auf dem Vektorraummodell basieren, ist linear in der Größe der Dokumentkollektion. Im Gegensatz dazu kann die Laufzeit der Hashing-Ansätze als konstant betrachtet werden. Das Verhältnis der Precision-Kurven in Abbildung 9 gibt Aufschluss über das Verhältnis dieser Konstanten. Insbesondere ließ sich in den Experimenten beobachten, dass die durchschnittliche Größe $|D'|$ der Ergebnismenge pro Suchanfrage linear mit der Größe $|D|$ der Kollektion steigt, falls Art und Verteilung der Dokumente

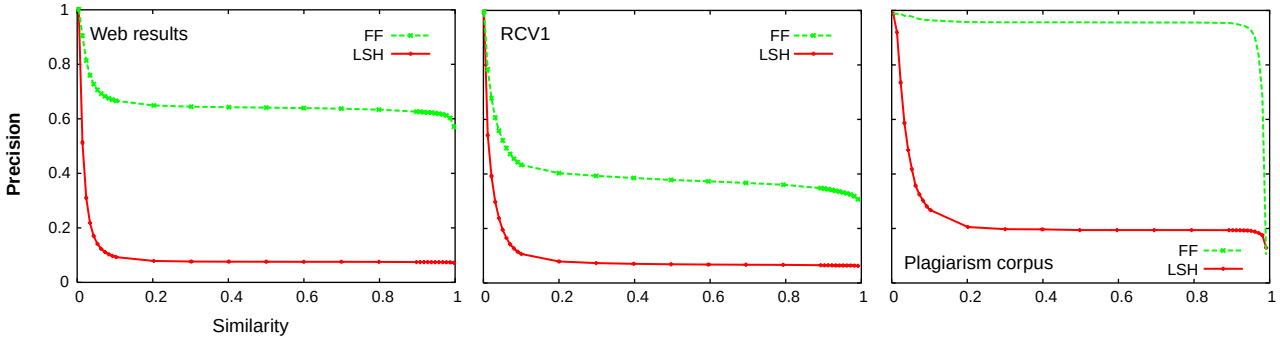


Abbildung 9: Die mit Fuzzy-Fingerprinting (FF) und Locality-Sensitive-Hashing (LSH) auf den drei Testkollektionen erzielte Precision in Abhängigkeit von der Ähnlichkeit.

der Kollektion sich nicht ändern.

Die Precision von Fuzzy-Fingerprinting wird durch die Anzahl k von Präfixäquivalenzklassen und die Anzahl r von Abweichungsintervallen je Fuzzifizierungsschemata gesteuert. Um die Precision zu erhöhen, genügt die Vergrößerung von einem der beiden Parameter. Der optimale Wert für k ist von der Retrieval-Aufgabe abhängig; typische Werte für r liegen zwischen zwei und vier. Die Precision von Locality-Sensitive-Hashing steigt mit der Anzahl k der verknüpften Hashfunktionen. Bei Verwendung der Hashfamilie von Datar *et al*. entspricht k der Anzahl der Zufallsvektoren je Hashfunktion; typische Werte für k liegen zwischen 20 und 100.

4.3 Diskussion

Die Ergebnisse der Experimente geben einen Überblick über das Verhalten hashing-basierter Indizierung bei der Identifikation fast gleicher Dokumente und bei der Ähnlichkeitssuche.

Fuzzy-Fingerprinting ist Locality-Sensitive-Hashing bei der Ähnlichkeitssuche im Hinblick auf den Recall nur leicht voraus. Wir erklären diesen Sachverhalt mit dem geringen Anteil von Dokumenten in den Kollektionen, deren paarweise Ähnlichkeit größer als 0.5 ist. Bezüglich der Precision ist Fuzzy-Fingerprinting dem Verfahren des Locality-Sensitive-Hashing überlegen. Dieser Sachverhalt spielt jedoch nur eine untergeordnete Rolle, da der Umfang der Ergebnismenge einer Anfrage – verglichen mit der Größe der Dokumentkollektion – im Normalfall um Größenordnungen kleiner ist. Das heißt, Locality-Sensitive-Hashing kann für die Ähnlichkeitssuche genauso gut verwendet werden wie Fuzzy-Fingerprinting.

Bei der Identifikation sehr ähnlicher Dokumente hingegen übertrifft Fuzzy-Fingerprinting das Verfahren des Locality-Sensitive-Hashing sowohl bezüglich Precision als auch Recall. Für diese Art von Retrieval-Aufgaben ist Fuzzy-Fingerprinting fast konkurrenzlos.

5 Zusammenfassung

Hashing-basierte Indizierung ist eine vielversprechende Technologie im textbasierten Information-Retrieval, die zuverlässige und effiziente Anfragen nach ähnlichen Dokumenten für verschiedene Retrieval-Aufgaben gestattet. Wir haben drei wichtige Aufgabenklassen identifiziert, in denen hashing-basierte Indizierung Verbesserungspotenzial bietet, nämlich Gruppierung, Ähnlichkeitssuche und Klassifikation.

Es wurden zwei Konstruktionsprinzipien für Hashfunktionen vorgestellt: Fuzzy-Fingerprinting und Locality-Sensitive-Hashing. Eine umfassende experimentelle Analyse beider Hashing-Ansätze wurde durchgeführt, um (*i*) ihre Einsetzbarkeit bei der Ähnlichkeitssuche und der Suche nach fast identischen Dokumenten zu demonstrieren, und (*ii*) sie bezüglich Precision und Recall miteinander zu vergleichen.

Die Ergebnisse zeigen, dass Fuzzy-Fingerprinting bei der Suche nach fast identischen Dokumenten dem Verfahren des Locality-Sensitive-Hashing bezüglich Precision und Recall überlegen ist. Bei der Ähnlichkeitssuche ist die Precision von Fuzzy-Fingerprinting deutlich höher als die von Locality-Sensitive-Hashing, wohingegen nur ein leicht höherer Recall beobachtet wurde. Unsere Analyse beschränkte sich auf das textbasierte Information-Retrieval; wir möchten aber betonen, dass die vorgestellten Konzepte und Algorithmen auch für Retrieval-Aufgaben aus anderen Domänen adaptierbar sind. Insbesondere Locality-Sensitive-Hashing wurde daraufhin ausgelegt, unterschiedliche Arten hochdimensionaler, vektorbasierter Objektrepräsentationen zu verarbeiten. Auch die Prinzipien hinter Fuzzy-Fingerprinting sind übertragbar.

Unsere aktuellen Forschungen beschäftigen sich damit, hashing-basierte Indizierung theoretisch zu analysieren und die dabei gewonnenen Erkenntnisse in speziellen Retrieval-Aufgaben umzusetzen. Ziel ist es, die Beziehung zwischen den Determinanten von Fuzzy-Fingerprinting und der Retrieval-Performanz zu quantifizieren, um optimierte Hashindizes konstruieren zu können.

Fuzzy-Fingerprinting wird als Schlüsseltechnologie in unseren Werkzeugen zur textbasierten Plagiatanalyse eingesetzt.

Literatur

Guy Aston und Lou Burnard. The BNC Handbook. `http://www.natcorp.ox.ac.uk/what/whatis.html`, 1998.

Ricardo Baeza-Yates und Berthier Ribeiro-Neto. *Modern Information Retrieval*. Addison Wesley, 1999.

Mayank Bawa, Tyson Condie und Prasanna Ganesan. Lsh forest: Self-tuning indexes for similarity search. In *WWW '05: Proceedings 14th international conference on World Wide Web*, S. 651-660, New York, NY, USA, 2005. ACM Press.

Andrei Z. Broder. Identifying and filtering near-duplicate documents. In *COM '00: Proceedings 11th Annual Symposium on Combinatorial Pattern Matching*, S. 1-10, London, 2000. Springer.

Moses S. Charikar. Similarity estimation techniques from rounding algorithms. In *STOC '02: Proceedings 34th annual ACM symposium on Theory of computing*, S. 380-388, New York, NY, USA, 2002. ACM Press.

Mayur Datar, Nicole Immorlica, Piotr Indyk und Vahab S. Mirrokni. Locality-sensitive hashing scheme based on p-stable distributions. In *SCG '04: Proceedings 20th annual symposium on Computational geometry*, S. 253-262, New York, NY, USA, 2004. ACM Press.

Aristides Gionis, Piotr Indyk und Rajeev Motwani. Similarity Search in High Dimensions via Hashing. In *Proceedings 25th VLDB Conference Edinburgh, Scotland*, 1999.

Piotr Indyk und Rajeev Motwani. Approximate Nearest Neighbor—Towards Removing the Curse of Dimensionality. In *Proceedings 30th Symposium on Theory of Computing*, S. 604-613, 1998.

P. Indyk. Stable distributions, pseudorandom generators, embeddings and data stream computation. In *FOCS '00: Proceedings 41st Annual Symposium on Foundations of Computer Science*, S. 189, Washington, DC, USA, 2000. IEEE Computer Society.

Y. Matsuo und M. Ishizuka. Keyword extraction from a single document using word co-ocurrence statistical information. *International Journal on Artificial Intelligence Tools*, 13(1):157-169, 2004.

Sven Meyer zu Eißen und Benno Stein. The AISEARCH Meta Search Engine Prototype. In Amit Basu und Soumitra Dutta, Eds., *Proceedings 12th Workshop on Information Technology and Systems (WITS 02), Barcelona*. Technische UniversitÃ?t Barcelona, Dezember 2002.

John P. Nolan. Stable distributions—models for heavy tailed data. http://academic2.american.edu/~jpnolan/stable/stable.html, 2005.

T.G. Rose, M. Stevenson und M. Whitehead. The Reuters Corpus Volume 1—From Yesterday's News to Tomorrow's Language Resources. In *Proceedings 3rd International Conference on Language Resources and Evaluation*, 2002.

Benno Stein und Sven Meyer zu Eißen. Near Similarity Search and Plagiarism Analysis. In M. Spiliopoulou, R. Kruse, C. Borgelt, A. Nürnberger und W. Gaul, Eds., *From Data and Information Analysis to Knowledge Engineering*, S. 430-437. Springer, 2006.

Benno Stein. Fuzzy-Fingerprints for Text-Based Information Retrieval. In Klaus Tochtermann und Hermann Maurer, Eds., *Proceedings 5th International Conference on Knowledge Management (I-KNOW 05), Graz*, Journal of Universal Computer Science, S. 572-579. Know-Center, July 2005.

Roger Weber, Hans-J. Schek und Stephen Blott. A Quantitative Analysis and Performance Study for Similarity-Search Methods in High-Dimensional Spaces. In *Proceedings 24th VLDB Conference New York, USA*, S. 194-205, 1998.

Wikipedia. Plagiarism. `http://de.wikipedia.org/wiki/Plagiat`, 2006.

Ian H. Witten, Alistair Moffat und Timothy C. Bell. *Managing gigabytes (2nd ed.): compressing and indexing documents and images*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1999.

Oren Zamir und Oren Etzioni. Web Document Clustering: A Feasibility Demonstration. In *Proceedings 21st annual international ACM SIGIR conference on Research and development in information retrieval*, S. 46-54, University of Washington, Seattle, USA, 1998.

Aspekte des Qualitätsmanagements bei der Implementierung einer Suchmaschine

Christoph Schindler, Dirk Burmeister
Deutsches Institut für Internationale Pädagogische Forschung
Informationszentrum Bildung
Frankfurt am Main
schindler@dipf.de, burmeister@dipf.de

Abstract

In diesem Aufsatz soll die geplante Implementierung von Suchmaschinentechnologien im Fachportal Pädagogik zum Anlass genommen werden, um sich mit den damit verbundenen neuen Anforderungen an ein Qualitätsmanagement auseinanderzusetzen. Im Zentrum stehen die Fragen, welche Zusammenhänge die Recherche-Situationen formen und welche Schlussfolgerungen sich daraus für ein Evaluationsdesign ergeben. Als analytisches Instrumentarium soll dabei eine soziotechnische Sichtweise auf das Information-Retrieval-System (IR) dienen.

1 Einleitung

Den Ausgangspunkt dieser Arbeit bildet die Überlegung, die Suche in der FIS Bildung Literaturdatenbank des Fachportals Pädagogik an gängige Suchmaschinen-Technologien anzupassen. Davon erhofft man sich eine Optimierung der Informationssuche durch eine zeitliche Verkürzung der Trefferanzeige sowie die Verbesserung der Treffergenauigkeit (Precision) und der Anzahl relevanter Treffer (Recall) mittels präziserer Steuerung und Verarbeitung. So soll u.a. die vorhandene Sortierung der Trefferliste nach Erscheinungsjahr durch ein Ranking ersetzt und die Suchanfrage durch Tokenization präzisiert werden.

Es wird erwartet, dass sich die Veränderungsprozesse in komplexer Weise auf die Nutzungs- und Produktionssituation des Information-Retrieval-Systems auswirken. Um diese besser antizipieren zu können, erscheint es sinnvoll, sich mit der Gestaltung der Recherche-Situation genauer auseinanderzusetzen. Dabei rücken die Fragen, welche Zusammenhänge die Recherche-Situation formen und welche Schlussfolgerungen für die Entwicklung eines Evaluationsdesigns daraus gezogen werden können, in den Mittelpunkt.

Die Formung des Systems wird als soziotechnischer Prozess verstanden, der in ein soziales Umfeld eingebettet ist, das nicht nur auf die Nutzungssituation beschränkt ist. In diesem Aufsatz werden daher die Produktionsprozesse des Systems in den Fokus gestellt. Diese Verschiebung der Perspektive vom Nutzer hin zu den Produzenten und deren soziotechnisches Umfeld ermöglichen den weiteren Einbezug von Qualitätsfaktoren auf den Produktionsprozess. Eine Evaluation eines Information-Retrieval-Systems kann sich dann mit impliziten gesellschaftlichen Vorannahmen und Voraussetzungen auseinandersetzen, die schon vor einer Nutzungsanalyse bestehen.

Die Analyse erfolgt am Fallbeispiel der FIS Bildung Literaturdatenbank, die als Verbund von ca. 30 Dokumentationseinrichtungen in Deutschland, der Schweiz und Österreich Literaturnachweise zu den Themenfeldern Bildungsforschung und Bildungspraxis sammelt. Die Koordinierungsstelle ist am Informationszentrum Bildung des Deutschen Instituts für Internationale Pädagogische Forschung angesiedelt. Der Anspruch dieser Einrichtung lautet, in Zusammenarbeit mit den kooperierenden Partnerinstitutionen alle bedeutsamen Sachgebiete des Bildungsbereichs im deutschsprachigen Raum abzudecken [Fachportal Pädagogik].

Im Folgenden sollen zunächst die Bedingungen der Evaluation im Allgemeinen, die Erfordernisse bei der Erforschung eines Recherchesystems und die Grundlagen für eine soziotechnische Analyse dargestellt werden. Anschließend werden am konkreten Beispiel der FIS Bildung Literaturdatenbank der Produktionsprozess aus soziotechnischer Perspektive betrachtet und sich daraus ergebende Schlussfolgerungen für das Evaluationsdesign diskutiert.

2 Konfiguration von Evaluation

Etymologisch lässt sich der Begriff Evaluation auf das lateinische Verb „valuere“, auf Deutsch „bewerten“, zurückführen. Dieser Bezug legt offen, dass es sich bei der Evaluation nicht um eine wertneutrale Form der Urteilsfindung handelt, sondern um eine, die abhängig ist von Wertvorstellungen und Normierungen. Sie orientiert sich an einem vorab gesetzten Vergleichsmaßstab, wobei sich zwangsläufig implizite Annahmen in das Evaluationsdesign einschreiben [Vgl. Ulrich 2003].

In der gängigen Vorstellung wird Evaluation als eine anwendungs- und zweckgerichtete Verfahrensweise mit dem Ziel der Optimierung und Entscheidungsfindung in komplexen Handlungssituationen angesehen. Im Unterschied zur naturwissenschaftlichen Forschung sollen mit Hilfe der Evaluation jedoch keine generellen Aussagen getroffen, sondern eine spezifische Situation soll beschrieben und bewertet werden. So erfolgt eine Evaluation im Rahmen des Qualitätsmanagements immer auch in Bezug auf einen konkreten, vorab umrissenen Kontext.

2.1 Evaluation von IR-Systemen

Der klassischen IR-Forschung liegt häufig eine naturwissenschaftliche Sichtweise von Forschung zu Grunde, die die Kriterien Objektivität, Validität und Reliabilität in das Zentrum der Verfahrensweise stellt [vgl. Tague-Sutcliffe 1992]. Als Maßstab für die Bewertung eines Informationssystems werden dabei die Effizienz, die sich nach den

System-Ressourcen Rechenzeit und Speicherplatz richtet, sowie die Effektivität, wie u.a. Trefferanzahl (Recall) und Treffergenauigkeit (Precision), angesehen.

Aktuelle Auseinandersetzungen mit der Evaluation von IR-Systemen konstatieren, dass diese Kriterien die alltäglichen Recherchesituationen nur ungenügend wiedergeben. So konstatiert Ferber, dass bei der Nutzung eines IR-Systems eine Komplexität zum Tragen kommt, die aus seiner Sicht weder theoretisch noch praktisch erfassbar ist: „Man müsste eine repräsentative Auswahl von Anwendungsproblemen und Benutzenden zur Verfügung haben und den Einfluss des IR-Systems auf die Lösung der Anwendungsprobleme isolieren und bewerten können“ [Ferber 2003: 83]. Die Folge ist, so Ferber, dass auf Grund der beschränkten Ressourcen zwangsläufig einige in der Praxis auftretende Einflussfaktoren nicht beachtet werden.

Lewandowski plädiert seinerseits für einen holistischen Ansatz bei der Bewertung von IR-Systemen [Lewandowski 2005: 15]. Er fordert, die Bedürfnisse der Nutzer einzubeziehen um der technikzentrierten Informatik eine ganzheitliche Perspektive entgegenzusetzen.

In der Diskussion um das Qualitätsmanagement von Informationsdienstleistungen gewinnt der Bezug auf die Formgebungsprozesse von Technologien immer mehr an Raum. So weist Rittberger darauf hin, dass beim Informationsqualitätsmanagement neben dem Endprodukt ebenso die internen Prozesse der Entwicklung sowie die Kompetenzen der Mitarbeiter in die Betrachtung aufgenommen werden sollten [vgl. Rittberger 2004]. Zudem gibt es im Qualitätsmanagement verstärkt Ansätze, der Komplexität in der Praxis Rechnung zu tragen, indem eine Vielzahl an Kriterien zur Beschreibung, Prüfung und Steuerung herangezogen werden [vgl. Kempa 2002, Rittberger, Rittberger 1997].

Einen weit verbreiteten Ansatz zur Entwicklung von Qualitäts- und Evaluationskriterien stellt in der organisatorischen Praxis die GAP-Analyse, bei der auf Basis von Kundenwünschen Qualitätsmerkmale ausgearbeitet werden. Evaluiert werden diese mittels unterschiedlicher empirischer Methoden, deren Ergebnisse in Diskrepanz zum Endprodukt gestellt werden [vgl. Kempa 2002].

Zusammenfassend lässt sich sagen, dass in der praktischen Auseinandersetzung mit Recherche-Situationen zunehmend holistische Betrachtungsweisen eingefordert werden. Neben der Problematisierung technikzentrierter Ansätze werden weitere Disziplinen einbezogen oder in Form von Kriterien in die Evaluation aufgenommen sowie teilweise Kundenbedürfnisse in die Kriterienbildung integriert.

Um den komplexen Zusammenhang zwischen Technik und Sozialität in der Recherche-Situation genauer zu betrachten, soll in den folgenden Abschnitten das Konzept des soziotechnischen Systems vorgestellt und dieses danach am Beispiel der FIS Bildung Literaturdatenbank konkretisiert werden.

2.2 Soziotechnische Systeme und soziotechnisches Engineering

Als soziotechnisches System wird ein Gefüge aus wechselseitigen Beziehungen zwischen Menschen, Praxen und Artefakten verstanden. Diese Sichtweise orientiert sich an der englischsprachigen Wissenschafts- und Technikforschung, die grob mit der Richtung Science and Technology Studies (STS) in sozialkonstruktivistischer Ausprägung bezeichnet werden kann.

Die STS wehren sich gegen die gängige Annahme Wissenschaft und Technik als von der Gesellschaft getrennte, wertfreie und autonome Kräfte zu denken. Stattdessen wird hier von einer sozialen Konstruktion sowie einer sozialen Formung („Social Shaping“) technischer und wissenschaftlicher Artefakte ausgegangen [vgl. Ilyes 2006].

Für die Anwendung unterschiedlicher Formen des Wissens beim Designprozess von technischen Systemen prägte der Technikforscher MacKenzie den Begriff „heterogene Technikentwicklung“ („heterogenous Engineering“). Er vertritt die Ansicht, dass ein erfolgreiches Engineering stets auch als „Engineering des Sozialen“ zu sehen ist. Demnach ist die Technikentwicklung dadurch gekennzeichnet, dass Ingenieure nicht nur mit Materialien und Zahlen verfahren, sondern auch soziale, wirtschaftliche und politische Auswirkungen einbeziehen sowie durch ihre Produkte zu deren Veränderungen beitragen [MacKenzie 1987, 198].

Callon führt aus, dass Ingenieure während der Technikentwicklung zwangsläufig soziologisches Wissen verarbeiten, unabhängig davon, ob sie dies intendieren oder nicht. Dies geschieht unumgänglich, da sie einerseits als Akteure innerhalb von sozialen Beziehungen interagieren, andererseits, da sie mit dem Anwendungshintergrund neuer Technologien immer auch ein Modell von Gesellschaft mit entwerfen und in die Systeme mit einschreiben [Callon 1987].

Daher soll in dieser Arbeit gerade auf ein „soziotechnisches Engineering“ Wert gelegt werden, indem Aspekte der sozialen Formung eines IR-Systems exemplarisch an den Produktionsprozessen der FIS Bildung Literaturdatenbank aufgezeigt werden sollen. Dabei sollen im Folgenden die wechselseitigen Beziehungen zwischen unterschiedlichen Professionen, Arbeits- und Nutzungspraxen, Standardisierungen, Routinen und Dokumenten in den Fokus rücken, die die Recherche in der FIS Bildung Literaturdatenbank formen.

3 Die FIS Bildung Recherche-Situation aus soziotechnischer Perspektive

Der sichtbare Handlungsablauf einer Recherche-Situation ist in seiner groben Abfolge schnell dargestellt. Ein Nutzer – zum Beispiel ein erziehungswissenschaftlicher Forscher, Student oder Pädagoge – gibt in das Online-Formular der FIS Bildung Literaturdatenbank beim Fachportal Pädagogik eine Anfrage ein, die nach seiner Einschätzung relevante Treffer für sein Informationsbedürfnis hinsichtlich bildungsrelevanter Themen liefert. Der zur Verfügung stehende Suchraum wurde zuvor durch die FIS Bildung in einem kooperativen Verbund erstellt. Die Dokumente wurden dabei nach Bildungsrelevanz und bibliographischen Kriterien ausgewählt, formal und inhaltlich beschrieben sowie anhand von Repräsentationen in einer Datenbank zur Verfügung gestellt. Eine schematische Darstellung dieses Prozesses bietet die folgende Abbildung:

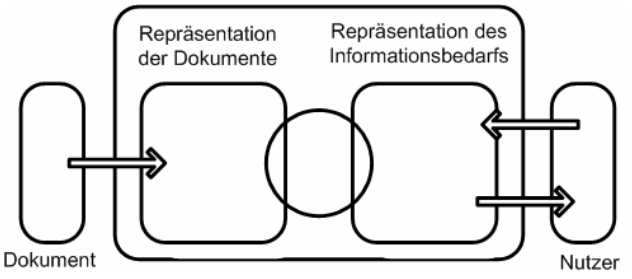


Abbildung 1: Schematische Darstellung der Recherche-Situation [vgl. Ferber 2003: 25]

3.1 Formungsprozesse in der Produktion

Die Informationsinfrastruktur der FIS Bildung Literaturdatenbank wird hauptsächlich von ca. 30 unterschiedlichen Dokumentationseinrichtungen erstellt. Dies geschieht kooperativ, eingebettet in die dokumentarischen Wissensarbeitspraxen der einzelnen Institute[1]. In den Instituten kümmert sich geschultes Fachpersonal um die dokumentarische Praxis, die das Überwachen von Publikationslandschaften, die Auswahl von Dokumenten, deren formale und inhaltliche Analyse sowie ihre Beschreibung [vgl. Rittberger, Rittberger 1997] und Eingabe in die FIS Bildung Literaturdatenbank umfasst.

Setzt man bei der Anzahl der möglichen Publikationsquellen und deren Überwachung durch bibliothekarisches Fachpersonal an, wird bereits im ersten Schritt der dokumentarischen Praxis durch die einzelnen Institute eine Auswahl getroffen. Sortiert wird nach Verlagen, Zeitschriften, Instituten, Projekten, Dokumentenservern etc., die es zu beobachten lohnt, da dort potenziell relevante Dokumente erscheinen können. Hervorzuheben ist, dass die dokumentarische Auswahl und damit die Relevanz der Dokumente für die Wissens- und Forschungspraxis von den vorhandenen Ressourcen und dem Auftrag der durchführenden Institution abhängig sind.

In einem weiteren Schritt werden anschließend aus dem Pool der möglicherweise relevanten Dokumente diejenigen ausgewählt, die formal und inhaltlich erschlossen werden sollen. Die notwendigen Beschreibungen der bildungsrelevanten Dokumente werden dabei nicht nur im System der jeweiligen Partnerinstitutionen gespeichert, sondern zusätzlich an die FIS Bildung Literaturdatenbank weitergeleitet.

Die Auswahl der Literaturnachweise, die dann tatsächlich in der FIS Bildung Datenbank erscheinen, richtet sich nach Kriterien, die in der FIS Bildung Policy festgelegt und auf deren Webseite veröffentlicht sind [Fachportal Pädagogik]. So muss das zu erschließende Dokument überwiegend oder teilweise einen pädagogischen Bezug aufweisen. Dabei sollte sich zumindest ein thematischer Aspekt den Themenfeldern zuordnen lassen, die sich an der Sektionenstruktur der Deutschen Gesellschaft für Erziehungswissenschaft (DGfE) orientieren. Damit erfolgt die inhaltliche Auswahl von relevanten Dokumenten stark in Ausrichtung an der vorgegebenen wissenschaftlich-disziplinären Landschaft der DGfE. Neben den inhaltlichen werden ebenso formale und qualitative Auswahlkriterien angewandt, wie u.a. inhaltliche Vollständigkeit, formale Korrektheit, Authentizität, Gutachter und statische Struktur. Diese Kriterien orientieren sich stark an der wissenschaftlichen Publikationspraxis. Zudem werden nur bestimmte Dokumenttypen zugelassen.

Der Suchraum in der Recherche-Situation ist neben der Auswahl und der Bestimmung der Kriterien ebenso abhängig von der Erschließung und Beschreibung der Dokumente, die in Form der Metadaten gespeichert werden. In der FIS Bildung Literaturdatenbank setzt dies eine bibliographisch-formale und inhaltliche Analyse des Dokumentes voraus. Somit werden neben den klassischen bibliographisch-formalen Angaben wie Titel, Autor, Publikationsjahr etc. Verschlagwortungen mit kontrollierter Schlagwortliste und automatischem Synonymabgleich durchgeführt sowie größtenteils Kurzreferate in Form von Abstracts erstellt.

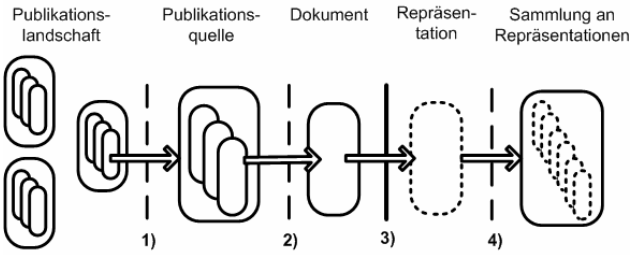


Abbildung 2: Schematische Darstellung des Produktionsprozesses

Festzuhalten ist, dass der Suchraum der Informationsinfrastruktur bei der Recherche-Situation durch die oben aufgeführten Wissenspraktiken hergestellt und geformt wird. Dabei finden mehrere Grenzziehungen durch Auswahlprozesse statt (siehe Abbildung 2), die den Dokumenten jeweils beim Überschreiten einen neuen Status und neue Bedeutung zuweisen. Erstens werden innerhalb der deutschsprachigen Publikationslandschaft Quellen ausgewählt, die für die Dokumentationseinrichtungen relevant sind. Unter den relevanten Publikationsquellen erfolgt zweitens eine Auslese der relevanten Dokumente. Diese werden drittens durch Dokumentationseinrichtungen erschlossen bzw. mit Repräsentationen versehen, welche potenzielle Kandidaten für die FIS Bildung Literaturdatenbank sind. Viertens werden die relevanten Repräsentationen der Dokumente in die Sammlung aufgenommen, die den Anspruch hat, die deutschsprachige Literatur für Bildungsforschung und -praxis zu repräsentieren.

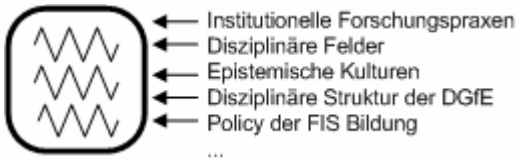


Abbildung 3: Schematische Darstellung der Formung der FIS Bildung Literaturdatenbank

Diese Aufzählung macht deutlich, dass der Suchraum sowohl von den Wissenspraxen der Dokumentationseinrichtungen abhängig ist als auch von der Kriterienbildung und –anwendung der FIS Bildung, die inhaltlich auf die wissenschaftlich-disziplinäre Struktur der DGfE ausgerichtet ist. Somit schreiben sich Änderungen in den Forschungs- und Dokumentationspraxen der kooperierenden Institutionen sowie in der wissenschaftlichen Strukturierung der DGfE in die Auswahl der Dokumente bzw. in die

[1] Zusätzlich zu den Erschließungen der Dokumentationseinrichtungen werden auch themenbezogene Recherchen durchgeführt. Zudem besteht die Möglichkeit Publikationen über ein webbasiertes Formular zu melden.

Gestaltung der FIS Bildung Literaturdatenbank ein (siehe Abbildung 3).

Anzumerken ist in diesen Zusammenhang, dass in der Forschungspraxis auch Spannungen zwischen Institutionen, disziplinären Feldern [Fry 2006] und „epistemic cultures“, die Wissen kreieren und gewährleisten [Knorr-Cetina 1999], auftreten können. Da sich jedoch bei den Produktionsprozessen von Dokumenten, die ebenso mit in die Recherche-Situation einzubeziehen sind [vgl. Belkin et al. 1982a,b], sowie bei der Erstellung der Repräsentation der Dokumente sowohl disziplinäre Felder als auch die Form der Erkenntnisgewinnung mit einschreiben [vgl. Akrich 1989], werden diese Grenzziehungen ebenfalls mit in die Sammlung übernommen.

Es stellt sich die Frage, wie sich die Implementierung neuer Suchmaschinentechnologien auf das komplexe Produktionsgefüge der FIS Literaturdatenbank auswirkt. Eine Neuordnung der Suche würde etwa die Trefferreihenfolge der Literaturnachweise beeinflussen und unter bestimmten Umständen sich auf die Zitationshäufigkeit eines Autors bzw. dessen wissenschaftlicher Reputation auswirken. Da Trefferanzeigen von Suchanfragen in enger Beziehung zur Dokumentenproduktion stehen, könnte sich daraus ein Konfliktpunkt innerhalb der oben erwähnten institutionellen und disziplinären Spannungen ergeben. Die Kriterienbildung für die Trefferreihenfolge muss also vor dem Hintergrund der soziotechnischen Produktionsprozesse betrachtet werden.

4 Schlussbetrachtungen

Das Fallbeispiel der FIS Bildung Literaturdatenbank zeigt, dass sich in die Recherche-Situation nicht nur von Seiten der Nutzer her Sozialität in das Retrieval System einschreibt, sondern ebenso – ob beabsichtigt oder nicht – durch die Wissens- und Produktionspraxen bei der Erstellung selbst. Für die IR-Evaluation hat dies unter anwendungsbezogenen Bedingungen, wie sie beim Qualitätsmanagement gegeben sind, weitreichende Folgen. Ohne den Einbezug soziotechnischer Zusammenhänge sind so nur begrenzt Aussagen über die alltägliche Recherche-Situation mit ihren komplexen Nutzungs- und Produktionspraxen zu treffen. Nichtbeachtete soziokulturelle Zusammenhänge schreiben sich als implizite Annahmen in die Versuchskonstruktion sowie -ergebnisse ein und entziehen sich dadurch ihrer Steuerung bzw. Operationalisierung.

Es erscheint notwendig, die Recherchesituation als analytisch offen zu bezeichnen um neue Zusammenhänge und Wechselwirkungen zwischen alltäglichen Praxen, Artefakten und Menschen in Betracht ziehen zu können. Diese Perspektive lässt sich in der Analyse von Nutzungssituationen heute zunehmend wiederfinden. Ebenso sollte sich dieser Ansatz jedoch auch in der Analyse von Produktionsprozessen niederschlagen. Arbeitsroutinen, Wissenspraxen und unterschiedliche Professionsgemeinschaften müssten mit einbezogen werden. Der entstehende Beschreibungsrahmen würde die Bildung von Kriterien und Evaluationsbedingungen in Bezug auf die Alltagspraxen der beteiligten Communities ermöglichen.

Abschließend lässt sich sagen, dass die Komplexität der alltäglichen Recherche-Situation eine interdisziplinäre Betrachtungsweise erfordert, die im besten Falle nicht nur von einer reinen Addition disziplinärer Sichtweisen ausgeht, sondern sich mit unterschiedlichen disziplinären Praxen, Grenzziehungen und Formen der Erkenntnisgewinnung sowie deren Auswirkungen auseinandersetzt.

Quellen

[Akrich 1989] M. Akrich. La construction d’un système socio-technique. Esquisse pour une anthropologie des techniques. In: Anthropologie et Sociétés, 13 Nr. 2. 1989.

[Belkin et al. 1982a] N. Belkin, R. Oddy, H. Brooks. *ASK for information retrieval. Part I - background and theory*. In *Journal of Documentation*, 38 Nr. 2, 61-71. 1982.

[Belkin et al. 1982b] N. Belkin, R. Oddy, H. Brooks. ASK for information retrieval: Part II. Results of a design study. In Journal of Documentation, 38 Nr. 3, 145-164. 1982.

[Bishop et al. 2003] A. P. Bishop, T. Pfeil Hoshi, N. A. Van House (Eds.). *Digital library use: Social practice in design and evaluation*. MIT Press, Cambridge, MA. 2003.

[Callon 1987] M. Callon. Society in the Making: The Study of Technology as a Tool for Sociological Analysis. In. W. E. Bjker; T. P. Hughes, T. J. Pinch (Eds). The Social Construction of Technological Systems. MIT Press, Cambridge, London. 1987.

[Fachportal Pädagogik] Fachportal Pädagogik. *Policy der FIS Bildung Literaturdatenbank.* http://www.fachportal-paedagogik.de/fis_bildung/ fis_policy.html ohne Datum.

[Ferber 2003] R. Ferber: *Information Retrieval*. dpunkt, Heidelberg. 2003.

[Fry 2006] J. Fry: *Scholarly research and information practices*. In *Information Processing and Management* 42, 299–316. 2006.

[Ilyes 2006] P. Ilyes. *Zum Stand der Forschung des englischsprachigen „Science and Technology“ (STS)-Diskurses*.2006. http://www.sciencepolicystudies.de/dok/ STS-Forschungsstand-1.1.pdf 2006.

[Knorr-Cetina 1999] K. Knorr-Cetina: *Epistemic Cultures*. Harvard University Press, Cambridge. 1999.

[Lewandowski 2005] D. Lewandowski. *Web Information Retrieval.* Reihe Informationswissenschaft der DGI. 2005

[MacKenzie 1987] D. MacKenzie. *Missile Accuracy: A Case Study in the Social Processes of Technological Change.* In W. E. Bjker; T. P. Hughes, T. J. Pinch (Eds). *The Social Construction of Technological Systems*. MIT Press, Cambridge, London. 1987.

[Rittberger 2004] M. Rittberger. *Informationsqualität*. In *Grundlagen der praktischen Information und Dokumentation*. R. Kuhlen, T. Seeger, D. Strauch (Eds). Saur, München, 315-321. 2004.

[Rittberger, Rittberger 1997] M. Rittberger, W. Rittberger. *Quality Measuring in the Production of Databases.* In *Journal of Information Science*, 23 Nr. 1, 25-37.1997.

[Tague-Sutcliffe 1992] J. Tague-Sutcliffe. *The Pragmatics of Information Retrieval Experimantation, Revisited.* In: *Information Processing & Management* 28 Nr. 4, 467-490. 1992.

[Ulrich et al. 2003] S. Ulrich, F. M. Wenzel: Partizipative Evaluation. Bertelsmann Stiftung, Gütersloh. 2003.

[Van House 2003] N. A. Van House. *Digital Libraries and Collaborative Knowledge Construction*. In A. P. Bishop, T. Pfeil Hoshi, N. A. Van House (Eds.). *Digital library use: Social practice in design and evaluation*. MIT Press, Cambridge, MA. 2003.

Ein Schema zur Auswahl geeigneter Evaluationsmethoden für die Evaluation von Information Retrieval Systemen mit Visualisierungskomponente

Sonja Hierl
Hochschule für Technik und Wirtschaft, Chur
CH-7000, Chur, Schweiz
sonja.hierl@fh-htwchur.ch

Abstract

Folgende Ausarbeitung begegnet dem Missstand, dass trotz der sich schnell entwickelnden Angebote von Suchmaschinen mit visueller Ergebnisrepräsentation noch kein Konsens gefunden wurde über eine gemeinsame Basis, auf deren Grundlage nachhaltige Evaluationen von Information Retrieval Systemen mit Visualisierungskomponente durchgeführt werden können. Diese Problematik wird anhand einer State-of-the-Art-Analyse aufgezeigt und es wird ein Lösungsvorschlag erarbeitet und exemplarisch getestet, der einen integrierten Ansatz zur Kombination geeigneter Evaluationsmethoden auf Grundlage eines morphologischen Rahmens empfiehlt.

1 Einführung

Auf dem Suchmaschinenmarkt ist ein Trend hin zu Information Retrieval Systemen (IRS) mit integrierter Visualisierungskomponente (VK) erkennbar.

Zum einen berichten diverse Publikationen, wie beispielsweise [Koshman, 2005; Reiterer, 2005; Zwol und Oostendorp, 2004; Reiterer, 2004; Mann, 2002; Sebrechts et al., 1999; Veeresamy und Belkin, 1996] von Forschungsprojekten zur Entwicklung solcher Applikationen, zum anderen steigern auf dem Markt befindliche Sysetme wie Grokker, Kartoo, Webbrain oder Liveplasma ihren Bekanntheitsgrad insbesondere innerhalb von Unternehmen als Desktop- bzw. Enterprise Search Engines.

Ziel derartiger IRS, die in der Regel eine Visualisierung der Ergebnisrepräsentation vornehmen, ist die Optimierung der Effektivität während des Suchprozesses aus Nutzersicht. Dies geschieht beispielsweise durch die Repräsentation von Relationen zwischen selektierten Ergebnisobjekten oder der Darstellung von Themenclustern, wodurch eine Aussage über die inhaltlichen Zusammenhänge der Treffer ermöglicht wird.

[Vaughan, 2004] stellt fest, dass die Entwicklung valider Evaluationstechniken derzeit nicht Schritt halten kann mit der rapiden Geschwindigkeit der Neuentwicklungen von Suchmaschinen mit visueller Ausgabe.

Zwar gibt es diverse Studien, in denen die im Rahmen von Forschungsprojekten entwickelten IRS mit VK abschliessend einer Qualitätsprüfung unterzogen wird. Diesen Studien ist jedoch gemein, dass sie mit den angesetzten Evaluationen unterschiedliche Zielsetzungen verfolgen und ihnen weiterhin kein einheitliches und systematisch basiertes Evaluationskonzept oder Untersuchungsdesign zugrunde liegt, anhand dessen kontrollierte Tests durchgeführt werden können, die anschliessend einen Vergleich der Ergebnisse ermöglichen.

Wie [Arnold, 2004] feststellt, liegen derzeit darüber hinaus nur sehr wenige breit abgestützte Untersuchungen vor, die die Wirksamkeit der Visualisierungen von IRS mit VK evaluieren und mit den erzielbaren Ergebnissen konventioneller Suchmaschinen und Information Retrieval Systemen mit Textausgabe vergleichen.

Durch das Fehlen einer gemeinsamen Grundlage und nachhaltiger Untersuchungen können nach [Cugini et al., 2000] folglich kaum repräsentative Schlüsse und empirisch abgestützte Aussagen zur generellen Wirksamkeit von Visualisierungen in der Ergebnisrepräsentation von IRS getroffen werden:

„One of the lessons of our experience is that no matter how much intuitive appeal a given interface might have, without some systematic testing, its real value remains unknown. Especially in the field of visualization, it is all too common for technical wizardry to be unaccompanied by any gain in efficiency."

2 State-of-the-Art der Evaluation von IRS mit integrierter Visualisierungskomponente

[Chen und Yu, 2000] führten erstmals eine umfassende Studie durch, die eine Meta-Analyse aktueller, empirischer Evaluationen von visuellen Informationssystemen zum Ziel hatte und identifizierten dabei u. a. die bereits oben skizzierten Problemstellungen:

„Empirical studies on information visualization are still very diverse and it is difficult to apply meta-analysis methods (…). A larger homogenous sample of studies would be needed to expect conclusive results"

Die Autoren kommen zum Schluss, dass die Schaffung einer einheitlichen Grundlage für die Evaluation von IRS mit VK in Form eines Referenzframeworks dringend erforderlich ist:

"This is the first attempt in raising the awareness that it is crucial to conduct empirical studies concerning information visualization systematically within a comparable reference framework"

[Plaisant, 2004] untersucht vier Jahre später auf dieser Metastudie aufbauend die gängige Evaluationspraxis für IRS mit Visualisierungskomponente und kommt zum Schluss, dass sich auch dann noch kein einheitliches Evaluationsdesign durchgesetzt hat.

Diese Feststellung wiederholen [Shneiderman und Plaisant, 2006] im Mai 2006. Die Autoren fordern einen Trend weg von Labor-geprägten Usabilitystudien hin zu ethnographischen Studien in der üblichen Arbeitsumge-

bung der Probanden, bei denen Unterbrechungen, Arbeitsplatz, Hilfeleistungen und der soziale Austausch wie gewohnt vorliegen. Weiterhin betonen sie die Relevanz der Durchführung von Studien, bei denen Probanden reale Aufgaben ihrer täglichen Arbeit durchführen und nicht vorgegebene Testaufgaben.

Auch in dieser Publikation erfolgt die Aufforderung zur Schaffung eines Evaluationsframeworks, anhand dessen IRS mit Visualisierungskomponenten untersucht und die Ergebnisse verglichen werden können.

Den Ansatz von Langzeitstudien sehen die Autoren hierbei als mögliche Grundlage für abgesicherte und generalisierbare Evaluationsergebnisse:

„Observations of dozens of users over months and years would do much to improve the reliability, validity, and gerealizability of the results. If dozens of software engineers using a source code static analysis visualization tools were found to eagerly adopt this new tool and increased their usage over several months this would be compelling evidence".

3 Interdependenzen zwischen Visualisierungs- und Retrievalkomponenten

Die Auswertung diverser Evaluationsstudien, u. a. von [Vaughan, 2004; Gremy *et al.,* 2006; Beg, 2005; Buckley und Voorhees, 2000; Jensen *et al*., 2005; Kwan und Venkatsubramanyan, 2006; Hawking *et al*., 2001; Mann, 2002; Reiterer *et al*., 2005; Zwol und Oostendorp, 2004] ergibt, dass zur Evaluation von IRS mit VK im wesentlichen Methoden aus den Bereichen der Retrievalperformanzmessung und der Gebrauchstauglichkeitsmessung angesetzt werden. So können neben den von [Plaisant, 2004] aufgeführten häufig vertretenen Ansätzen

- „Kontrollierter Experimente zum Vergleich von Designelementen
- Usabilityevaluation einer Anwendung
- Kontrollierter Experimente zum Vergleich zweier oder mehrerer Anwendungen, sowie
- Case Studies in realistischen Szenarien."

vor allem Ansätze identifiziert werden, in denen anhand klassischer Retrievaleffektivitätsmasse wie Recall und Precision die Retrievaleffektivität von IRS mit VK gemessen wird.

In den angesetzten Evaluationen wird jedoch meistens nicht auf eine bewusste Kombination geeigneter Methoden geachtet, vielmehr werden angewendete Methoden voneinander losgelöst betrachtet.

Es wird folglich nicht berücksichtigt, dass zwischen Visualisierung, Usabilitymessung und Retrievaleffektivitätsmessung durchaus Interdependenzen bestehen, auf die es zu achten gilt.

So haben aus Sicht der Autorin beispielsweise Interaktionen von Nutzer auf der visuellen Oberfläche einen Einfluss auf die Retrievalfunktionalität eines Systems, was wiederum Auswirkungen auf die Usability des IRS hat.

Weiterhin lassen sich in Abhängigkeit von Retrievalfunktionalität und Gebrauchstauglichkeit Ansprüche an die Gestaltung der Nutzeroberfläche identifizieren. Durch die Kombination von Retrieval- und Visualisierungskomponente ergeben sich folglich Interdependenzen, die sich auf die Qualität des Systems auswirken, wie in Abbildung 1 dargestellt.

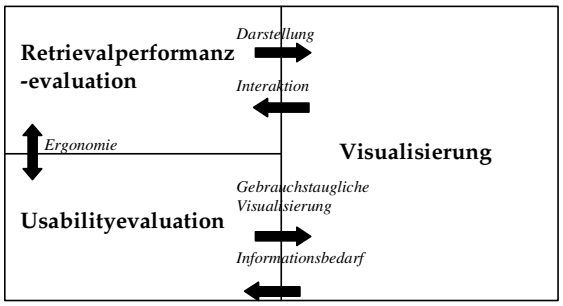


Abbildung 1: Interdependenzen Retrivalperformanz-Visualisierung-Usabilityevaluation

Die gegenseitigen Einflüsse der drei Aspekte lassen sich wie folgt erläutern.

3.1 Interdependenzen zwischen Retrievaleffektivitätsevaluation und Visualisierung

Die visuelle Darstellung von Retrievalergebnissen ermöglicht in der Regel eine Interaktion zwischen Nutzer und IRS. Durch Interaktions- und Verzerrungstechniken kann der Anwender des Systems die Darstellung der Treffer verändern, eine Auswahl oder Einschränkung vornehmen und somit nicht nur die Trefferdarstellung, sondern auch die durchgeführte Suchanfrage verändern [Keim, 2004].

Die Visualisierung hat folglich unmittelbaren Einfluss auf die im System implementierten Retrievalalgorithmen: In Abhängigkeit der Interaktions¬mög¬lich¬keiten des Nutzers, muss das IRS die entsprechenden Veränderungen der Suchanfrage und der Ergebnismenge verarbeiten um anschliessend die erneut ermittelten und verfeinerten Ergebnisse visuell zu präsentieren.

Im Gegenzug hat auch das Retrieval auf die Visualisierung einen unmittelbaren Einfluss, da beispielsweise Dataminingalgorithmen zu Ergebnissen führen, bei denen Cluster, Hierarchien, Beziehungen etc. vorhanden sind, die es in geeigneter Form zu Visualisieren gilt.

Die Retrievalperformanzevaluation kann somit als Interdependent zur implementierten Visualisierung gewertet werden und umgekehrt.

3.2 Interdependenzen zwischen Visualisierung und Usabilityevaluation

Eine der zentralen Aufgaben einer Visualisierung zur Retrievalergebnisrepräsentation ist mitunter die Hilfestellung der Definition des Informationsbedürfnisses des Nutzers. Wie die viel zitierten Arbeiten [Belkins *et al.,* 1982] belegen, besteht bei der Formulierung von Suchanfragen durch Anwender eines IRS das Problem des Anomalous State of Knowledge: Der Nutzer kennt lediglich sein subjektiv empfundenes Informationsbedürfnis, oder Teile davon, nicht jedoch den tatsächlich vorhandenen objektiven Informationsbedarf.

Durch diese fehlende Kenntnis des Informationsbedarfs ergibt sich ein Defizit in der Suchanfrageformulierung, die unter Umständen zu unrelevanten Treffermengen führt.

Ziel eines IRS mit Visualisierungskomponente sollte es sein, diesem Defizit zu begegnen und dem Nutzer durch die Darstellung beispielsweise semantisch belegter Relationen von Treffern oder geclusterten Ergebnissen eine Hilfestellung zu bieten in der Erkenntnis, welche Informa-

tion seinen Bedarf hinsichtlich der zu erfüllenden Aufgabe er tatsächlich decken könnte.

Der Erfolg einer Visualisierung, diesen Mehrwert für den Nutzer zu bieten, ist in einer Evaluation des IRS zu bewerten. Mithilfe Methoden der Usabilityevaluation sollte also nicht nur grundsätzlich festgestellt werden, ob ein IRS mit VK für den Nutzer allgemein gebrauchstauglich gestaltet ist, sondern auch ob die gewählten Visualisierungen dem Zweck der Definition des Informationsbedürfnisses dienlich sind.

Im Gegenzug hat auch das Ziel der Erreichung einer höchstmöglichen Gebrauchstauglichkeit Einfluss auf die VK, da die gewählten Visualisierungen nicht nur geeignet sein sollten im Hinblick auf die zugrunde liegende Datenbasis und die zu visualisierenden Elemente. Vielmehr sollten Visualisierungskomponenten auch den Anforderungen entsprechen, gebrauchstauglich zu sein in Bezug auf ihre Intuitivität und Verständlichkeit im eingesetzten Kontext (also beispielsweise die Verwendung intuitiv interpretierbarer Metaphern als Symbole bei einer Relationenvisualisierung).

3.3 Interdependenzen zwischen Usabilityevaluation und Retrievaleffektivitätsevaluation

Hinsichtlich der Retrievalperformanz- und Usabilityevaluation sind ebenfalls Interdependenzen zu beachten.
Einerseits kann festgestellt werden, dass viele Instrumente, die zur Messung der Gebrauchstauglichkeit verwendet werden, bei entsprechendem Einsatz auch wertvolle Ergebnisse hinsichtlich der Retrievalperformanzevaluation erbringen können.

Z. B. kann das automatisierte Mitloggen von Interaktionen eines Probanden, das für die Aufmerksamkeitsanalyse in der Usabilitymessung eingesetzt wird, ebenfalls für die Datengewinnung genutzt werden, auf deren Basis sich Recall- oder Precisionwerte errechnen lassen.

Durch eine breite methodische Abstützung kann somit ein Phänomen anhand unterschiedlicher Vorgehensweisen gemessen und die Ergebnisse besser interpretiert werden.

Bei der Planung und Durchführung einer umfassenden Evaluation sollte folglich auf eine gewissenhafte Zusammenstellung der eingesetzten Instrumente geachtet werden.

Zum anderen kann festgestellt werden, dass Anforderungen an ein System bzw. dessen Oberfläche, die sich aufgrund von Aspekten der Gebrauchstauglichkeit ergeben, durchaus einen Einfluss haben können auf die Gestaltung der Retrievalperformanz und umgekehrt.
Soll z. B. bei einem IRS die für den Nutzer erforderliche Art des Informationsbedürfnisses berücksichtigt werden, um einen darauf optimierten Retrievalalgorithmus anzuwenden, hat dies zur Folge, dass das System aus Nutzersicht komplexer und dadurch schwerer zu verwenden wird. Während für versierte Anwender die Auswahl von Vorteil ist, könnte sie sich für ungeübte Nutzer aus Sicht der Gebrauchstauglichkeit überflüssig oder gar nachteilig gestalten. Es müssen folglich Wege gefunden werden, diese aus Retrievalperformanzsicht sinnvolle Ergänzung auf ergonomische Weise in die Systemoberfläche zu integrieren um Einbussen in der Gebrauchstauglichkeit zu vermeiden.

Allgemein müssen Aspekte der Retrievaleffektivität, wie beispielsweise Metaphern, die für die Relationenvisualisierung eingesetzt werden, in der Usabilityevaluation berücksichtigt werden und deren Eignung für die Durchführung der Recherche überprüft werden.
Die wechselseitigen Interdependenzen zwischen Usabilityevaluation und Retrievaleffektivitätsmessung dürfen folglich nicht unberücksichtigt bleiben und es muss ein gezielter Einsatz von Methoden vorgenommen werden, der einen ganzheitlichen und integrierten Ansatz verfolgt.

3.4 Auswirkungen auf das Evaluationsdesign

Zusammenfassend stellt sich bei einer Evaluation eines IRS mit VK nicht nur die Frage, ob eine Visualisierungsart für den Einsatz in einem bestimmten IRS geeignet ist, sondern auch, ob sie den implementierten Retrieval¬algorithmen entspricht, diese in geeigneter Form abbilden kann und weiterhin ob sie Gebrauchstauglich ist im Sinne der Unterstützung des Nutzers bei der Identifikation und Deckung seines Informationsbedarfs.

Diese Aspekte müssen bei einer Evaluation ebenso berücksichtigt werden, wie die klassischen Fragestellungen der Retrievalperformanz oder der Eignung eingesetzter Metaphern für die Ergebnisrepräsentation, was einen breit angelegten Ansatz zur Folge hat.

Die Herausforderung besteht somit darin, anhand eines integrierten Designs ein Evaluationsframework zu entwickeln, das nicht nur Methoden kombiniert, sondern auch den oben genannten neu auftretenden Fragestellungen begegnet.

Weiterhin sollte eine Grundlage geschaffen werden, auf der ein Vergleich durchgeführter Evaluationen möglich ist um Synergieeffekte zu erzielen und Unterschiede sowie Ähnlichkeiten in den Resultaten identifizieren, analysieren und daraus generalisierbare Schlüsse ziehen zu können.

4 Morphologischer Kasten für die Auswahl geeigneter Methoden zur Evaluation von IRS mit Visualisierungskomponente

4.1 Spannungsfeld bei der Kombination geeigneter Methoden

Bei der Kombination von Methoden zur Evaluation von IRS mit VK befinden sich die Instrumente im Spannungsfeld zwischen Laborstudien einerseits, die durch eine künstlich geschaffene Umgebung sehr präzise aber unter Umständen auch leicht verfälschte Ergebnissen erzielen lassen und Feldstudien andererseits [Plaisant, 2004]. Bei letzteren werden Probanden für eine Evaluation möglichst authentische und reale Arbeitsbedingungen geboten, wobei die Störfaktoren nicht behoben werden und Ursachen für die Artung der Ergebnisse aus diesem Grund nicht immer identifizierbar sind.

Als weitere Dimension sind die Objektivität der Ergebnisse einerseits und ein hoher Grad an Nutzerbeteiligung andererseits zu berücksichtigen, wie aus dem Portfolio in Abbildung 2 ersichtlich wird.

Die Zusammenstellung eines integrierten Methoden-Mix für die Evaluation eines IRS mit VK sollte dieses Spannungsfeld berücksichtigen und das Portfolio zur Begegnung oben Identifizierter Herausforderungen stets umfassend abdecken.

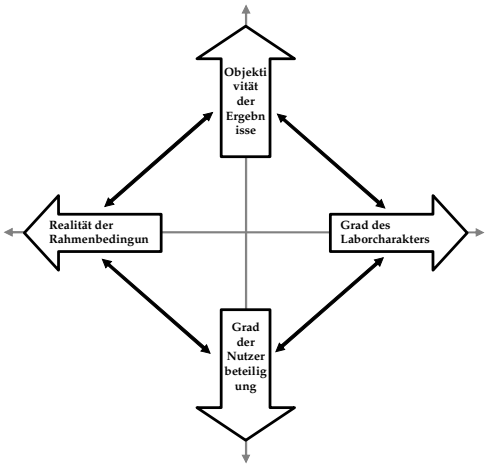


Abbildung 2: Spannungsfeld der Evaluation von IRS mit Visualisierungskomponenten

Auf Grundlage der untersuchten Studien zur Evaluation von IRS mit VK wurden rund 40 Methoden in verschiedenen Variationen aus den Bereichen der Retrievaleffektivitäts- und der Gebrauchstauglichkeitsmessung zusammengetragen und charakterisiert sowie hinsichtlich des oben erläuterten Spannungsfelds klassifiziert. Eine Zusammenstellung der Methoden findet sich unter folgender URL:
http://www.informationswissenschaft.ch/index.php?id=299

4.2 Morphologischer Kasten für die Klassifikation von Evaluationsmethoden

Für die Klassifikation von geeigneten Methoden und Instrumenten, die den Zielsetzungen einer Evaluation entsprechen, wurde in Anlehnung an [Mussgnug und Stowasser, 2004] ein morphologischer Kasten entwickelt, anhand dessen sich unterschiedliche Ansätze zwischen Usability- und Retrievaleffektivitätsevaluation einordnen lassen und gleichzeitig dem oben aufgezeigtem Spannungsfeld Rechnung trägt (Vergleiche Tabelle 1).

DIMENSION	**SKALA bzw. Ausprägung**			
Zielsetzung				
Zielsetzung und Art der Evaluation	Vergleichend	Funktionalität	Leistung (Algorithmen)	Design
Rahmenbedingungen				
Kosten	Gering	Mittel	Hoch	
Trainingsaufwand	Gering	Mittel	Hoch	
Menge erforderlicher Probanden	Keine	Wenige	Viele	
Erforderliche Apparaturen bzw. Software	Auflistung (m) = Muss, (k) = Kann			
Durchführung				
Zeitlicher Aufwand	Gering	Mittel	Hoch	
Untersuchungsort	Feld	Kontrolliertes Feld	Labor	
Auswertungsdimensionen				
Ergebnisdimension	Qualitativ		Quantitativ	
Bezugsdimension	Subjektiv		Objektiv	
Untersuchungsart	Analytisch	Experimentell	Beobachtend	Fragend

Tabelle 1: Morphologischer Kasten für die Klassifikation von Evaluationsmethoden für IRS mit Visualisierungskomponente

Auf Grundlage dieses morphologischen Kastens wurde eine Morphologie erstellt mit den identifizierten Methoden und Evaluationsinstrumenten, die ebenfalls unter folgender URL aufgerufen werden kann: http://www.informationswissenschaft.ch/index.php?id=299

4.3 Ergebnisse einer exemplarischen vergleichenden Evaluation

Auf den bisherigen Erkenntnissen aufbauend wurde im Rahmen einer kleinen Beispielevaluation, die keinen Anspruch erhebt, repräsentativ zu sein, der vorgeschlagene Ansatz in der Empirie exemplarisch überprüft.

Hierzu wurde eine vergleichende Evaluation der Suchmaschinen Yahoo (http://search.yahoo.com) mit konventioneller Listenausgabe und Grokker (www.grokker.com) mit einer visuellen Ergebnisrepräsentation mit fünf Probanden vorgenommen, bei der unter anderem folgende Methoden zum Tragen kamen:

- Kontrolliertes Experiment (Usability-Test) mit vorgegebenen Aufgabenstellungen
- „Lautes Denken“
- Screen-Capturing mit Auswertung
- Fragebogen
- Tagebuchstudie
- Retrievaleffektivitäts-Evaluation zur Erhebung u. a. folgender Masse: Relative Recall@n, Precision@n, Jewel Measure, First Retrieved Document Rank nach [Kwan und Venkatsubramanyan, 2006]

Bei der Auswertung der Evaluation kann festgestellt werden, dass sich die in einem Fragebogen von Probanden geäusserten Bewertungen nicht immer decken mit den anderweitig erhobenen Kennzahlen.

Beispielsweise gaben alle Probanden an, sie hätten das Gefühl, die Anzahl der erforderlichen Interaktionen seien bei Grokker höher gewesen, als bei Yahoo. Die Auswertung des Screen-Capturing ergibt jedoch, dass während des Usability-Tests zur Erfüllung der Aufgaben im Schnitt rund 20% mehr Interaktionen auf der Oberfläche von Yahoo vorgenommen wurden, als auf der visuellen Oberfläche von Grokker.

Ähnliche scheinbare Widersprüche ergaben sich hinsichtlich der Qualität der erzielten Treffer, die mit Methoden der Retrievaleffektivitätsmessung und der Auswertung des Screen-Capturing erhoben und gleichzeitig durch Befragung der Probanden eingeschätzt wurde.

In diesen Beispielen kann anhand der breiten methodischen Abstützung festgestellt werden, dass sich durch das IRS mit VK zwar bessere Ergebnisse erzielen liessen, sich diese Feststellung jedoch nicht in der Einschätzung aus Probandensicht widerspiegelte.

Die exemplarische Evaluation macht somit deutlich, dass bei gezieltem Einsatz und der integrierten Kombination geeigneter Methodenansätze durchaus Synergieef-

fekte zum Tragen kommen. Weiterhin wird daraus ersichtlich, dass sich bereits mit wenigen Mitteln eine breit abgestützte Evaluation durchführen lässt, die durch den Einbezug qualitativer und quantitativer Masse eine gute Ausgangsbasis bieten für die Interpretation von Ergebnissen sowie die Identifikation von Auslösern scheinbarer Differenzen im Erhebungsmaterial.

Die Schwächen einiger Evaluationsmethoden werden somit durch die Stärken anderer Methoden ausgeglichen und die erhobenen Ergebnisse weisen insgesamt eine höhere Qualität auf und lassen sich besser auswerten und interpretieren.

5 Ausblick

Die Auswahl und Kombination geeigneter Methoden für eine nachhaltige und umfassende Evaluation von IRS mit Visualisierungskomponente bedeutet nach wie vor eine grosse Herausforderung.

Anhand einer Morphologie lässt sich ein integrierter Ansatz verfolgen, der eine möglichst breite Abstützung der Ergebnisse hinsichtlich aller relevanten Aspekte eines IRS mit visueller Ausgabe gewährleistet. Künftig gilt es auf Grundlage des Evaluationsframeworks eine konkrete Evaluationsumgebung für die Durchführung von Evaluationen von IRS mit VK zu gestalten, die als Ausgangsbasis für den evaluationenübergreifenden Vergleich dient. Auf diesen Ergebnissen beruhend lassen sich langfristig repräsentative und allgemeingültige Aussagen zur Eignung von Visualisierungen im Information Retrieval ableiten.

Das vorgelegte Framework gilt es hierzu in weiteren Schritten anhand der Ergebnisse empirischer Erprobung sukzessive zu verfeinern und zu optimieren. Vorliegender Vorschlag ist somit als erster Schritt in Richtung einer einheitlichen Evaluationsgrundlage zu verstehen, die gemäss der Aussage diverser Autoren langfristig gesehen unabdingbar ist.

Referenzen

[Arnold, 2004] C. Arnold. Visualisierung im Information Retrieval. Magisterarbeit in der Philosophischen Fakultät IV (Informationswissenschaft) der Universität Regensburg: Regensburg, 2004.

[Bar-Ilan *et al.*, 2004] J. Bar-Ilan, M. Levene, M. Mat-Hassan. Dynamics of search engine rankings - a case study. In *Proceedings of the 3rd international workshop on web dynamics*, New York, 2004.

[Beg, 2005] M. M. S. Beg. A subjective measure of web search quality. In *Information Sciences* Volume 169, Issues 3-4, 1 February 2005, S. 365-381.

[Belkin *et al.*, 1982] N. Belkin, R. Oddy, H. Brooks. ASK for information retrieval: Part I. background and theory. In *Journal of Documentation*, 38(2) Seiten 61-71, 1982.

[Buckley und Voorhees, 2000] C. Buckley, E. M. Voorhees. Evaluating Evaluation Measure Stability. In *SIGIR 2000*. Belkin, N. J., Ingwersen, P., und Leong, M.-K. (eds.); ACM, S. 33-40, Athen, 2000.

[Chen und Yu, 2000] C. Chen, Y. Yu. Empirical studies of information visualization: a metaanalysis. In *International Journal of Human-Computer Studies*, 53 (2000) 5, Seiten 851-866, 2000.

[Cugini, 2000] J. Cugini. Presenting Search Results: Design, Visualization and Evaluation. In: *Workshop: Information Doors - Where Information Search and Hypertext Link*, San Antonio

[Gremy *et al.*, 1999] F. Gremy, J. M. Fessler, M. Bonnin. Information systems evalua-tion and subjectivity. In *International Journal of Medical Informatics* Volume 56, Issues 1-3 , December 1999, S. 13-23.

[Grokker] Grokker: Suchmaschine Grokker, URL: http://www.grokker.com, Stand: 25.06.2006.

[Hawking *et al*., 2001] D. Hawking, N. Craswell, P. Bailey *et al*. Measuring search engine quality. In *Information Retrieval*, 4 Nr.1, S. 33-59, 2001.

[Jensen *et al*., 2005] E. C. Jensen, S. M. Beitzel, O. Frieder, O. *et al.* A framework for determining necessary query set sizes to evaluate web search effectiveness. In *Special interest tracks and posters of the 14th international conference on World Wide Web*, May 10-14, 2005, Chiba, Japan.

[Kartoo] Kartoo: Suchmaschine Kartoo, URL: http://www.kartoo.com, Stand: 25.06.2006.

[Keim, 2002] D. A. Keim. Information Visualization and Visual Data Mining. In *IEEE Transactions on visualization and computer graphics,* Vol. 7, No. 1, January-March 2002

[Koshman, 2005] Koshman. Testing user interaction with a prototype visualiza-tion-based information retrieval system. In *Journal of the American Society for Information Science and Technology*, 56(8) 2005, Seiten 824-833, 2005.

[Kwan und Venkatsubramanyan, 2006] S. K. Kwan, S. Venkatsubramanyan. An Economic Model for Comparing Search Services. In *Proceedings of the 39th Annual Hawaii International Conference on System Sciences* (HICSS'06). S. 107-116, 2006.

[Liveplasma] Liveplasma: Suchmaschine Liveplasma URL: http://www.liveplasma.com/, Stand: 25.06.2006.

[Mann, 2002] T. M. Mann. Visualization of Search Results from the World Wide Web, Dissertation, Universität Konstanz, 2002.

[Mussgnug und Stowasser, 2004] J. Mussgnug, S. Stowasser. Ein Schema zur Auswahl geeigneter Usability-Methoden - Dargestellt am Beispiel der Blickbewegungsanalyse. In *Proceedings of the 2nd annual GC-UPA Track Paderborn*, September 2004, Paderborn.

[Plaisant, 2004] C. Plaisant. The Challenge of Information Visualization Evaluation. In *Proceedings of Conference on Advanced Visual Interfaces AVI'04*.

[Reiterer *et al.*, 2005] H. Reiterer, G. Tullius, T. M. Mann. INSYDER: a content-based visual-information-seeking system for the Web. In International Journal on Digital Libraries, Volume 5, Issue 1, Mar 2005, Seiten 25 - 41.

[Reiterer, 2004] H. Reiterer. Visuelle Recherchesysteme zur Unterstützung der Wissensverarbeitung. In Hammwöhner, R.; Rittberger, M.; Semar, W. (Hrg.): *Wissen in Aktion. Der Primat der Pragmatik als Motto der Konstanzer Informationswissenschaft. Festschrift für Rainer Kuhlen*. Konstanz, 2004. Seiten 1–21.

[Shneiderman und Plaisant, 2006] B. Shneiderman, C. Plaisant. Strategies for Evaluating Information Visualization Tools: Multi-dimensional In-depth Long-term Case Studies. In *Proceedings of the BELIV'06 Workshop*, Venice Seiten 61–77, 2006.

[Sebrechts *et al.*, 1999] M. Sebrechts, J. Vasilakis, M. Miller, et al. (1999): Visualization of Search Results: A Comparative Evaluation of Text, 2D, and 3D Interfaces. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval,* S. 3-10, 1999.

[Vaughan, 2004] Vaughan, L. (2004): New measurements for search engine evaluation proposed and tested. In *Information Processing and management* 40 (2004) S. 677-691, 2004.

[Veerasamy und Belkin, 1996] A. Veerasamy, N. J. Belkin. Evaluation of a tool for visualization of information retrieval results. In *Proceedings of the 19th Annual international ACM SIGIR Conference on Research and Development in information Retrieval* (Zurich, Switzerland, August 18 - 22, 1996). SIGIR '96. ACM Press, New York, NY, Seiten 85-92, 1996.

[Webbrain] Webbrain: Suchmaschine Webbrain, URL: http://www.webbrain.com, Stand: 25.06.2006.

[Zwol und Oostendorp, 2004] R. Van Zwol, H. Van Oostendorp. Google's "I'm feeling lucky", Truly a Gamble?. In *Zhou, X. et al. (Hrg.) (2004): Web Information Systems - WISE 2004, Proceedings of the 5th International Conference on Web Information Systems Engineering.* Brisbane, Australia, Seiten 378-390, 2004.

Web Content Mining for Information on Information Scientists

Sarah Risse
University of Hildesheim
D-31135, Hildesheim, Germany
sarah_risse@web.de

Abstract

This paper presents a search system for information on scientists which was implemented prototypically for the area of information science, employing Web Content Mining techniques. The sources that are used in the implemented approach are online publication services and personal homepages of scientists. The system contains wrappers for querying the publication services and information extraction from their result pages, as well as methods for information extraction from homepages, which are based on heuristics concerning structure and composition of the pages. Moreover a specialised search technique for searching for personal homepages of information scientists was developed.

1 Introduction

Along with the constant growth of the internet, its importance has continuously increased, while at the same time the problem of information overload has emerged. Finding relevant information in the huge amounts of data to satisfy a precise information need, such as finding information on a currently active scientist, can be quite tedious and time consuming. The area of Web Content Mining comprehends techniques to improve information search and usage on the Web, such as methods for information extraction and integration from web documents and databases, optimisation of search engine results and the development of specialised search engines [Liu and Chang, 2004]. Employing Web Content Mining techniques, a system for searching for information on scientists - homepage URL, email, photo, list of publications and projects - was implemented prototypically for the area of information sciences in German-speaking countries. As resources the publications services DBLP and CiteSeer and the corresponding personal homepage of the target person are used. For the task of finding the personal homepages of information scientists, a specialised search technique was developed.

2 System Overview

As the search term the first and last name of a scientist is entered by the user. This is used to query the publication services DBLP and CiteSeer. From their result pages the publication details are extracted and by comparing the titles of the single publications integrated into one list, thereby eliminating double entries. In the next step a Google-search with the name as search term is performed and the potential homepage of the target person is filtered out of the result list, using heuristics describing typical characteristics of personal homepages in the scientific area. From the homepage the information items academic title, email, photo, list of publications, list of projects and CV are extracted, if present. In the two following sections the methods for information extraction used in the system and the specialised search technique for personal

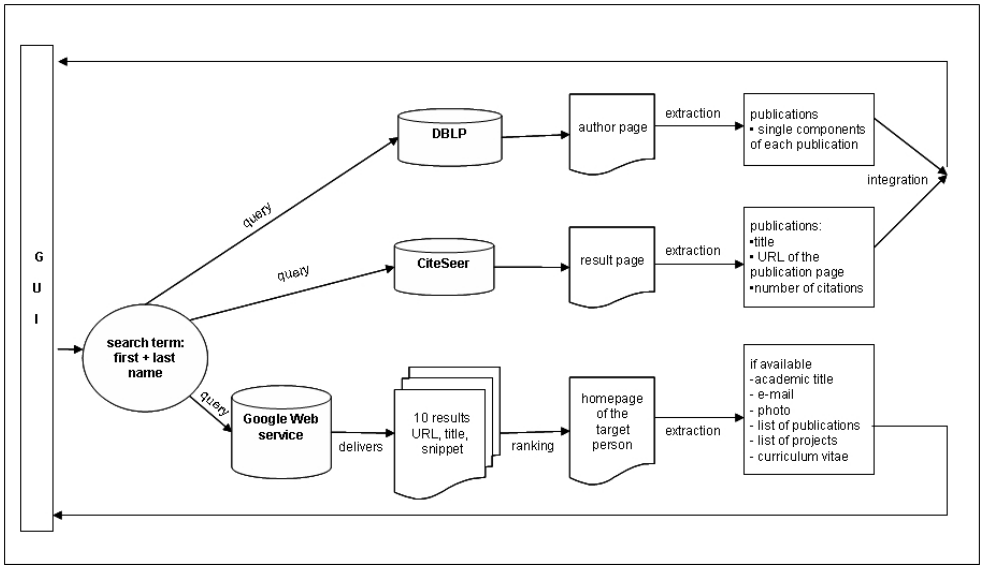


Figure 1: System overview [Risse 2006]

homepages of information scientists are described in more detail.

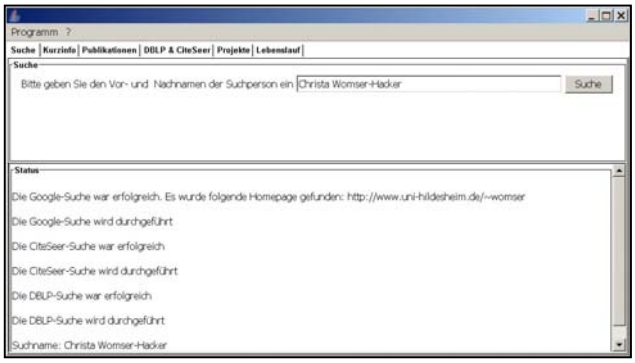


Figure 2: Search page with status information

3 Information Extraction from Source Pages

The manually implemented wrappers for extracting the publication entries from DBLP and CiteSeer make use of the layout and structure of the result pages. Therefore, the relevant HTML-tags are located and the enclosed data extracted.

The methods for locating and extracting the information items from the homepages rely on the observation that the personal homepages of information scientists form a relatively homogeneous set of pages and thus follow certain conventions. These encompass both the structure of the pages and the keywords used for labeling certain content areas. Hence the system assumes that the main page contains the complete name and academic title of the target person, the contact details including the email and a photo of the scientist. Further information items such as a list of publications and research projects as well as a CV are either listed sequentially on a page - separated with subheadings - or located on subpages, that are linked from the main page.

Thus, the academic title, e-mail and photo can be extracted using simple string comparisons and regular expressions - in case of the photo heuristics concerning the metadata provided with the picture are used. The other information items are extracted by using lists of keywords typically used to label these areas, e.g. publications, projetcs, research, CV, curriculum vitae, and the assumption that all subheadings or links to subpages are formatted identically. In the case of a sequentially constructed homepage the extraction rule identifies in this manner all subheadings and extracts the data between two subheadings. In case of a subpage, the rule extracts all data following the respective keyword on that page.

The extracted information items are finally displayed in an integrated way on the result pages of the search system, as figure 3 shows.

4 Specialised Search for Personal Homepages

The developed search technique uses typical characteristics of information scientists' homepages to filter the Google-search results for the homepage of the target person. Following the approach used in HPSearch[1], a specialised search engine for homepages of computer scientists described in [Hoff 2002], URL, title and snippet of information scientists' homepages were analysed. Thus, typical features such as the occurrence of the name of the target person, an academic title or the use of terms as information science identified. These features were transferred into a weighting function, that filters the Google-results, after eliminating e.g. pages from publication services or older pages by comparing the academic titles mentioned.

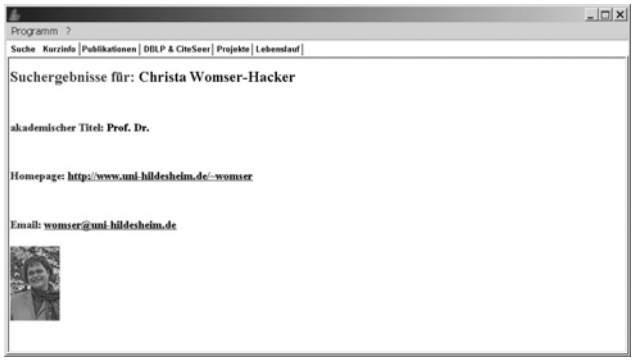


Figure 3: Result page for "Christa Womser-Hacker"

5 Outlook

A first evaluation proved with a recall of 0.65 and a precision of 0.71 for the homepage search module, that the implemented approach is promising. The heuristics used in the weighting function need to be analysed regarding their relevance and their coverage. A ranking approach and a flexible query expansion with frequent terms form the publication list are other potential ways of improvement.

The wrappers for the DBLP and CiteSeer work almost without errors, but a special care needs to be taken for the problem of name disambiguation.

The heuristics applied in the rules for extracting the information items from the personal homepages need to be extended as the evaluation showed that they are not sufficient. Moreover an algorithm to identify content parts in homepages could avoid the extraction of irrelevant data along with relevant information.

References

[Hoff, 2002] Gerd Hoff. Ein Verfahren zur thematisch spezialisierten Suche im Web und seine Realisierung im Prototypen HomePageSearch. *Fachbereich IV der Universität Trier. 2002.*

[Liu and Chang, 2004] Bing Liu and Kevon Cheng-Chuan Chang. Editorial: Special Issue on Web Content Mining. In: *SIGKDD Exploratins Vol. 6(2).* pp 1-4.

[Risse, 2006] Web Content Mining nach Informationen zur wissenschaftlich tätigen Personen im Umfeld der Informationswissenschaft. *Magisterarbeit Universität Hildesheim*.
http://web1.bib.uni-hildesheim.de/edocs/2006/514703415/meta

System URLs:
CiteSeer: http://citeseer.ist.psu.edu
DBLP: http://www.informatik.uni-trier.de/~ley/db

1 The system is available under http://hpsearch.uni-trier.de, but it is not updated anymore.

Multilinguales Web Retrieval im Rahmen von WebCLEF 2006

Ben Heuwing, Robert Strötgen
Information Science, University of Hildesheim,
Marienburger Platz 22
D-31141 Hildesheim, Germany
stroetgen@uni-hildesheim.de

Abstract

Dieser Beitrag beschreibt Retrievalexperimente mit einem umfangreichen multilingualen Korpus im Rahmen von WebCLEF 2006 an der Universität Hildesheim. Im Vordergrund stand die Nutzung von HTML Strukturelementen, der Einsatz von Blind Relevance Feedback und die Evaluierung des sprachunabhängigen Indexierungsansatzes.

1 Einleitung

Einen umfangreichen, heterogenen und multilingualen Korpus effizient in Hinblick auf Rechenleistung zu indexieren und eine hohe Retrievalqualität zu erreichen, war Ziel der diesjährigen Experimente mit dem EuroGOV-Korpus im Rahmen des Web Tracks des Cross Language Evaluation Forum (CLEF).

Für WebCLEF 2005 konnte die Universität Hildesheim mit einem sprachunabhängigen Indexierungsansatz das beste System für mehrsprachiges Retrieval entwickeln [Jensen *et al.*, 2006]. In diesem Jahr sollte versucht werden, das bei der ersten Teilnahme im letzten Jahr verwendete System in Bezug auf die Vorverarbeitung und Säuberung der Korpusdaten zu verbessern. Ein weiteres Ziel war die Implementierung einer Blind Relevance Feedback Option und deren Evaluierung. Die direkte Herangehensweise mit einem multilingualen Index und ohne Übersetzung der Anfragen, die im letzten Jahr zum Erfolg vor allem beim multilingualen Task (Anfragen und Ergebnisse in verschiedenen Sprachen) geführt hatte, wurde erweitert, wobei die jeweils besten Ansätze (vor allem die Indexierung ganzer Wörter statt eines N-Gram Ansatzes) weiter verfolgt wurden. Da sich der multilinguale Track jedoch für viele Teilnehmer als wenig erfolgversprechend erwiesen hatte, wurde in diesem Jahr nur der Mixed-Monolingual Task angeboten, bei dem Ergebnisse nur in der Sprache der Anfrage gefordert sind. Zu Testzwecken wurde trotzdem mit dem neuen System ein multilingualer Run erstellt und eingereicht.

2 Das System

Das System bringt zunächst den Korpus in ein gut weiterzuverarbeitendes Format. Die Indexierung und die Suche wurde in Java und auf der Basis der Suchmaschinen-API *Apache Lucene* implementiert.

2.1 Vorverarbeitung des EuroGOV2-Korpus

Der 80GB große EuroGOV2-Korpus umfasst ca. 3.6 Mio. Internetseiten in verschiedenen Formaten in 20 verschiedenen Sprachen von den Seiten von Regierungen europäischer Länder und der EU. Der Korpus besteht aus 157 Dateien in einem XML-ähnlichen Format mit jeweils maximal 25.000 Dokumenten. Zu jedem Dokument gibt es Metadaten wie die Ursprungs-URL und Angaben aus dem HTTP-Header.

Die eigentlichen Inhalte der Dokumente befinden sich in einem CDATA-Bereich, innerhalb dieser Elemente werden XML-ähnliche Konstruktionen von einem XML-Parser nicht als solche aufgefasst. Allerdings konnten so verschachtelte CDATA-Elemente entstehen, die in XML nicht zulässig sind. Dies ist einer der Gründe, warum das Format des Korpus kein wohlgeformtes XML ist.

Bei der Vorverarbeitung werden daher nun zunächst alle nicht XML-konformen Zeichen [XML W3C Recommendation] heraus gefiltert. Realisiert wurde dies durch einen effizient auf Ebene des Zeichenstroms arbeitenden Filter. Fehlerquelle sind vermutlich die unterschiedlichen Zeichenkodierungen der Dokumente. Dies dürfte Auswirkungen auf die Retrievalqualität vor allem bei problematischen Kodierungen haben.

Einige nicht maskierte Sonderzeichen der in den Metadaten angegebenen URLs müssen im nächsten Schritt unter der Verwendung von regulären Ausdrücken behandelt werden. Auf diese Weise werden auch verschachtelte CDATA- Elemente entfernt, die in wohlgeformten XML-Dokumenten nicht zugelassen sind.

Um mittels eines XML-Parsers gezielt auf Inhalte von einzelnen in den Dokumenten enthaltenen HTML-Elementen zuzugreifen, können diese aus dem CDATA-Bereich extrahiert und als neue Elemente in die XML-Struktur eingefügt werden. Aufgrund dieser Maßnahmen konnten alle Text- bzw. HTML-Dokumente indexiert werden.

2.2 Indexierung

Das entwickelte System setzt als Basis Suchmaschine *Lucene* [Lucene Projekt Homepage] ein. *Lucene* erlaubt die Erstellung von Indizes mit mehreren Feldern. Aufgrund einer vorherigen Analyse der Häufigkeit des Auftretens der verschiedenen HTML-Elemente wurde entschieden, die Inhalte von `<title>` und `<h1>`-Elementen zu einem Indexfeld *title*, die Inhalte der anderen extrahierten Elemente in einem Feld *emphasised* zusammenzufassen, und so beim Retrieval mit unterschiedlichen Gewichtungen experimentieren zu können. Das Dokument wird einmal komplett (*content*) und einmal beschnitten auf 50 Wörter aus der Mitte des Dokuments (*content_cutoff*) indiziert. Termvektoren für das Blind Relevance Feedback (BRF) werden nur für die Felder *title*, *emphasised* und *content_cutoff* berechnet. Auf die Nutzung der gesamten Volltexte für das BRF wurde mit Rücksicht auf die be-

schränkten Ressourcen vor allem bezüglich der Indexerstellung verzichtet. Der Index mit Termvektoren hatte eine Größe von ungefähr 6GB.
Die bereits vorhandene multilinguale Stopwortliste, die 13 Sprachen umfasst, wurde um die im Korpus am häufigsten auftretenden Wörter erweitert. Diese erweiterte Liste beinhaltete 4722 Wörter. Die Idee, eine weitere, titelspezifische Liste einzusetzen, die z.B. auch automatisch erstellte und daher nicht bedeutungstragende Konstruktionen wie ‚no title' berücksichtigt, wurde fallen gelassen, da diese nicht in dem hohen Maße wie vermutet auftraten.

Die Einteilung in einzelne Tokens wurde dem in Lucene vorhandenen *StandardAnalyzer* überlassen, dabei werden bei den Wörtern -s Endungen entfernt und Interpunktion behandelt. Im letzten Jahr hatte sich gezeigt, dass diese Methode auch sprachunabhängig gute Ergebnisse liefern kann.

2.3 Retrieval

Um die Anfragen zu erstellen, wird der *Lucene QueryParser* eingesetzt. Die daraus entstandene Anfrage kann dann durch Gewichtung und Blind Relevance Feedback modifiziert werden. Das Ranking der Ergebnisse basiert auf einer längennormalisierten tf-idf-Formel.

Vor dem Hintergrund des Mixed-Monolingual Tasks nutzen wir zusätzlich die in den Metadaten der Topics angegebenen Zieldomains, um die Ergebnisse auf Dokumente in dieser Domain einzuschränken. Dies ist ohne großen Rechenaufwand möglich durch die *QueryFilter*-Klasse, die den zusätzlichen Vorteil hat, dass sie die Rankingergebnisse nicht weiter beeinflusst. Die Domains sind Teil der Dokument-IDs und werden während des Indexierens extrahiert und in ein eigenes Feld geschrieben. Ohne diesen Schritt musste eine Suche mit Platzhaltern auf den IDs ausgeführt werden, was zu Performance-Problemen führte.

Es wurde darauf geachtet, dass die einzelnen Felder beliebig gewichtet werden konnten, diese Option ist zu Evaluierungszwecken auch über die Kommandozeile verfügbar. Eine hohe Gewichtung des *title*-Feldes (20:1:1 im Verhältnis zu den beiden anderen Inhaltsfeldern) brachte hierbei die größten Vorteile.

Ebenfalls zu Evaluierungszwecken kann die Anzahl der für das Blind Relevance Feedback eingesetzten Dokumente und der für die Anfrage verwendeten Terme sowie die Gewichtung zur ursprünglichen Anfrage von der Kommandozeile aus verändert werden. Es wird eine an der Universität Hildesheim für den CLEF Ad-Hoc Track erstellte Implementierung eingesetzt, die auf den von Lucene bereitgestellten Termvektoren arbeitet und die Termgewichte über einen Robertson Selection Value berechnet [Hackl *et al.,* 2005].

3 Ergebnisse

3.1 Topics WebCLEF 2006

Wie schon erwähnt stand in diesem Jahr der Mixed-Monolingual Task im Vordergrund [WebCLEF Homepage]. Die Topics für dieses Jahr bestanden aus 319 manuell erstellten Topics (neu erstellte und ein Teil der Topics von WebCLEF 2005) und 1620 automatisch erstellten, das hierzu verwendete Verfahren wurde nicht bekannt gegeben.

3.2 Runs

Für die Teilnahme wurden nach Experimenten mit den Topics von WebCLEF 2005 ein Run mit starker Gewichtung des Titels und zwei Runs mit unterschiedlich gewichtetem Blind Relevance Feedback zur Anfrageerweiterung sowie ein Run für den multilingualen Task erstellt. Für alle Runs wurde zum Vergleich der Mean Reciprocal Rank[1] (MRR) mit den Topics von WebCLEF 2005 berechnet und die durchschnittliche Häufigkeit eines Treffers unter den ersten 5, 10, 20, 50 Ergebnissen (*average success*) ermittelt.

Die Runs für den Mixed-Monolingual Task zeigen in den Experimenten (Tabelle 1) insgesamt verbesserte Ergebnisse im Vergleich zu denen von WebCLEF 2005 und den danach durchgeführten Postexperimenten (Tabelle 2). Der beste eingereichte Run für den Mixed-Monolingual Task von WebCLEF 2005 erreichte einen MRR von 0,1603, während in den Postexperimenten ein MRR von 0,2377 erreicht wurde [Jensen *et al.*, 2006]. Der beste Run (*UHiTitle*) in diesem Jahr zeigte mit einem MRR von 0,2807 also eine Verbesserung von 0,043 Punkten. Der *average success at 50* verbesserte sich von 0,253 auf 0,5192.

Die Verbesserungen können auf die umfassendere Indexierung und die erweiterte Stopwortliste, vor allem aber auf die Verwendung zusätzlicher Metadaten (Zieldomain) zurückgeführt werden. Vermutlich führte die umfassendere Indexierung zu einem höheren Recall (Auswirkungen auf *average success*) und der Domainfilter vor allem zu höherer Precision (Auswirkungen auf MRR). Ein ansonsten dem diesjährigen Run *UHiTitle* entsprechender Durchlauf ohne Domainfilter zeigte weniger deutliche Verbesserungen beim MRR (0,0175 Punkte), aber trotzdem einen deutlich verbesserten *average success at 50* von 0,4570.

Der multilinguale Run hatte im Vergleich zum besten des letzten Jahres einen deutlich verbesserten MRR von 0.2443, der *average success at 50* verbesserte sich auf 0.4442. Hier konnte kein Domainfilter verwendet werden, die umfassendere Indexierung hatte jedoch wahrscheinlich positive Auswirkungen auf den Recall.

Das Blind Relevance Feedback in der durchgeführten Form scheint nicht zu Verbesserungen geführt zu haben. Allerdings kann der Einsatz eventuell durch weitere Experimente optimiert werden.

Name	Bemerkungen	MRR	Average Success at 50
UHiTitle	Titel^20	0.2807	0.5192
UHiBRF1	BRF (Gewichtung 1.0)	0.2731	0.4973
UHiBRF2	BRF (Gewichtung 0.5)	0.2771	0.5082
UHiMu	multilingual	0.2443	0.4442

Tabelle 1: Ergebnisse der Runs von 2006 mit Topics von 2005

Name	Bemerkungen	MRR	Average Success at 50
UHiSMo	offizieller Mixed-Monolingual Run	0.1603	0.2870
UHiSMu	offizieller multilingual Run	0.1370	0.2587
UHiSTiMo	bester Mixed-Monolingual der Postexperimente	0.2377	0.2530
UHiSTi101	multilingual Run aus Postexperimenten	0.2117	0.2117

Tabelle 2: Ergebnisse WebCLEF 2005 und Postexperimente [Jensen *et al.*, 2006]

[1]MRR = 1 / Rang des ersten relevanten Dokuments in der Ergebnisliste [Jensen *et al.*, 2006]

Literatur

[Hackl *et al.,* 2005] Hackl, René; Mandl, Thomas; Womser-Hacker, Christa (2005): Ad-hoc Mono- and Multilingual Retrieval Experiments at the University of Hildesheim. In: Working Notes of the 6th Workshop of the Cross-Language Evaluation Forum, CLEF 2005. Sept. 2005, Wien.

[Jensen *et al.*, 2006] Niels Jensen, René Hackl, Thomas Mandl, Robert Strötgen. Web Retrieval Experiments with the Euro-GOV Corpus at the University of Hildesheim. In: Accessing Multilingual Information Repositories: 6th Workshop of the Cross-Language Evaluation Forum, CLEF 2005, Vienna, Austria, Revised Selected Papers. Springer [LNCS 4022]

[Lucene Project Homepage] http://lucene.apache.org

[WebCLEF Homepage] http://ilps.science.uva.nl/WebCLEF/

[XML W3C Recommendation] http://www.w3.org/TR/2004/REC-xml11-20040204/#charsets

Dynamisches Relevanz-Feedback im Patent-Retrievalsystem PatentAide

René Hackl
Universität Hildesheim
31141 Hildesheim, Deutschland
rene.a.hackl@gmx.de

Abstract

Im Patent Retrieval haben sich Rankingverfahren und Methoden wie Relevanz-Feedback noch nicht etabliert. An Ranking Systemen wird vor allem die mangelnde Transparenz für den Benutzer bemängelt. Das System PatentAide versucht, aufbauend auf einer Analyse der Rechercheprozesse im Patent Retrieval, ein Ranking-System zu implementieren. PatentAide unterstützt wichtige Techniken im Patent-Retrieval Prozess wie Term-Erweiterung, bietet ein geranktes Ergebnis und erlaubt darüber hinaus dynamisches Relevanz-Feedback.

1 Einleitung

Patente befinden sich an der Schnittstelle von Recht, Technik und Wirtschaft. Sie sind exklusive Schutzrechte, die dem Patentinhaber für einen begrenzten Zeitraum – maximal 20 Jahre – das alleinige Verwertungsrecht gewähren. Im Gegenzug zur Gewährung dieser umfangreichen Schutzrechte[1] legt der Patentanmelder die Erfindung offen. Es wird geschätzt, dass über 80% des technischen Wissens allein in Patenten vorliegt. Damit sind Patente eine der wichtigsten Quellen detaillierter technischer Information. Für Unternehmen nehmen Patente in Bereichen wie Forschung und Entwicklung, Marketing und Intellectual Property Rights Management die Position einer Schlüsselressource ein. Nicht nur wird der „internationale Kampf um Marktanteile [...] zunehmend über den Innovationsvorsprung entschieden“ [Weckend u. Wurzer, 2004], auch strategische Verfahren wie der Austausch von Patentlizenzen sind zu wichtigen Instrumenten im Unternehmensalltag geworden.

2 Patent Retrieval

Die Patentrecherche ist ein komplexes, zeitaufwändiges Arbeitsgebiet, das weitgehende Anforderungen an Rechercheure stellt. Rechercheergebnisse müssen höchsten Ansprüchen in Bezug auf Gründlichkeit, Qualität und Vollständigkeit genügen. Häufig hängen wichtige Forschungs- und Geschäftsentscheidungen von Patentanalysen ab. Zentrale Bestandteile der Patentrecherche umfassen die Entwicklung von Suchstrategien, die Auswahl von Informationsquellen und den Umgang mit zur Verfügung stehenden Werkzeugen.

Der Zugriff auf und Umgang mit Informationsquellen wird allerdings immer selbstverständlicher, so dass nicht nur Berufsrechercheure als Patentsucher auftreten, sondern auch und in zunehmendem Maße Wissenschaftler, Manager, Ingenieure und Patentanwälte. Daher ist eine Hauptforderung der Nicht-Profis, die Produkte und Werkzeuge mögen sich ihrem Niveau anpassen und sie in der Suche besser unterstützen [Mendelsohn, 2000]. Analog dazu argumentiert [Poynder, 2000], die Usability der vorhandenen Systeme müsse sich verbessern, die Gewinnung von Mehrwert müsse einfacher zu erreichen sein.

Im Information Retrieval etablierte unterstützende Techniken wie Ranking und Relevanz-Feedback [s. Womser-Hacker, 1997] kommen in der Patentdomäne allenfalls sporadisch und nicht den Anforderungen entsprechend zum Einsatz. IR-Systeme, die auf der booleschen Logik aufsetzen, bilden seit Jahrzehnten die Grundlage der Patentrecherche. Neuere Werkzeuge und Produkte verfehlen mehrheitlich die speziellen Anforderungen. Insbesondere wird fehlende Kontrollierbarkeit bemängelt.

3 Gestaltungsprinzipien des Prototypen

Das System PatentAide wurde in einem Kooperationsprojekt zwischen der Informationswissenschaft der Universität Hildesheim und dem Fachinformationszentrum Karlsruhe Gesellschaft für wissenschaftlich-technische Information mbH (FIZ Karlsruhe)[2] entwickelt. Das Projekt geht der Frage nach, inwiefern die unterstützenden Information – Retrieval – Techniken Ranking und Relevanz-Feedback in der Patentrecherche zum Einsatz kommen können**.** Beiden Techniken ist normalerweise ein ausgesprochener *black box* Charakter gemein. Diese Eigenschaft macht sie ungeeignet für den Patentbereich. Das Hauptziel des Prototypen besteht daher darin, zu prüfen, wie transparente Bedingungen für die Komponenten geschaffen werden können. Dabei werden die Angabe statistischer Werte, der Gebrauch von Visualisierungen und verschiedene Interaktionsformen vor dem Hintergrund typischer Arbeitsabläufe thematisiert.

Die Entwicklung des Systems ging von Anfang an unter partizipatorischen Gesichtspunkten vonstatten. Durch das frühzeitige hinzuziehen von Experten und iteratives Vorgehen wurde gewährleistet, dass

[1] §§ 9-14 PatG

[2] http://www.fiz-karlsruhe.de

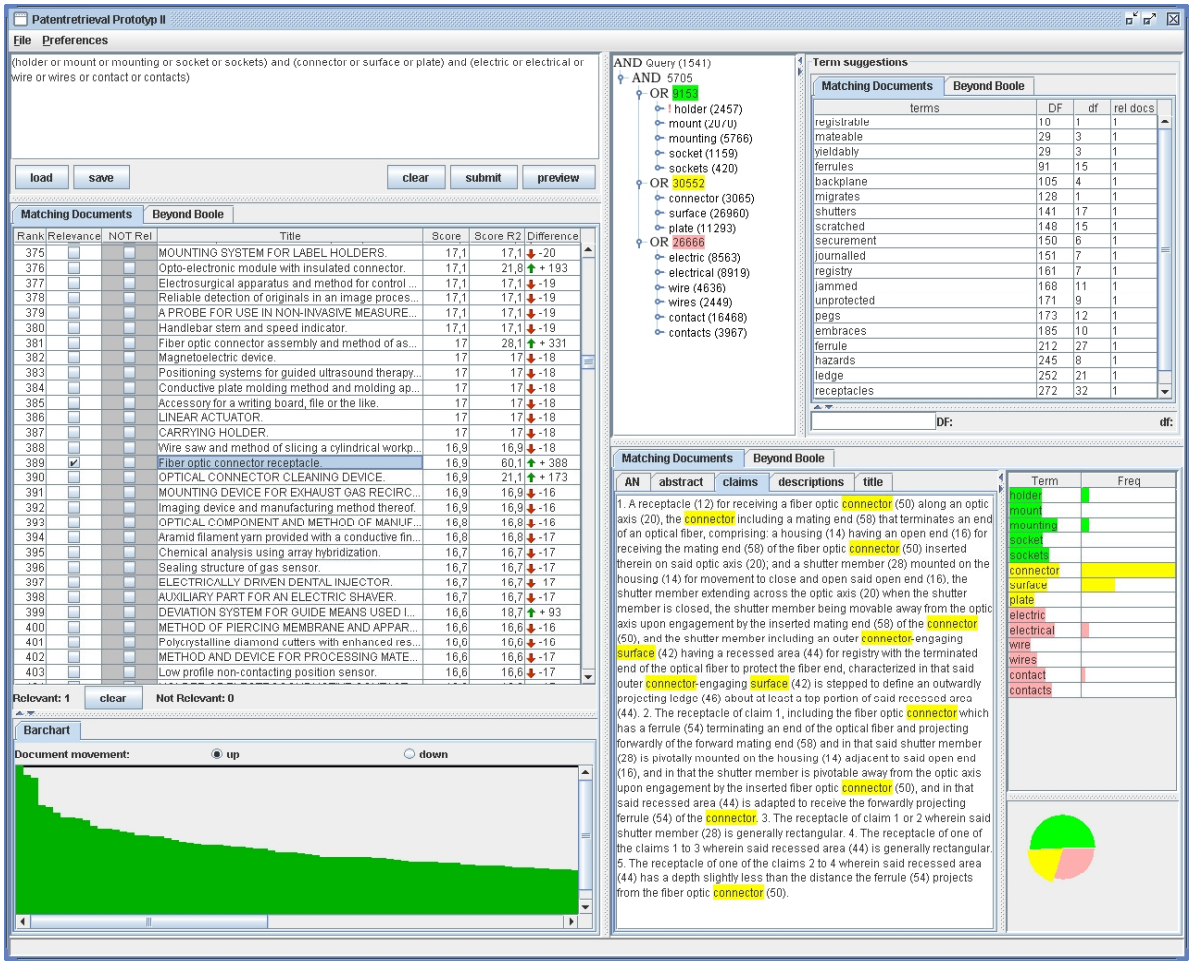


Abbildung 1: Benutzungsoberfläche des PatentAide

gewonnene Erkenntnisse nicht nur von akademischen Interesse sind. Die Patentexperten des FIZ Karlsruhe wurden bereits in der planerischen Phase einbezogen. Für die Relevanz-Feedback Komponente wurde als Grundlage auf ein an der Universität Hildesheim entwickeltes System zurückgegriffen, das bereits mehrfach evaluiert wurde [s. Hackl und Mandl, 2006]. Nach den ersten Entwicklungszyklen wurden immer weiter entwickelte Versionen einem breiteren Publikum und externen Fachleuten vorgestellt und sind auf positive Resonanz gestoßen.

Abbildung 1 zeigt die zweite Version des Prototypen von PatentAide. Eine Anfrage wurde zunächst über das Anfragefenster links oben in einen speziell entwickelten Preview-Modus überführt. Dieser Modus, das schmale, mittlere Fenster, repräsentiert die Anfrage in einer Baumstruktur und gibt Hinweise auf die Zusammensetzung der Treffermenge. Weiterhin unterstützt er verschiedene Arten der Anfragemodifikation wie z.B. der Erweiterung um Thesauruseinträge. Im weiteren wurde die modifizierte Anfrage abgeschickt und in der resultierenden, gerankten Trefferliste (links in der Mitte) ein Dokument angesehen. Die Dokumentansicht befindet sich in der rechten unteren Hälfte. Es ist anhand der verschiedenen Graustufen zu erahnen, dass die Terme aus der Anfrage, die im Textabschnitt „claims“ zu sehen sind, farblich hervorgehoben sind. Außerdem werden die Vorkommen aller Anfrageterme im Gesamtdokument durch farbige Balken angezeigt. In der rechten unteren Ecke ist eine Studie zur Darstellung der Termgewichte auf einen Blick zu sehen. Das betrachtete Dokument wurde für relevant befunden und entsprechend markiert. Zum einen sind daraufhin die Dokumentgewichte neu berechnet worden. Das Ergebnis dieses Prozesses wird sowohl in einer Spalte der Ergebnisliste wie auch in dem Balkendiagramm links unten dargestellt. Zum anderen wurden Termvorschläge auf Grundlage der Relevanzinformation ermittelt. Diese potentiellen Erweiterungsterme werden in der Tabelle rechts oben aufgeführt. Sie können per Drag&Drop in der gewünschten Position der Baumdarstellung eingefügt werden.

Diese Term-Erweiterung zur Optimierung einer Anfrage ist ein typisches Szenario im Patent-Retrieval. Transparenz und Kontrollierbarkeit werden durch die sofortige Anzeige der Änderungen im Preview-Modus erreicht. Die Treffermenge, auf der gearbeitet wird, wird bis zu der expliziten Anweisung nicht verändert.

Auf diese Weise können Anfragen schrittweise verfeinert und Antwortmengen explorativ erschlossen werden. Durch bereitgestellte Konfigurationsmöglichkeiten kann der Prototyp individuell angepasst werden.

4 Ausblick

Der vorgestellte Stand besitzt die angestrebten Kernfunktionalitäten. In weiterhin iterativen Entwicklungsschritten werden jetzt einige Feinheiten und Optimierungen getestet. Anschließend erfolgt eine Evaluierung mit Patentexperten, in der insbesondere untersucht wird, inwiefern die aufgezeigten Möglichkeiten zur Unterstützung der täglichen Recherchearbeit beitragen.

Literatur

[Hackl u. Mandl, 2006] René Hackl and Thomas Mandl. Bilingual Retrieval Experiments with Social Science Documents. In Accessing Multilingual Information Repositories: 6th Workshop of the Cross-Language Evaluation Forum, CLEF 2005, Vienna, Austria, Revised Selected Papers. Berlin et al.: Springer [Lecture Notes in Computer Science 4022]

[Mendelsohn, 2000] Susan Mendelsohn. Patterns formed by a single shot of malt. In *Information World Review*, Juni 2000.

[Poynder, 2000] Richard Poynder. Web-challenged. In *Information World Review*, Juli/August 2000.

[Weckend u. Wurzer, 2004] Edwin Weckend and Alexander Wurzer: Patentrecherche als Produktionsprozess: Elemente eines Qualitätsmanagements. In *PATINFO 2004*, 2004, S. 151-176

[Womser–Hacker, 1997] Womser–Hacker, Christa: *Das MIMOR-Modell. Mehrfachindexierung zur dynamischen Methoden-Objekt-Relationierung im Information Retrieval*. Habilitationsschrift. Universität Regensburg, Informationswissenschaft, 1997.

Workshop on Knowledge and Experience Management

Alexandre Hanft & Martin Schaaf

Objectives

The workshop on Knowledge and Experience Management, which is held as part of the LWA Workshop series, is organized by the German Computer Science Society's (GI) Special Interest Group on Knowledge Management (GI-Fachgruppe Wissensmanagement, FGWM, http://www.fgwm.de). The Special Interest Group on Knowledge Management addresses computer science methods for capturing, development, utilization, and maintenance of knowledge for organizations and networks of people. The focus is on scientific foundations and practical applications.
The workshop aims to provide an interdisciplinary forum for researchers and practitioners for exchanging ideas and innovative applications around knowledge and experience management. Hence, we encouraged submissions describing ongoing research efforts as well as demonstrations of recent research software prototypes. The topics of interest for this workshop are:

- Applications of Knowledge and Experience Management (Corporate Memories, E-Commerce, WWW, Design, Tutoring/eLearning, Robotics, Medicine, etc.)
- "Lessons-Learned" for IT-based Knowledge Management solutions
- Integration of Knowledge Management and Business Processes
- Agile approaches to Knowledge Management
- Peer-2-Peer Technologies for Knowledge Management
- Knowledge Representation (Ontologies, Similarity, Retrieval, Adaptation knowledge, etc.)
- Authoring and Maintenance Support Systems
- Just-In-Time Retrieval and Just-In-Time Knowledge Capturing
- Methods for Knowledge and Experience Retrieval (Case-Based Reasoning, Logic-based approaches, Text-based approaches etc.)
- General Methods of Knowledge Management

Papers

The present workshop proceedings volume includes a total of seven papers reporting on basic and applied research. Each paper has been reviewed by at least two members of the program committee.

Acknowledgements

We would like to thank all authors for submitting papers and the members of the program committee for their support during the organization of the workshop and for reviewing the papers within a very short period of time.

Program Committee

- Klaus-Dieter Althoff, Universität Hildesheim
- Ralph Bergmann, Universität Trier
- Harald Holz, DFKI GmbH
- Ioannis Iglezakis, DaimlerChrysler AG
- Mirjam Minor, Universität Trier
- Markus Nick, Fraunhofer IESE
- Ulrich Reimer, Universität Konstanz
- Bodo Rieger, Universität Osnabrück
- Thomas Roth-Berghofer, DFKI GmbH
- Rainer Schmidt, Universität Rostock
- Gerhard Schwabe, Universität Zürich
- Steffen Staab, Universität Koblenz-Landau
- Ljiljana Stojanovic, FZI an der Universität Karlsruhe
- York Sure, AIFB Universität Karlsruhe

Alexandre Hanft & Martin Schaaf
Hildesheim, October 2006

An Ontology-Driven Management of Change*

Normen Müller
International University Bremen
D-28717, Bremen, Germany
n.mueller@iu-bremen.de

Abstract

Current document management systems (DMS) are designed to coordinate the collaborative creation and maintenance process of documents through the provision of a centralized repository. The focus is primarily on managing documents themselves. Relations between and within documents and effects of changes are largely neglected. To avoid inefficiencies, conflicts, and delays the support of modification management is indispensable.

Here I present the design of the LOCUTOR system that aims to provide management of change functionality for arbitrary XML documents ranging from *informal*, e.g. instruction or construction manuals, to *formal* documents.

1 Introduction

We live in the information age: Huge amounts of information are available at our fingertips and computers influence every aspect in life. In particular we have to deal with e-documents everywhere. *Document engineering*,

> is the computer science discipline that investigates systems for documents in any form and in all media. As with the relationship between software engineering and software, document engineering is concerned with principles, tools and processes that improve our ability to create, manage, and maintain documents [DocEng, 2006].

Of this broad field only small parts have found their way into practice, e.g. *document management systems* (DMS). Current DMS are designed to coordinate the collaborative creation and maintenance process of documents through the provision of a centralized repository. The focus is primarily on managing documents themselves. Relations between and within documents as well as effect of changes on these relations are largely neglected, although information reuse and distribution could seriously benefit from such a relation management. Therefore human reviewers are needed for *management of change* (MOC), i.e., to maintain consistency after modifications. A costly, tedious, and error-prone factor in document life-cycles that is often neglected to cut cost leading to sub-optimal and often disastrous results.

*Research Proposal of a PhD thesis

1.1 A Running Example

To sharpen our intuition about the issues involved let us consider the following situation: Immanuel — a coauthor of a technical report $\mathcal{R}$ — is responsible for some sections therein. He starts writing with some fundamentals $\boxed{1}$ and then builds on that: $\boxed{2} \rightarrow \boxed{1} \leftarrow \boxed{3}$. To enable other authors and interested parties to review and reuse his work he commits $\mathcal{R}$ to a shared DMS. Andrea — a division leader, reporting the work of her group to a client — accesses the DMS and obtains a working-copy of $\mathcal{R}$. She decides to set up some slides $\mathcal{S}$ based on Immanuel's parts of $\mathcal{R}$ in a different order. After a while Immanuel's coauthor Michael checks out the current version of $\mathcal{R}$. He notices some discrepancies within $\boxed{1}$, modifies it to his satisfaction yielding $\boxed{1}$, and commits his revision back to the DMS.

In current DMS this is were the story ends and the problems start:

P1 How to decide whether the modifications of $\boxed{1}$ conflict with the unchanged $\boxed{2}$ and $\boxed{3}$? So do Michael or Immanuel also have to modify $\boxed{2}$ and $\boxed{3}$?

P2 How to decide what sort of modifications Michael performed, i.e., did he modify the meaning, the layout, or did he just correct some typos?

P3 How to decide whehter Andrea has to be notified so that she does not mis-represent the state of affairs?

P4 How to decide whether Andrea actually does need the modified version of $\boxed{1}$?

Recapitulating the problem:

> *Relations between and within the documents are not represented in current* DMS. (†)

i.e., copies of $\mathcal{R}$ do not display the fact that $\boxed{2}$ and $\boxed{3}$ depend on $\boxed{1}$ and copies of $\mathcal{S}$ do not display the fact that $\mathcal{S}$ uses $\mathcal{R}$ and $\boxed{1}$, $\boxed{2}$, and $\boxed{3}$ in particular.

Thus current DMS do not solve (**P1**) – (**P4**)! Immanuel would have to contact Michael to get detailed information of the applied modifications or he would have to completely re-read $\boxed{1}$ and verify on his own if the modifications are in conflict with $\boxed{2}$ or $\boxed{3}$. So this workflow becomes tedious and error-prone. In particular there is still the open question: Who informs Andrea? Neither Immanuel nor Michael are aware of the fact Andrea is setting up some slides partially based on their technical report. Thus, Andrea has to inform herself, i.e., continuously check the state of $\mathcal{R}$ and verify by herself if, regarding her slides, the applied modifications are significant.

To avoid these inefficiencies, conflicts, and delays, and to emphasize the importance of common information spaces in decentralized working environments the integration of a system support into DMS to manage modifications as well as relations is indispensable.

2 A Structured View of Documents

I use a *structured view of documents* to facilitate MoC, information reuse and consistency. In contrast to file- and line-based systems like the SUBVERSION system [SVN, 2006], I consider documents as structured collections of information units. In this context I define w.l.o.g. a *document* as *a self-contained* XML*-based composition of information units*.

PROBST ET AL. [Probst *et al.*, 1997] posits that to obtain meaning from a single *data* element, e.g. a formula or a quantity, we need another component: We need some *context* for its interpretation (see [Kohlhase and Kohlhase, 2005] for a deeper explanation). That is why "self-contained" is part of the definition.

The reason why I base the definition on XML formats is on the one hand that many standard formats are already available as XML and others can easily be defined via DTD, XML Schema or RELAXNG. On the other hand I want to foster open, structural document formats and leverage context indication in the form of content markup. Furthermore by using XML-based document formats some structural information like information units being a constituting part of another information unit is already straightforward given by the syntax.

This combination of content markup and information units makes it a document by my definition.

The following sections describe how I propose to identify data elements in the notion of information units and how to define non-grammatical relations between them. Based on that I present a two-layered view of documents which I will finally expand to a *two-layered two-dimensional view*.

2.1 Informations Units and Ontological Relations

For *formal* documents like specifications or programs the relations between information units, e.g. routine/subroutine, are quite clear and various structuring operations have been proposed for modularization. Main motivations for modularization have been the sharing of subspecifications within one specification, the reuse of specifications, and the structuring of proof obligations. Furthermore the structure of specifications can also be exploited when the effects of changes are analyzed [Mossakowski *et al.*, 2006]. Therefore some initial research has been conducted on methods and tools [Autexier *et al.*, 2002; Mossakowski, 2005] managing the consistency and change of formal documents. However, all these systems base their MoC on the inherent underlying (formal) mathematical structure of the documents.

For handling *informal* documents the situation is completely different. The grammatical and non-grammatical relations between and within the documents are rather clear to humans, but how to make these machine understandable?

Therefore I propose to use knowledge representation (KR) methods, in particular on the notion of a *system ontology* [Krieg-Brückner *et al.*, 2004b][1]. This is an ontology describing the data model of a system or the representation language the system and its applications are based on independently of their respective syntactical realization. Thus I am not bound to any specific document format but yet able to capture semantic interrelations, e.g. *illustrates*, *refines* or *depends-on*, even between (fragments of) informal documents.

For representing ontologies various artificial languages and notations have been proposed. I use Description Logics (DL), a family of knowledge representation languages that can be used to represent the terminological knowledge of an application domain in a structured and formally well-understood way. A KR system based on DL provides facilities to set up knowledge bases, to reason about their content, and to manipulate them. A knowledge base (KB) comprises two components, the TBOX and the ABOX. The TBOX introduces the terminology, i.e., the vocabulary of an application domain, while the ABOX contains assertions about named individuals in terms of this vocabulary. As statements in the TBOX and in the ABOX can be identified with formulae in first-order logic (FOL)[2] the description language has a model-theoretic semantics — that is an "*account of meaning in which sentences are interpreted in terms of a model of, or abstract formal structure representing, an actual or possible state of the world*" [Matthews, 1997]. Thus a KB is equivalent to a set of axioms in first-order logic and like any other set of axioms, it contains implicit knowledge that can be made explicit through inferences[3].

So in order to maintain consistency within and between documents after modifications, i.e., to reason on changes, I represent a system ontology inside the LOCUTOR system as the TBOX, while the ABOX is dynamically synthesized out of the documents and the information units in particular[4].

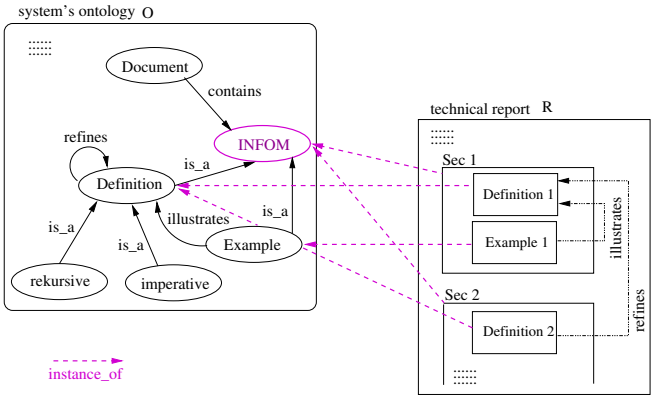


Figure 1: INFOMs and Ontological Relations

To identify information units, I predefine the concept of information units to be part of any (user-defined) system ontology (Figure 1). A concrete elaboration of the term "information unit" is a further part of the research I want to undertake. For the purpose of this article one can pragmatically think of information units as "*tangible/visual text fragments potentially adequate for reuse*" constituting

[1] Called *system's ontology* there.

[2] Note: DL is a decidable fragment of FOL!

[3] Technical term in the DL world: reasoning.

[4] Note, there will be *a* ABOX for each document, so that *the* ABOX constitutes the union of all respective documents.

the content of documents. To distinguish the term "information unit" between common speech and the ontological concept, I will call from now on the ontological concept INFOM[5]. To distinguish between *grammatical* and *non-grammatical*[6] relations, I call the latter *ontological* relations and subsume both by the term *structural* relations.

To clarify the terms INFOM and *ontological* relations let us recall our running example (cf. section 1.1). We presume one of the authors of the technical article $\mathcal{R}$ has established a system ontology $\mathcal{O}$ declaring all concepts and relations of the domain of interest $\mathcal{R}$ is related to, e.g., an ontology describing the concepts of a customer requirements specification. Now, Immanuel does have the ability to "tag" his fragments of $\mathcal{R}$ with concepts of $\mathcal{O}$ (Figure 1). Thus, he is able to explicitly identify information units: 1 is an individual of the concept "definition" Def, 2 is an individual of the concept "example" Ex *illustrating* the first Def, and 3 is also an individual of the concept "definition" Def but *refining* the first one. Note, regarding the pragmatic definition of information units, Immanuel is also able to "tag" grouping elements within $\mathcal{R}$, e.g. sections and paragraphs, by concepts of $\mathcal{O}$.

Thus, by making information units and relations between them explicit, we solved the former problem (†)[7].

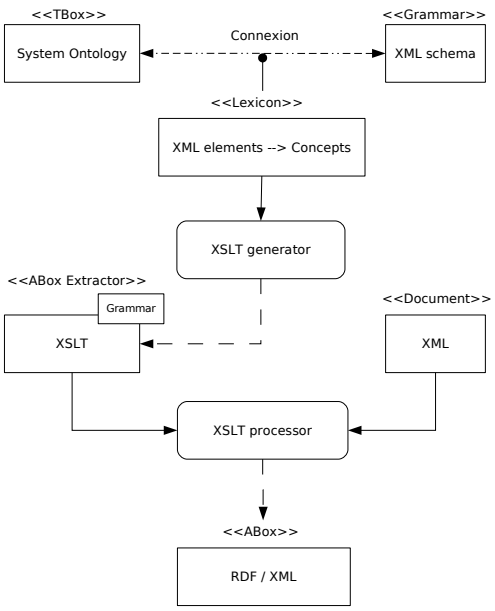


Figure 2: Connexion between concepts and XML elements

The open question is how to design the connexion between a system ontology and respective XML documents, i.e., how to dynamically synthesize an ABOX out of a document. In order to leave the documents untouched, I suggest a *stand-off markup*. Markup is said to be stand-off, or external, when the markup data are placed outside of the text it is meant to tag. The markup therefore points to, rather than wraps, the relevant data content. Therefore I will develop a meta-language to set up a *lexicon* describing the connexion between concepts of a system ontology and XML elements. An XSLT generator then builds up — based on such a lexicon and the respective grammar — an XSL transformation, say an ABOX EXTRACTOR. Finally a ABOX of a respective document is synthesized by an XSLT processor and encoded in RDF/XML. Such a generated RDF/XML document constitutes the stand-off markup. The reason why I propose to use XSLT is that this language is in particular designed to map XML to XML and here to map a XML document markup to RDF/XML respectively.

2.2 Narrative and Content Layer

Following [Verbert and Duval, 2004] and [Kohlhase, 2006] I separate documents into two layers: A *narrative* and a *content* layer both of which consist of INFOMs and are composed via relations. The pictorial representation of the two layers is given in Figure 3.

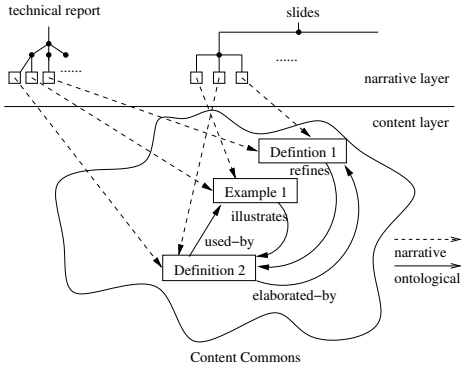


Figure 3: Narrative and Content Layer

The presentational order of information units in documents is represented on the narrative layer whereas the information units themselves and the ontological relations between them are placed in the content layer[8]. The connection between the narrative and the content layer is represented via *narrative* relations (analogous to symbolic links in UNIX). The information units and the ontological relations build up the "content commons" [CNX, 2006]. Thus we clearly separate the conceptual level from the discourse presentation level.

Figure 4 consolidates the classes of relations we defined so far. Structural relations $\mathcal{SR}$ subsume grammatical $\mathcal{GR}$ and ontological relations $\mathcal{OR}$. As to the fact a system ontology describes the data model behind the representation format the grammatical relations have to be

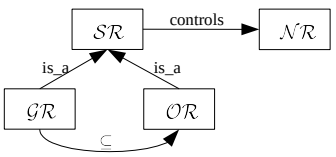


Figure 4: Taxonomy of Relations

[5] A little word-play on "atom". I use the word "atom" in terms of not being further divisible.

[6] Relations defined within a system ontology.

[7] Being aware of the facts that documents and their generation/use are part of socio-technical processes and bridging the knowledge gap between mental models and knowledge representation is a problem the AI community tries to solve for half a century now, I aspire to achieve further insights through my case studies (cf. 4) to set up automated annotation tools to keep authors handcrafted annotation to a minimum.

[8] How far information units could also emerge on the narrative layer is a further research I want to undertake.

a subset of the ontological relations. Narrative relations $\mathcal{NR}$ are controlled by structural relations, i.e., the order of referenced INFOMs is verified. For example, without a previous definition the usage of a technical term within a technical report does not make sense.

To clarify the significance of such a layered view of documents, let us go back to our running example. For simplicity we assume the initially identified information units are derived from the technical report $\mathcal{R}$. Thus Andrea — the author of the slides — does not have to copy these information units but rather just "links"[9] to them. Only the new order of the old information units within the new information product is stored on the narrative layer and narrative relations refer to the respective information units already stored on the content layer.

Note, by assembling information units and respective structural relations we build up the foundations for a interdisciplinary information pool, i.e., pooling of information units related to various domains of interest. Therefore in further research I will also investigate how to compose documents of heterogeneous[10] INFOMs to provide information harvesting at a highest level.

So up to now we have reached a *two-layered view of documents* but have neglected the *ontological* relations between the identified information units so far! Only by using this additional information we will be able to establish a consistent and expressive management of change, i.e., we will be able to handle dependencies between information units and compute the effects of changes on theses dependencies (cf. section 3). Therefore let us look back on the situation in our scenario where Michael is modifying information unit $\boxed{1}$, say the first $\boxed{\text{Def}}$. Now he is aware of the interrelations between the different parts of $\mathcal{R}$, in particular LOCUTOR will notify him about the fact that $\boxed{2}$ and $\boxed{3}$ depend on $\boxed{1}$. Furthermore, by recognizing the narrative relations, LOCUTOR can also notify Andrea about the modifications (**P3**). We will discuss how to solve (**P1**), (**P2**) and (**P4**) in section 3.

2.3 The Concept of Variants

Following initial work in the MMiSS [Krieg-Brückner *et al.*, 2004a] project, in my approach I am also aware of the concept of *variants* [Mahnke and Krieg-Brückner, 2004]. This expands the application area not only "in-the-breadth" but also "in-the-depth". Thus, by extending the well-known concept of *versions* and *revisions* by the concept of variants, the life-cycle of documents will no longer be only along a horizontal time line but also along a vertical line of variants. On the document level I call the concept of versions, revisions, and variants *document states*. I will model the concept of variants by expanding the (default) set of ontological relations by a further one called *variant-of*.

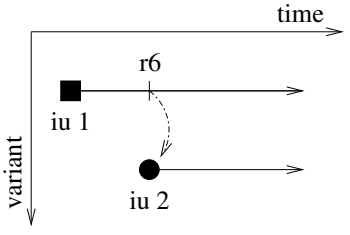


Figure 5: The Variant Dimension

To demonstrate the dimension of variants in a more "dimensional" way Figure 5 depicts another possible scenario: After modifying any information unit iu_1 several times (up to revision number r_6[11]) another user or the initial user herself decides to develop a variant of iu_1. To keep it simple one can imagine iu_2 to be a "language-variant" of iu_1, e.g. iu_1 is written in English and iu_2 in German. By an user annotating information unit iu_2 to be a variant of information unit iu_1 we will be able to build up a complete management of variants, i.e., the states and changes of the original information unit, the variants, and all relations between any of them will be managed as well.

To sharpen the notion of the term *variant* in our running example let us go back to Andrea. Remember she wanted to set up some slides $\mathcal{S}$ regarding $\boxed{\text{Def}}$, $\boxed{\text{Ex}}$, and $\boxed{\text{Def}}$ from the technical report $\mathcal{R}$. However, in general slides

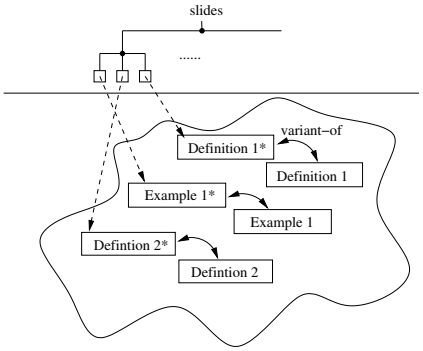


Figure 6: Variants of Infoms

represent a different, say more compact presentation of information. So Andrea will not use the INFOMs one-to-one, but rather modify them to "fit" her presentation. Figure 6 demonstrates the described situation[12]. Andrea is now able to characterize her new information units and the relations between $\mathcal{S}$ and $\mathcal{R}$ still hold.

Based on the arising complex network between documents and information units, respectively, I also propose to integrate value-added services into LOCUTOR. E.g. one of them identifies most referenced INFOMs to capture "useful" and "valuable" information units. Thus I recognize a further open research question: How to enable authors to search the content commons[13], i.e., how to handle the following scenario: Let there be an article $\mathcal{A}_1$ consisting of INFOMs $\boxed{\Lambda}$ and $\boxed{\Omega}$. Now another author wants to write an article $\mathcal{A}_2$ also using $\boxed{\Lambda}$. How do we assist the second author? Does he have to check out $\mathcal{A}_1$, copy-and-paste $\boxed{\Lambda}$ into $\mathcal{A}_2$ and LOCUTOR will take care to identify that $\boxed{\Lambda}$ is already inside the content commons? And, in particular, how does the author get to know that $\boxed{\Lambda}$ exists, anyway? Therefore I hope the case study (cf. section 4) will uncover authors' requirements.

3 MoC on NarCons

Up to know we have elaborated a structured view of (informal) documents. It appears that this two-layered, two-dimensional view is represented by a graph consisting of

[9] Concretion of "links" between entire documents is a further part of the research I want to undertake.

[10] INFOMs declared in different system ontologies.

[11] Think of the well-known SUBVERSION work-flow.

[12] I omit further ontological links for a better readability.

[13] Here I will consider results achieved in the case-based reasoning community.

a narrative layer and a content layer to be called NARCON here. Thereby we have already facilitated information reuse.

Now, I will describe first ideas towards a management of change on NARCONs to achieve *consistent* information reuse, i.e., a MOC on NARCONs to maintain consistency during the development of various document states. Thus this section is less a report on solutions than an attempt to publicize first suggestions towards a consistent management of change. Figure 7 depicts a survey of the proposed MOC system.

Documents will have to identify the underlying language $\mathcal{L} := \langle \mathcal{M}, \mathcal{O} \rangle$ they are an instance of: I will regard a language to be a pair consisting of a model $\mathcal{M} = (\mathcal{G}, \mathcal{E})$ and a system ontology $\mathcal{O}$. A model $\mathcal{M}$ consists of a grammar $\mathcal{G}$ together with an equality theory $\mathcal{E}$. $\mathcal{G}$ defines the syntactical rules to build up valid document and following the initial work in [Eberhardt and Kohlhase, 2004] $\mathcal{E}$ defines when two NARCONs are considered to be equal. Thus the system will use the model $\mathcal{M}$ to compute structural differences Δ between two document states (cf. section 3.1). Following the MMISS project [Krieg-Brückner *et al.*, 2004a] I will use a system ontology $\mathcal{O}$ to describe further semantic dependencies (cf. section 2.1). The LOCUTOR system will use this additional information to compute long-range effects of changes (cf. section 3.2). Furthermore to operate on representatives rather than on singletons I propose a taxonomy of change relations $\mathcal{CR}$ (cf. section 3.2) to enable authors to classify Δ. So to systematically reason on such a classified Δ (cf. section 3.2), say to compute the structural semantic[14] closure ($\mathcal{SSC}$) ∇ of each classified $\delta \in \Delta$ I will develop inference rules consolidated in a change relation calculus $\mathcal{CRC}$. Subject to an $\alpha \in \mathcal{CR}$ and a $\delta \in \Delta$ the $\mathcal{SSC}$ of an information unit iu regarding $\delta : \alpha$ denotes all further information units affected by the modifications w.r.t. their structural relations to iu.

In particular I want to bring into light that annotating is rewarded by getting even more automatic assistance in the future:

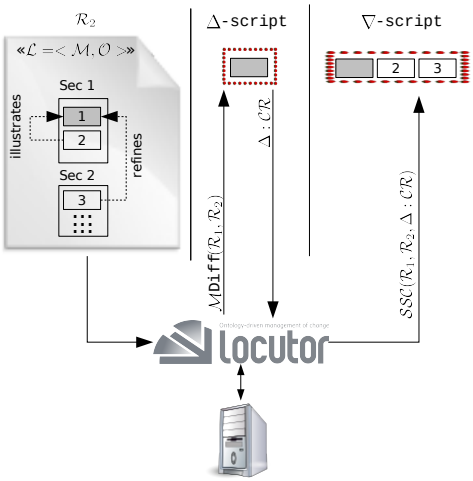


Figure 7: Management of Change

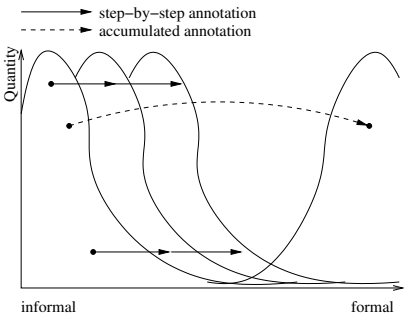


Figure 8: The Shifting Wave

> "*The flatter a document the less the assistance!*"

Figure 8, called the "The Shifting Wave", depicts this slogan. In my approach I want to lead authors on the one hand to annotate informal documents step-by-step, i.e., to provide informal documents more and more with structural semantics and on the other hand to annotate their modifications. As a consequence of each single step the wave shifts a little bit more towards the formal world and thus can be better kept under control by formal systems, i.e., the computation of long-range effects is improved. But note, I do not want to ask too much of authors all at once! It is up to an author to which level she will annotate her changes.

3.1 Computation of Structural Differences

I propose to base the computation of structural differences on the insights of XML-`diff` tools and the initial work of [Eberhardt and Kohlhase, 2004]. According to this I will transform `diff`–algorithms and unification-based techniques, proposed there, to operate on NARCONs.

The first suggestion for such a computation of structural differences is to define a function $\mathcal{M}\texttt{Diff} : \mathcal{D} \times \mathcal{D} \rightarrow \Delta$, where $\mathcal{D}$ denotes NARCON-graphs and Δ a `diff`-script comprising structural differences between NARCON-Graphs.

With "$\mathcal{M}$" in the function name I want to stress to model a strong semantic notion of equality to generate more compact and less intrusive edit scripts. For instance, if we know that whitespace carries no meaning in a document format, two documents are considered equal, even if they differ (with respect to the distribution of whitespace characters) in every single line; as a consequence, Δ would be empty. This motivates the following general statement of the problem at hand [Eberhardt and Kohlhase, 2004]:

> **The General Difference Computation Problem (DCP)**: Let $\mathcal{K}$ be a class of NARCONs and an equality theory $\mathcal{E}$ on $\mathcal{K}$. Given two NARCONs $\mathcal{S}$ and $\mathcal{T}$, find an optimal edit-script that transforms $\mathcal{S}$ to $\mathcal{T}$.

In particular I will engage myself in the general DCP modulo an equality theory ($\mathcal{E}$-DCP) left unsolved in [Eberhardt and Kohlhase, 2004].

To exemplify the functionality of $\mathcal{M}\texttt{Diff}$ let us go back to our running example. If we apply $\mathcal{M}\texttt{Diff}$ on $\mathcal{R}$ after the modifications initiated by Michael the output of $\mathcal{M}\texttt{Diff}(\mathcal{R}_1, \mathcal{R}_2)$ would be $\Delta = \{$ [redacted] $\}$.

[14] I use the term "structural semantics" in sense of marking-up the meaning by structure, i.e., the meaning of an information unit is obtained by its relations to other information units. I do not need any other entailment relation to model semantics but rather concentrate just on the structure.

Up to this stage I want to point out that I did not use any ontology-based information[15], but only operate on properties defined in $\mathcal{M}$. Furthermore I want to stress that I will not handle information units in terms of a "black box", but consider changes within the inner structure as well as in the content, e.g. modifications on the actual text of Def. So one could say, that we have achieved a NARCON-based variant of SUBVERSION so far.

But let us now consider a situation where Michael modified the meaning of Def. The output of $\mathcal{M}$Diff would be the same, omitting Ex and the second Def, which is correct but unsatisfying.

In the next section I will explain how I propose to extend Δ to also capture the structural semantic closure of structural differences.

3.2 Computation of Long-Range Effects of Changes

By regarding *all* relations in general and the ontological relations in particular the system will be able to compute long-range effects of changes and give authors significant feedback of the impact of their modifications. That is, identifying exactly *when*, *where*, *why*, and *by what* updates could corrupt documents w.r.t. the structural relations. Thus not only the directly affected information unit is reported to the user, but all structural related ones as well w.r.t. the classification of Δ.

A Taxonomy of Change Relations

In order to be able to reason on changes, say to reason on Δ, I will develop a taxonomy of *change relations* $\mathcal{CR}$ to classify structural changes. As to the matter of fact the implementation of an automatism to classify structural changes is "AI-hard" I will enable authors to annotate Δ with $\mathcal{CR}$ (short: $\Delta : \mathcal{CR}$). Note, by this additional information about structural changes we solve (**P2**)! So we extend the *two-valued states* of changes, i.e., modified and non-modified, to *annotated two-valued states* of changes. To clarify the notion of a $\mathcal{CR}$-taxonomy I demonstrate a "first-try-example" (see Figure 9) for a toy example.

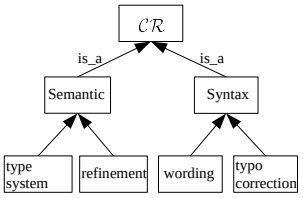


Figure 9: A $\mathcal{CR}$-taxonomy

To demonstrate the emphasis of classified change relations, let us recall our running example, again especially regarding Michael: He modified the first Def without being aware of the fact that other information units depend on this one. We already solved this problem with the new view of documents and NARCON-graphs, respectively. However, so far we are only able to notify Michael and Immanuel about the fact that there are some dependencies, rather than to notify them about the effects of Michael's modifications on these dependencies. So if Michael now classifies his modifications to be syntactical, e.g. typo corrections, the system will compute and fix these changes with respect to the structural relations defined in $\mathcal{L}$, i.e., the system will merge the typo corrections into the next document state of existing working-copies just like in the SUBVERSION approach. If, however, Michael classifies his changes to be semantical, e.g. if he changed the entire type system of the first Def, the situation to fix such a modification changed! So in order to compute and manage the long-range effects of (semantic) changes I will elaborate a system for *reasoning on classified structural changes*.

Reasoning on Classified Structural Differences

To systematically reason on annotated changes, say to reason on $\Delta : \mathcal{CR}$, I will develop inference rules consolidated in a $\mathcal{CR}$-Calculus ($\mathcal{CRC}$) operating on NARCONs. Regarding the proposed calculus I will build on the $\mathcal{DG}$–calculus operating on development graphs [Hutter, 2000] to evaluate what properties and rules can be adopted for NARCONs. A main aspect in this analysis will be the structural properties of development graphs and the calculus itself. Then, based on the $\mathcal{CRC}$, I propose to deduce the effects of changes on structural relations, i.e., with these "rules of re-action to changes" at hand I will define an algorithm to compute for each $\Delta : \mathcal{CR}$ (short: $\ddot{\Delta}$) the structural semantic closure ($\mathcal{SSC}$) ∇, that is, all information units structurally related to the ones explicitly affected by Δ. Therefore I propose another function with the following signature:

$$\mathcal{SSC} : \mathcal{D} \times \mathcal{D} \times \ddot{\Delta} \rightarrow \nabla$$

Here ∇ extends Δ in the sense of $\nabla := \Delta \cup \{(iu, \texttt{trace}(iu)) | iu \in \mathcal{IU}_{\mathcal{O}}\}$, where $\mathcal{IU}_{\mathcal{O}}$ denotes the set of semantically affected INFOMs and $\texttt{trace}(iu)$ represents the path of involved ontological relations.

To clarify the functionality of the suggested $\mathcal{SSC}$ function, let us again take our running example into account but now let us assume Michael changed the meaning of the first Def, e.g. he classifies his changes to be a modification to the type system of Def denoted by the $\mathcal{CR}$ concept $\mathcal{TS}$. So $\mathcal{SSC}$ would compute

$$\mathcal{SSC}(\mathcal{R}_1, \mathcal{R}_2, \boxed{} : \mathcal{TS}) =$$

$$\{\boxed{}, (\boxed{\text{Ex}}, \textit{illustrates}), (\boxed{\text{Def}}, \textit{refines})\}$$

So we finally solved (**P1**) and (**P4**) and are able to give answers to the until now outstanding question "How does one Δ affect existing relations and how do existing relations affect the computation of ∇, respectively?"

As can seen from the illustrative running example the "great challenge" of my thesis is

- to define *ontological relations* for MOC, e.g. a possible additional relation might be *adapted-analogously*, to facilitate authors to augment their informal documents by more *structural semantics*
- to define proper *change relations* to "characterize" modifications
- to define a calculus parameterized by *classified change relations* operating on NARCONs

in order to compute how changes will be reflected onto the pool of information units of composing documents. I hope the result will improve consistent information reuse and distribution.

4 Case Study

I will undertake three case studies to evaluate applicability of my proposed system:

[15] If one wants to involve ontologies at this stage this would correspond to the creation of an ontology $\mathcal{O}$ with just a concept "document" "is_a"-related to the concept *infom*.

The Lecture Study A "NARCON-like" approach has already been successfully used within the sTEX project [Kohlhase, 2005] to enable authors to add semantic information to documents without changing the visual appearance. A large corpus of slides for the lecture General Computer Science I & II at International University Bremen have been marked up by my supervisor MICHAEL KOHLHASE using sTEX. But the project currently lacks any management of change! So this gives me a great ability to test my suggestions on a large amount of data.

The e-Learning Study The Connexions e-Learning system is a rapidly growing collection of free scholarly materials and a powerful set of free software tools to help *authors* publish and collaborate, *instructors* rapidly build and share custom courses, and *learners* explore the links among concepts, courses, and disciplines [CNX, 2006]. As a matter of fact that during my thesis I am sponsored by the EU-project ONCE-CS [ONCE-CS, 2005] to integrate OMDOC [Kohlhase, 2006] into the Connexions projects. Besides integrating my MOC into the system, I will add more structural semantics to the corpus of this projects via the OMDOC system ontology to improve the links among concepts, courses, and disciplines.

The Wiki Study SWIM [Lange and Kohlhase, 2006] is a semantic wiki for collaboratively building, editing and browsing a mathematical knowledge base. Its pages, containing mathematical theories, are stored in OMDOC format. This project is currently being developed by CHRISTOPH LANGE for his master thesis. CHRISTOPH LANGE is an upcoming Ph. D. student in the KWARC group (http://kwarc.eecs.iu-bremen.de/) and so I hope to benefit from his collaborations and the SWIM user interface on the one hand and to assist his work with my MOC on the other hand.

Acknowledgments

The author would like to thank Dieter Hutter, Michael Kohlhase, and Christoph Lange for stimulating discussions in the early stages of this work. Moreover I want to thank the anonymous reviewers for giving me valuable feedback and comments in the course of the review process. So in my future work I will consider the recommended PhD thesis of Makus Nick [Nick, 2005] and his legitimated complaint on the lack of discussing the complexity of my proposed system I could not answer in this early stage, however.

References

[Autexier *et al.*, 2002] S. Autexier, D. Hutter, T. Mossakowski, and A. Schairer. The Development Graph Manager MAYA, 2002.

[CNX, 2006] CONNEXIONS. Project home page at http://www.cnx.org, seen August 2006.

[DocEng, 2006] The ACM Symposium on Document Engineering. Web site at http://www.documentengineering.org, seen April 2006.

[Eberhardt and Kohlhase, 2004] Frederick Eberhardt and Michael Kohlhase. A Document-Sensitive XML-CVS Client. unpublished KWARC blue notes, 2004.

[Hutter, 2000] Dieter Hutter. Management of Change in Structured Verification. In *Proceedings 15th IEEE International Conference on Automated Software Engineering*, number 2000 in ASE, pages 23–34. IEEE Computer Society, 2000.

[Kohlhase and Kohlhase, 2005] Andrea Kohlhase and Michael Kohlhase. An Exploration in the Space of Mathematical Knowledge. In Michael Kohlhase, editor, *Mathematical Knowledge Management, MKM'05*, number 3863 in LNAI. Springer Verlag, 2005.

[Kohlhase, 2005] Michael Kohlhase. Semantic markup for TEX/LATEX. Manuscript, available at http://kwarc.eecs.iu-bremen.de/software/stex, 2005.

[Kohlhase, 2006] Michael Kohlhase. OMDOC – *An open markup format for mathematical documents [Version 1.2]*. Number 4180 in LNAI. Springer Verlag, 2006.

[Krieg-Brückner *et al.*, 2004a] B. Krieg-Brückner, B. Krämer, D. Basin, J. Siekmann, and M. Wirsing. Multimedia Instruction in Safe and Secure Systems. Abschlussbericht, Universität Bremen, 2004. BMBF project 01NM070, 2001-2004.

[Krieg-Brückner *et al.*, 2004b] Bernd Krieg-Brückner, Arne Lindow, Christoph Lüth, Achim Mahnke, and George Russell. Semantic interrelation of documents via an ontology. In G. Engels and S. Seehusen, editors, *DeLFI 2004, Tagungsband der 2. e-Learning Fachtagung Informatik, 6.-8. September 2004, Paderborn, Germany*, volume P-52 of *Lecture Notes in Informatics*, pages 271–282. Springer-Verlag; D-69121 Heidelberg, Germany; http://www.springer.de, 2004.

[Lange and Kohlhase, 2006] Christoph Lange and Michael Kohlhase. A semantic wiki for mathematical knowledge management. In Max Völkel, Sebastian Schaffert, and Stefan Decker, editors, *Proceedings of the 1st Workshop on Semantic Wikis, European Semantic Web Conference 2006*, Budva, Montenegro, 2006. CEUR Workshop Proceedings. To appear, provisional online version at http://www.eswc2006.org/technologies/usb/proceedings-workshops/eswc2006-workshop-semantic-wikis.pdf.

[Mahnke and Krieg-Brückner, 2004] A. Mahnke and B. Krieg-Brückner. Literate ontology development. In Robert Meersman, Zahir Tari, and Angelo Corsaro et al., editors, *On the Move to Meaningful Internet Systems 2004: OTM 2004 Workshops*, volume 3292 of *Lecture Notes in Computer Science*, pages 753–757. Springer; Berlin; http://www.springer.de, 2004.

[Matthews, 1997] P. H. Matthews. *The Concise Oxford Dictionary of Linguistics.* Oxford University Press, 1997.

[Mossakowski *et al.*, 2006] T. Mossakowski, S. Autexier, and D. Hutter. Development graphs – proof management for structured specifications. *Journal of Logic and Algebraic Programming*, 67(1-2):114–145, 2006.

[Mossakowski, 2005] Till Mossakowski. *Heterogeneous Specification and the Heterogeneous Tool Set*. Habilitation, Universität Bremen, 2005.

[Nick, 2005] Markus Nick. *Experience Maintenance through Closed-Loop Feedback*. PhD thesis, Technische Universität Kaiserslautern, October 2005.

[ONCE-CS, 2005] Open Network of Centres of Excellence in Complex Systems. Web site at http://complexsystems.lri.fr/Portal/tiki-index.php, 2005.

[Probst *et al.*, 1997] G. Probst, St. Raub, and Kai Romhardt. *Wissen managen*. Gabler Verlag, 4 (2003) edition, 1997.

[SVN, 2006] The Subversion Project. Web site at http://subversion.tigris.org/, seen August 2006.

[Verbert and Duval, 2004] Katrien Verbert and Erik Duval. Towards a Global Component Architecture for Learning Objects: A Comparative Analysis of Learning Object Content Models. In *Proceedings of the EDMEDIA 2004 World Conference on Educational Multimedia, Hypermedia and Telecommunications*, pages 202–208, 2004.

Der benutzerorientierte Datenbankentwurf im Anwendungsfeld Car Multimedia

Steffen Weichert
Informationswissenschaft
Universität Hildesheim
D-31141, Hildesheim
weichert@uni-hildesheim.de

Gesine Quint
Informationswissenschaft
Universität Hildesheim
D-31141, Hildesheim
quint@uni-hildesheim.de

Abstract

Im vorliegenden Beitrag werden benutzerpartizipative Verfahren im Rahmen des Datenbankentwurfs für ein Informationssystem vorgestellt.
Dabei wird aufgezeigt, wie Extreme Programming als zentraler Ansatz der agilen Software Entwicklung die synergetische Verflechtung des traditionell technologiebetriebenen Software Engineering (SE) mit benutzerzentrierten Verfahren des User-Centered Design (UCD) ermöglichen kann und welche Mehrwerte sich daraus ergeben.
Da insbesondere die Kommunikation zwischen Systementwicklern und Experten im vorgestellten Projekt einen hohen Stellenwert einnahm, werden entsprechende Vorgehensweisen, aufgetretene Probleme sowie Lösungsansätze in der Anforderungsanalyse diskutiert. Der Einsatz von Interview- und Beobachtungstechniken wird dabei am Beispiel der Erfassung des Car Multimedia Anwendungsfeldes zum Zweck der Daten- und Systemmodellierung verdeutlicht.

Einleitung

Unternehmen verfügen vielfach über heterogene und verstreute Datenbestände. Selten finden alle Mitarbeiter die für sie relevanten Informationen in einer für ihre Bedarfe aufbereiteten Art. Noch immer stecken viele wertvolle Informationen in Aktenordnern oder in Aufzeichnungen einzelner Mitarbeiter. Die Einführung eines datenbankbasierten Informationssystems kann dabei unterstützen, unterschiedlichen Benutzergruppen einen für sie optimierten Zugriff auf ausgewählte Informationen zu geben. Voraussetzung für ein solches Informationssystem ist ein Modell, dessen Einheitlichkeit und Vollständigkeit nur unter Einbeziehung der Benutzer garantiert werden kann.
Dieser Beitrag stellt geeignete Methoden am Beispiel eines realen Projekts vor und illustriert deren Einsatz in der Praxis. Die Verfasser vertreten die Meinung, dass es die konsequente Benutzerorientierung von Beginn an war, die das vorgestellte Projekt zu einem Erfolg werden ließen und dass es vor allem die unterschiedlichen Methoden der direkten Kommunikation mit Benutzern sind, die besser bedienbare Informationssysteme entstehen lassen.

SE und UCD im Projekt EIKON

Im Rahmen des Kooperationsprojekts EIKON (Einbaukonfigurationssystem) zwischen der Blaupunkt GmbH und dem Fachbereich Informations- und Kommunikationswissenschaften der Universität Hildesheim wurde ein komplexes web- und datenbankbasiertes Informationssystem entwickelt, das es Benutzern ermöglicht, sich anhand von Fahrzeugangaben (z.B. Hersteller, Modell etc.) über einbaubare Car Multimedia Produkte (Lautsprecher, Navigationsgeräte, Adapterkabel etc.) und deren Kompatibilitäten zu informieren. Das insgesamt zweijährige Kooperationsprojekt wurde im Jahr 2005 vorerst abgeschlossen. Die Anforderungsanalyse war zu diesem Zeitpunkt beendet und die iterative Systementwicklung mit projektbegleitender Evaluation brachte ein Gesamtsystem hervor, von dem nach wie vor große Teile von der Blaupunkt GmbH erfolgreich eingesetzt werden.
Für die Anforderungsanalyse im Datenmodellierungsprozess wurde ein Vorgehen gewählt, das eine enge Zusammenarbeit zwischen dem Entwicklungsteam und den Fachexperten des Car Multimedia Anwendungsfeldes als den zukünftigen Benutzern erlaubte.
Obwohl eine Verflechtung von traditionell technologiegetriebenen Software Engineering Methoden mit benutzerzentrierten Methoden des Usability Engineering seit Jahren propagiert wird (vgl. [Sharp et al. 2006:32], [Faulkner & Culwin 2000:61]), scheint die Übertragung in die Praxis nach wie vor problematisch. Dies liegt daran, dass sowohl in der Ausbildung (vgl. [Faulkner & Culwin 2000]) als auch bei der Verankerung von HCI-Experten in Betrieben wenig Wert auf die Verschmelzung der getrennt gewachsenen Disziplinen gelegt wird und stattdessen benutzerzentrierte Ansätze eher als „Add-On“ oder sogar fälschlicherweise als Gegenpol zum Software Engineering betrachtet werden, die nur im Interface-Design eine Rolle spielen.
Noch immer werden UCD-Methoden vom Software Entwicklungsprozess losgelöst angewandt (vgl. [Juristo & Ferre 2006:1079]). Begriffe wie „Design“ und „user interface“ scheinen zu implizieren, dass benutzerzentrierte Methoden erst dann zum Einsatz kommen, wenn es um die Dekoration des eigentlichen Systems durch eine Benutzerschnittstelle geht (vgl. [Seffah & Metzker 2004:73]). Statt eines integrativen Ansatzes, bei dem aus dem vollen Methodenumfang geschöpft wird, werden von Softwareentwicklern häufig nur ausgewählte Konzepte des UCD - wie Anforderungsanalyse oder Usability Testing - übernommen.
Im vorgestellten Projekt wurde ein Vorgehen gewählt, das Benutzer kontinuierlich in jeder Projektstufe einbindet. Eine solche Verschmelzung von Experten als den zukünftigen Benutzern und Entwicklern zu einer funktionalen Einheit entspricht der Idee des Extreme Programming (vgl. Beck 2003). Diese populäre Methode aus der agilen Softwareentwicklung wurde unter anderem aus folgenden Gründen eingesetzt:

- Die Anforderungen an das zu entwickelnde System waren der beteiligten Abteilung der Blaupunkt GmbH nicht von Anfang an und in vollem Umfang bewusst: „Wir möchten irgendeine datenbankbasierte Lösung für unseren Katalog“ lautete etwa die generelle Tendenz zu Projektbeginn.
- Anforderungen sowie Prioritäten unterlagen im Projektverlauf einem raschen Wechsel: Insbesondere in frühen Projektphasen kamen ständig neue oder veränderte Anforderungen an das System hinzu.
- Ein kleines dreiköpfiges Projektteam ermöglichte die Praktik des Pair Programming (vgl. [Williams & Kessler 2002]): Auf eine klassische Rollenverteilung wurde zugunsten einer Aufgabenverteilung nach Bedarf und Fähigkeiten verzichtet.
- Der Wunsch, am Entwicklungsprozess so weit wie möglich beteiligt zu werden, war bei der Benutzergruppe ausgeprägt. Durch wöchentliche Treffen wurde darauf geachtet, dass Benutzer und Entwickler auf dem gleichen Informationsstand blieben und die Benutzer in allen Projektphasen beteiligt wurden.
- Eine offene Kommunikationskultur herrschte sowohl im Entwicklerteam als auch in der Abteilung der Fachexperten bzw. zukünftigen Systembenutzer. So oft wie möglich wurde Face-to-Face Kommunikation zur Informationsvermittlung verwendet.

Es ist diese auch von [Beck et al. 2001] im Agile Manifesto postulierte Interaktion im Entwicklerteam sowie zwischen Entwicklerteam und Auftraggebern bzw. Benutzern, die nach Meinung der Verfasser die Brücke zwischen UCD und Software Engineering schlägt.
Spielte die Benutzerorientierung auch im Rahmen des User Interface Design (vgl. [Quint 2003]) eine entscheidende Rolle, so soll es im Folgenden vor allem um die Vorzüge der frühen Einbeziehung von zukünftigen Benut zern im Vorfeld und Verlauf der Datenmodellierung gehen.

Ausgangssituation

Alle für die Modellierung relevanten Daten werden bei einer regelmäßig stattfindenden Fahrzeuguntersuchung im Production Center der Blaupunkt GmbH erhoben. Zu diesen Daten gehören Angaben über verschiedene Einbauschächte, die Fahrzeugelektrik bis hin zu Pinbelegungen an bestimmten Kabeln sowie die entsprechenden Produktempfehlungen. Vor der Einführung des datenbankbasierten Informationssystems wurden die Ergebnisse der Untersuchung handschriftlich festgehalten, in Word-Dokumente und Excel-Tabellen übertragen und im PDF-Format im Internet oder in einem Katalog veröffentlicht (vgl. Abbildung 1).

Abbildung 1: Datenverarbeitung vor EIKON

Als Ergebnis der Datenmodellierung entstand schließlich ein umfangreiches Informationssystem mit verschiedenen Schnittstellen zu diversen Benutzergruppen. Abbildung 2 gibt einen Überblick über dieses Gesamtsystem.
Wie die Abbildung zeigt, können alle erhobenen Daten über ein Data Management System komfortabel in die Datenbank eingepflegt werden. Es existieren außerdem verschiedene Benutzeroberflächen, die jeweils an die speziellen Informationsbedarfe der identifizierten Benutzergruppen angepasst wurden. Voraussetzung für das Datenbanksystem, welches von unterschiedlichen Abteilungen mit unterschiedlichen Herangehensweisen an die heterogene Datenbasis benutzt wird, war eine umfangreiche Phase des Datenbankentwurfs.
Im Folgenden wird schwerpunktmäßig auf Vorgehensweisen, Probleme und Ergebnisse bei der Kommunikation zwischen Systementwicklern und Experten eingegangen.

Benutzerorientierte Anforderungsanalyse

Während der Anforderungsanalyse im EIKON-Projekt wurden die Anforderungen aller potenziellen Benutzergruppen an das zu entwickelnde Informationssystem erhoben.
Einen hervorgehobenen Stellenwert nimmt die Anforderungsanalyse im EIKON-Entwicklungsprozess aus folgenden Gründen ein:

- Informationen über Objekte und Beziehungen lagen verteilt, unstrukturiert und zum Teil implizit vor. Ein Großteil der zu modellierenden Zusammenhänge basierte auf Erfahrungswissen der langjährigen Mitarbeiter und war vor Projektbeginn weder standardisiert noch expliziert worden.
- Die Systementwickler waren Laien auf dem Fachgebiet der Fahrzeugelektronik und des Car Multimedia Zubehörs.
- Die im Laufe des Projekts zunehmende Komplexität war anfangs weder den Experten der Blaupunkt GmbH noch den Datenbankentwicklern seitens der Universität bewusst.

Bei der Anforderungsanalyse in Softwareprojekten handelt es sich nach wie vor um die Projektphase mit den größten Herausforderungen an das Entwicklerteam (vgl. [Pekkola et al. 2006]). Allein ein Drittel aller Projekte werden aufgrund mangelnden Anforderungsmanagements ergebnislos abgebrochen (vgl.[Schienmann 2002:9]). Durch suboptimale Analyse entstehende Fehlerbehebungskosten rechtfertigen sogar die Forderung nach speziell ausgebildeten Personen mit Schnittstellenkompetenz und Kommunikationsvermögen (vgl. [Kraut/Streeter 1991 zitiert nach Rauterberg 1992:116]). Die Informationswissenschaft der Universität Hildesheim hat den Anspruch, diesen Bedarf an speziellem Personal zu decken.
Für die Systementwickler war es zum einen nötig, sich in die Lage der zukünftigen Systembenutzer zu versetzen und den fachlichen Hintergrund zu verstehen. Zum anderen stellte gerade diese Leistung auf Grund der Komplexität der Anwendung eine große Herausforderung dar. Deshalb wurde eine enge Zusammenarbeit zwischen den Experten, und den Entwicklern als Experten für Informationssysteme für sinnvoll betrachtet.

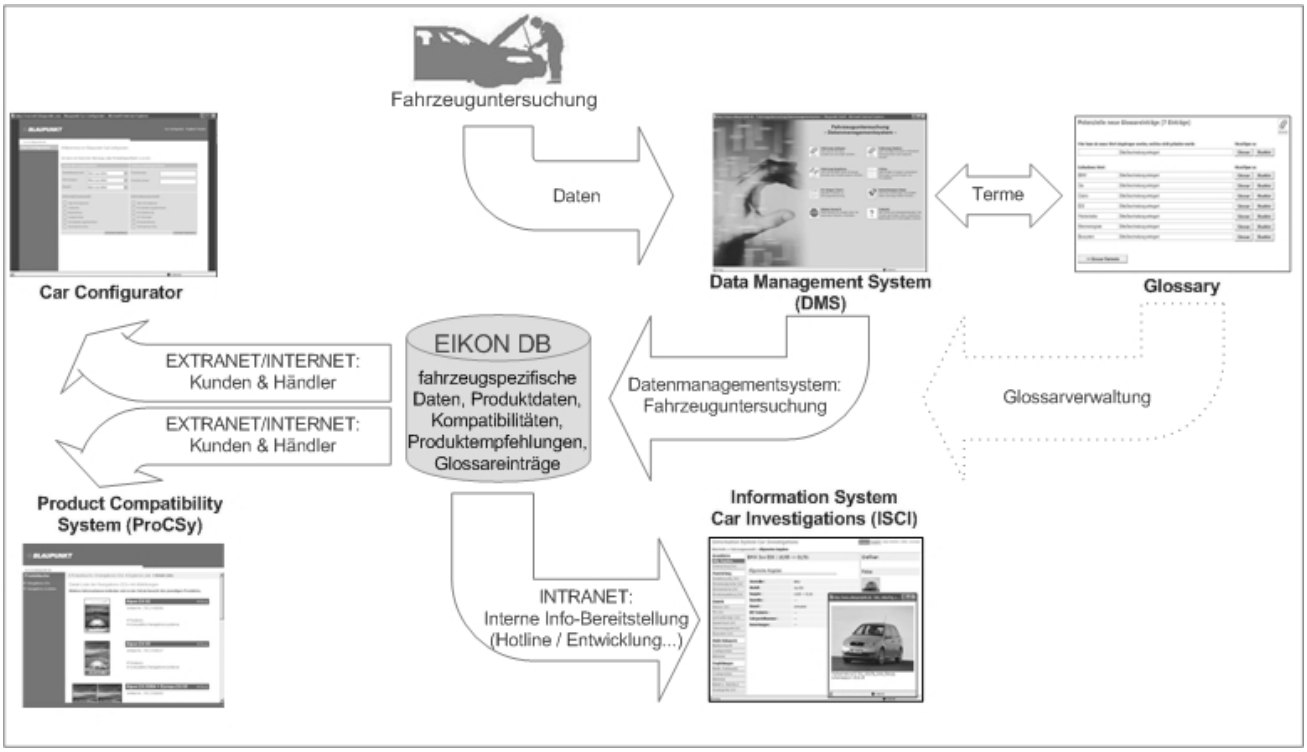


Abbildung 2: Das EIKON-System

Die folgenden Vorteile eines solchen benutzerpartizipativen Ansatzes bei der Systementwicklung gelten für die Datenbankmodellierung im EIKON-Projekt:

Vorteile für die Benutzer:

- Einsicht in und Verständnis für Möglichkeiten und Grenzen neuer Technologien und die Arbeit der Entwickler,
- größere Identifikation mit bzw. höhere Akzeptanz der Lösung,
- höhere Motivation zur Benutzung des Systems
- kürzere Einarbeitungszeit
- Weniger Fehler im Umgang mit dem System.

Vorteile für die Entwickler:

- Feedback zu ihrem Produkt,
- Qualifizierungsmöglichkeiten in Fachfragen,
- bessere Einsicht in die Arbeit der Benutzer,
- größere Sicherheit bei der Lösungsfindung
- Vermeidung von Funktionalitäten, die Benutzer nicht benötigen.

Der letzte Punkt betont auch die Einhaltung des Simplicity Prinzips aus dem Agile Manifesto (vgl. [Beck et al. 2001]): „the art of maximizing the amount of work not done“. Neben den genannten Vorteilen ist die höhere Qualität der Modellierung als übergeordneter Mehrwert der benutzerpartizipativen Vorgehensweise zu sehen.
Wie sich im vorgestellten Projekt zeigte, sorgte insbesondere der hervorgehobene Stellenwert, der der Kommunikation zwischen Entwicklern und Experten als zukünftigen Systembenutzern eingeräumt wurde, für die gewünschten Synergieeffekte aus Software Engineering und UCD. Kommunikation stellt neben Einfachheit, Feedback, Mut und Respekt einen der zentralen Werte des Extreme Programming dar (vgl. [Ambler 2006]). Wie die lessons-learned aus dem EIKON-Projekt zeigen, kann der bewusste Einsatz von kommunikationsstarken Methoden, wie die im Folgenden behandelten, dafür sorgen, dass über die Wissensakquisition hinaus wertvolle Mehrwerte für ein produktives Team entstehen können, bei dem Teamarbeit, Offenheit und stetige Kommunikation im Vordergrund stehen.
Möglichkeiten hierfür sind im Rahmen der Anforderungsanalyse vor allem in den folgenden Aktivitäten zu sehen:

- Dokumentenanalyse
- Interviews
- Beobachtungen

Dokumentenanalyse

Zu den für das Datenmodell relevanten wichtigsten Dokumenten gehörten der Fahrzeuguntersuchungsbogen und der Einbauempfehlungskatalog. Im Folgenden soll detailliert auf diese Dokumente und Schwierigkeiten bei der Analyse eingegangen werden.

Fahrzeuguntersuchungsbogen:

Als eines der wichtigsten Dokumente für die im Datenmodell zu repräsentierenden Entitäten und Relationen stellten die Fahrzeuguntersuchungsbögen hauptsächlich aufgrund fehlender Standardisierung ein erhebliches Problem bei der Wissenserhebung dar. Die auf einer Microsoft Word Vorlage basierenden Dokumente, welche im Anschluss an die Fahrzeuguntersuchung ausgefüllt wurden, sind über mehrere Jahre eingesetzt worden und unterlagen regelmäßigen Änderungen und Ergänzungen. Dies hatte erhebliche Inkonsistenzen zur Folge. Als Beispiel für sol-

che Inkonsistenzen, die erst durch den Vergleich mehrerer Fahrzeuguntersuchungsbögen erkennbar wurden, soll der Bereich Elektronik des Fahrzeuguntersuchungsbogens zweier Fahrzeuge herangezogen werden. Für die Elektronik von Fahrzeug 1 wurden u.a. die folgenden Angaben gemacht:

Abbildung 3: Fahrzeuguntersuchungsdaten Elektronik für Fahrzeug 1 (Ausschnitt) Quelle: Fahrzeuguntersuchungsbogen (Blaupunkt GmbH)

Bei der Untersuchung von Fahrzeug 2, ein Jahr später, wurden hingegen wesentlich mehr Angaben über die Elektronik des Fahrzeugs gemacht:

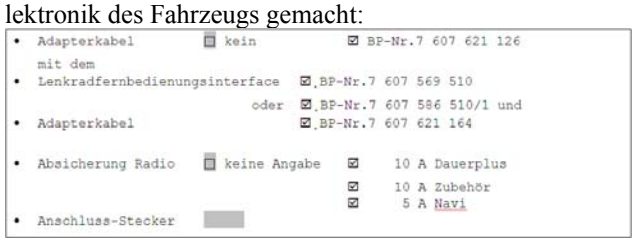


Abbildung 4: Fahrzeuguntersuchungsdaten Elektronik für Fahrzeug 2 (Ausschnitt) Quelle: Fahrzeuguntersuchungsbogen (Blaupunkt)

Wie in Abbildung 3 und 4 deutlich wird, wurde bei der Untersuchung von Fahrzeug 1 lediglich ein Kabel erfasst. Bei der Untersuchung von Fahrzeug 2 hingegen wurden insgesamt fünf Kabel mit entsprechender Blaupunkt-Produktnummer festgehalten. Auch die Anzahl der Angaben für die vorgefundene Radio-Absicherung unterscheiden sich: Während bei Fahrzeug 1 lediglich eine Ampère-Zahl (25 A) angegeben wird, werden bei Fahrzeug 2 insgesamt drei Angaben gemacht und näher spezifiziert.
Hinzu kommt eine zum Teil nur für interne Mitarbeiter verständliche Verwendung von Abkürzungen („Navi“ für „Navigation“ etc.) und Kurzschreibweisen. Beispielsweise ist nicht sofort ersichtlich, dass es sich bei der Angabe der Artikelnummer für ein Lenkradfernbedienungsinterface mit der Schreibweise
BP-Nr: 7 607 586 510/1
bereits um zwei Artikelnummern für Interfaces handelt, nämlich um
BP-Nr. 7 607 586 510 und *BP-Nr. 7 607 586 511.* Derartige Schreibweisen, durch die der Techniker bei der Daten-Eingabe Zeit spart, erschwerten das Verständnis der im Fahrzeuguntersuchungsbogen erfassten Daten erheblich.
Für den Umgang mit Abkürzungen und Inkonsistenzen in der Terminologie wurde gemeinsam mit den Experten und zukünftigen Benutzern eine Liste angelegt und regelmäßig aktualisiert. Begleitend wurde ein abteilungsübergreifender Prozess zur Standardisierung von Fachterminologie bei der Blaupunkt GmbH angestoßen.
Die aufgeführten Problemausschnitte zeigen, dass der Umgang mit dem Fahrzeuguntersuchungsbogen als eine der wichtigsten Quellen extrem schwierig und nur durch begleitende strukturierte und fokussierte Interviews zu einzelnen Bereichen des Bogens möglich war.

Einbauempfehlungskatalog:

Als zweite wichtige Quelle für das Datenmodell stellte auch der Einbauempfehlungskatalog ein zum Teil schwierig zu analysierendes Dokument dar. Die dort in Tabellenform wiedergegebenen Informationen waren kaum verständlich. Die beschränkten Möglichkeiten der Informationsvisualisierung in Tabellenform auf zwei DIN-A4-Seiten eines Katalogs waren somit nicht nur Auslöser für die Projektinitiierung, sondern auch ein Problem bei der Dokumentenanalyse, wie folgendes Beispiel zeigt:

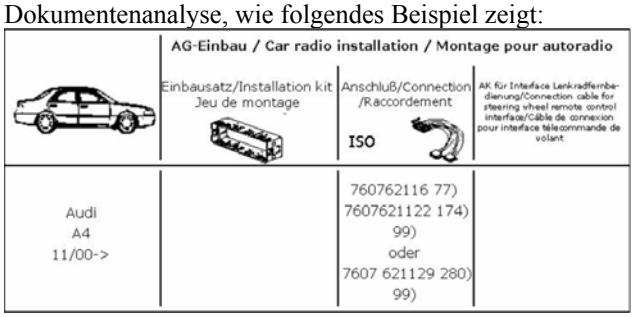


Abbildung 5: Informationsvisualisierung im Einbauempfehlungskatalog: Einbauempfehlung für Radioanschluss-Kabel (Quelle: Selbst erstellt nach Einbauempfehlungskatalog, Blaupunkt GmbH)

Der in Abbildung 5 dargestellte Zusammenhang zeigt eine von zahlreichen Schwierigkeiten beim Umgang mit dem Einbauempfehlungskatalog:
Unter AG-Einbau[1] findet man in der Spalte *Anschluss* im Beispiel drei Kabel[2], die von der Blaupunkt GmbH für den Radioanschluss empfohlen werden sowie verschiedene Einbauhinweis-Nummern[3], die durch eine Legende am Ende des Katalogs erläutert werden. Dadurch, dass die Informationsausgabe im Katalog von der Druckerei auf sechs Zeilen pro Tabellenfeld und eine feste Feldbreite beschränkt ist, stehen einige Hinweisnummern hinter und einige unter der zugehörigen Artikelnummer. Im Beispiel ist nicht erkennbar, ob sich der Einbauhinweis 99) nur auf das unmittelbar vorangehende Kabel oder auf beide darüber stehenden Kabelempfehlungen bezieht. Die zum Teil in den Einbauhinweisen angegebenen Einschränkungen für die Gültigkeit der Empfehlung sowie zusätzliche Kabelempfehlungen für spezielle Vorrüstungen machen die Darstellung noch unverständlicher.
Zur Überprüfung der korrekten Modellierung und zur Überbrückung der semantischen Lücke zwischen Experten und Entwicklern wurden neben Experteninterviews

1 AG = Autoradiogerät.
2 Artikelnummer 7607621116: Adapterkabel für Strom/Masse VW/Audi,
Artikelnummer 7607621122: Adapterkabel für Audi A3-8 mit Aktivlautsprechern,
Artikelnummer 7607621129: Adapterkabel für VW/Audi mit aktiver Antenne Adapterkabel für VW/Audi mit aktiver Antenne.
3 Hinweis 77: „Zum Lautsprecheranschluss, bei Vorrüstung, ggf. Verlängerung verwenden (15cm) 7606647093.“
Hinweis 174: „Adapterkabel für Fahrzeuge mit Aktivlautsprechervorrüstung.“
Hinweis 99: „Für Autoradios ab 80 Watt Ausgangsleistung zusätzlich Kabel 7607884093.“
Hinweis 280: „Aktive Antenne“.

zunächst Papier- und später HTML-Prototypen der geplanten Benutzeroberfläche als „gemeinsame Sprache“ verwendet.
Als Lösung für die problematische Darstellung im Katalog wurde für das User Interface zwar die tabellarische Darstellung beibehalten, jedoch wurden die Information hinter Fußnoten oder Artikelnummern in Popup-Fenstern untergebracht, so dass Abbildungen und Erklärungstexte jederzeit verfügbar sind (vgl. Abb. 6).

Abbildung 6: User Interface mit Produkt-Popup

Aus den aufgeführten Gründen waren auch im Umgang mit dem Empfehlungskatalog zahlreiche begleitende Interviews notwendig, um Unklarheiten mit Hilfe der Fachexperten zu beseitigen. Auf die verschiedenen Interviewtechniken wird im folgenden Kapitel eingegangen.

Interviewtechniken bei der Anforderungsanalyse

Bei dieser Technik wird versucht, durch Befragung das Wissen des Experten zu bestimmten Zusammenhängen zu erheben. Im EIKON-Projekt fanden alle durchgeführten Interviews am Arbeitsplatz des Experten statt, um es dem Befragten zu ermöglichen, auf Dokumente zuzugreifen, die an seinem Arbeitsplatz verfügbar sind. Eine Tonband-Aufzeichnung der Interviews wurde nicht durchgeführt, da eine Aufteilung des dreiköpfigen Projektteams in Protokollanten und Moderator für ausreichend erachtet wurde. Unmittelbar nach der Durchführung der Interviews wurden die Mitschriften der zwei Protokollanten in einer Sitzung ohne Experten diskutiert, um Wahrnehmungsverzerrungen und Fehlinterpretationen zu vermeiden.
Die Protokolle aus Experteninterviews waren die wichtigsten und zugleich unproblematischsten Informationsquellen im Projekt, da sie direkt auf der Basis der Gespräche mit dem Fachexperten erstellt wurden.
Hinsichtlich der Art und des Umfangs von Interviews lassen sich verschiedene Vorgehensweisen unterscheiden. Dieser Artikel folgt der Einteilung nach [Nikolopoulos 1997:102ff.] in folgende drei Interviewklassen:

- Unstrukturierte Interviews
- Strukturierte Interviews
- Fokussierte Interviews

Unstrukturierte Interviews

Unstrukturierte Interviews wurden hauptsächlich zu Beginn des Projekts eingesetzt. Sie ähneln einer normalen Unterhaltung, so dass der Befragte in einer zwanglosen Atmosphäre eine Einführung in das Fachgebiet geben konnte. Mit Hilfe der unstrukturierten Interviews in der Anfangsphase des Projekts konnte ein guter Überblick über das Car Multimedia Fachgebiet gewonnen werden. Neben dem Erlangen eines ersten Einblicks in die Domäne, waren die unstrukturierten Interviews am Anfang auch eine hervorragende Gelegenheit, eine gute Arbeitsbeziehung zwischen Experten und Entwicklerteam aufzubauen und somit „Kommunikation“ als einen zentralen Wert des Extreme Programming zu verinnerlichen.

Strukturierte Interviews

Der wesentliche Unterschied zwischen strukturierten und unstrukturierten Interviews liegt im genauen Ablauf: Während das unstrukturierte Interview in einer freien Form durchgeführt wird, gibt es bei einem strukturierten Interview eine Art Programm, welches Punkt für Punkt abgearbeitet wird.
Nach jedem durchgeführten Interview wurden die Ergebnisse direkt besprochen, analysiert und standardisiert in einem Protokoll festgehalten. In der Regel fand das nächste Interview in einem Abstand von fünf Tagen statt, so dass die Zwischenzeit zur Analyse und später zur Datenmodellierung genutzt werden konnte. Des Weiteren wurde in dieser Zeit ein neuer Leitfaden für das kommende Interview erstellt. In ihn flossen neben bisher nicht ausreichend beantworteten auch neue Fragen ein, die sich im Verlauf der fünf Tage ergaben.
Abbildung 7 verdeutlicht diesen Ablauf.

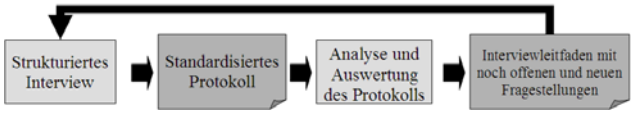


Abbildung 7: Vorgehensweise für strukturierte Interviews im EIKON-Projekt

Strukturierte Interviews dienten vor allem dazu, ein möglichst umfassendes Wissen der gesamten Car Multimedia Domäne zu erlangen. Eine thematische Verzweigung während des Gesprächs war bei dieser Art von Interview sowohl in horizontaler als auch in vertikaler Ebene möglich.

Fokussierte Interviews

Fokussierte Interviews wurden erst zu einem relativ späten Projektzeitpunkt eingesetzt, da sie vor allem dazu dienen, einzelne Konzepte und Zusammenhänge des Anwendungsgebiets zu durchdringen. Im Gegensatz zu strukturierten Interviews sollen sie thematisch „tiefes Wissen“ [Gabriel 1992:212] hervorheben.
Insbesondere für den extrem komplexen Problemkontext des Radio-Einbaus waren fokussierte Interviews eine Hilfe, um Wissen über Relationen zu erlangen. Ein Beispiel für den Ablauf eines solchen Interviews sieht wie folgt aus:

Moderator: „Um alle Daten zu einem spezifischen Fahrzeug in der Datenbank ablegen und wieder aufrufen zu können, benötigen wir eine eindeutige Eigenschaft des Fahrzeugs, welche bei jedem Fahrzeug anders ist. Wäre dafür die Fahrgestellnummer geeignet?“
Experte: „Sie haben Recht, die Fahrgestellnummer ist eindeutig. Wir können sie jedoch nicht verwenden, da sie nicht immer auf den Fahrzeuguntersuchungsbögen erfasst wurde.“
Moderator: „Welche Angaben über ein Auto wären denn nötig, um es eindeutig zu identifizieren?“
Experte: „Ein bei uns untersuchtes Fahrzeug kann jederzeit anhand der Kombination aus Hersteller, Modell und Baujahr identifiziert werden.“
Moderator: „Aber in einigen Fällen fehlt auf den Fahrzeuguntersuchungsbögen auch das Baujahr.“
Experte: „In dem Fall kann man alternativ die Baureihe verwenden. Eine der beiden Angaben ist auf jeden Fall vorhanden.“

Dieses Beispiel-Interview war wichtig um Eigenschaften festzulegen, mit denen ein Fahrzeug eindeutig identifiziert werden kann. Hätte die erste Vermutung der Systementwickler zu einer falschen Modellierung geführt, konnte durch die gezielte Befragung der Experten dieser Fehler vermieden werden. In der Datenbank wird nun jedes Fahrzeug durch die genannten drei Angaben festgelegt und der Primärschlüssel „Fahrzeuguntersuchungsnummer“ (FU_Nr) kann zugewiesen werden.
Wie der Einstieg in das Gespräch zeigt, boten die Interviews eine Möglichkeit für Benutzer und Entwickler, sich gemeinsam neues Wissen anzueignen, aber auch Fachwissen über die jeweilige Domäne („Datenmodellierung“ vs. „Radioeinbau“) auszutauschen und voneinander zu lernen.

Beobachtungstechniken bei der Anforderungsanalyse

Die als Ergänzung zu Interviewtechniken eingesetzten Beobachtungstechniken dienten u.a. dazu, schwer verbalsierbares implizites Wissen zu erheben, aber auch dazu, die durch andere Methoden erhobenen Informationen zu validieren.
Hierzu nahm das Entwicklerteam an mehreren Fahrzeuguntersuchungen im Production Center der Blaupunkt GmbH teil.
Der Umgang mit Ergebnissen der Expertenbeobachtung, die ebenfalls in Protokollen festgehalten wurden, ist problematischer als die Analyse der Protokolle aus Interviews. Eine Schwierigkeit stellte beispielsweise die Tatsache dar, dass bei den beobachteten Fahrzeuguntersuchungen drei Experten anwesend waren, die an unterschiedlichen Orten am Fahrzeug arbeiteten, so dass es zeitweise schwierig war, allen Experten die notwendige Aufmerksamkeit entgegenzubringen. Videoaufnahmen, die in diesem Zusammenhang geholfen hätten, waren nicht erlaubt.
Dennoch konnte durch die direkte Beteiligung an Fahrzeuguntersuchungen wertvolles Terminologiewissen erworben werden und Fragen zur Abfolge von Arbeitsschritten nicht nur beantwortet, sondern auch am konkreten Fahrzeug verdeutlicht werden.
Es werden im Folgenden die durchgeführte Protokollanalyse und die Introspektion vorgestellt.

Protokollanalyse

Beim protokollanalytischen Verfahren wurde der Experte in der Regel mit einem konkreten Problem beauftragt und gebeten, bei der Lösung die so genante Methode des „lauten Denkens“ (vgl. [Schneider 1994:40]) anzuwenden, das heißt, seine Lösungsschritte laut zu kommentieren.
Die Protokollanalyse diente hervorragend dazu, bereits erfasstes Terminologiewissen zu festigen, indem nicht nur Fachtermini in einem konkreten Kontext verwendet wurden, sondern zudem die verschiedenen Produkte gesehen, näher untersucht, zerlegt und somit besser eingeprägt werden konnten.
Des Weiteren wurden fahrzeugspezifische Abhängigkeiten der Produkte direkt an einem Fahrzeug veranschaulicht, indem etwa ein Lautsprecher-Einbauort komplett auseinander gebaut wurde und der Experte auf diese Weise aufzeigen konnte, wann zum Beispiel ein bestimmter Einbausatz notwendig ist, um einen Blaupunkt-Lautsprecher als passend zu empfehlen.

Introspektion

Die Introspektion ist im Fall des EIKON-Projekts als Ergänzung zur Protokollanalyse zu verstehen. Bei dieser Methode arbeitete der Experte nicht direkt an einer konkreten Aufgabe und kommentiert sie, sondern gab Einschätzungen über die Lösungsmöglichkeiten eines Problems an, ohne die Lösungsschritte konkret durchzuführen (vgl. [Schneider 1994:41]).
Eine typische Fragestellung der Introspektion, die bei der Fahrzeuguntersuchung verwendet wurde, ist:
„Und wie würden Sie folgendes Problem lösen?“
Auf diese Art konnte nicht nur die Vorgehensweise bei dem untersuchten Fahrzeug beobachtet werden, sondern auch Problembereiche angeschnitten werden, die nur bei einigen speziellen Fahrzeugen auftreten. So ergab sich etwa folgender Ablauf:

Experte (untersucht die rechte Hintertür eines Fahrzeugs): „Hier haben wir die hintere Tür des Fahrzeugs komplett geöffnet ohne Seitenverkleidung und können den Einbauort der Lautsprecher gut ausmessen. Die Maße trage ich hier in den Fahrzeuguntersuchungsbogen unter ‚Hintertüren’ ein.“
Moderator: „Vermessen Sie auch die linke hintere Tür?“
Experte: „Nein, bis auf einige Ausnahmen sind die Maße der linken und rechten Tür gleich. Es gibt jedoch Ausnahmen, wie zum Beispiel einige Mercedesmodelle, bei denen die vorderen und die hinteren Türen unterschiedlich ausgestattet sind.“
Moderator: „Und wo würden sie diese zusätzlichen Maße im Fahrzeuguntersuchungsbogen vermerken, für die kein Feld vorgesehen ist?“
Experte: „Unter ‚Sonstige Bemerkungen’.“

Dieser Gesprächsablauf wurde als Grundlage für eine Diskussion mit dem Fachexperten darüber genutzt, ob im dem zu entwickelnden System zugrunde liegenden Datenmodell weitere Entitäten für neue Einbauorte wie vordertuer_links geschaffen werden müssen. Bis dahin waren die Türeinbauorte für Lautsprecher lediglich in Vordertür und Hintertür unterteilt. Eine speziellere Einteilung etwa in *vordertuer_links* und *vordertuer_rechts* wurde jedoch letztlich nicht für nötig gehalten, da es sich bei den Ab-

weichungen um sehr seltene Ausnahmefälle handelt, die in Speicherfeldern wie „Sonstige Bemerkungen“ vermerkt werden können.
Zusammenfassend lässt sich zur Expertenbeobachtung sagen, dass sie ein hervorragendes Mittel darstellt, den Bruch zwischen anwendungsbezogener Fachsprache bei den Entwicklern auf der einen und technischer Fachsprache bei den Experten und zukünftigen Systembenutzern auf der anderen Seite zu überwinden. Zu dieser Überwindung tragen laut [Rauterberg 1992: 114] „gemeinsam erlebte, sinnlich erfahrbare Kontexte“ bei. Die Kommunikationsbarriere lasse sich in dem Zusammenhang umso besser überwinden, „[…] je stärker der semantische Kontext des jeweils Anderen erfahrbar wird“ (ebd.).

Fazit

Es konnte gezeigt werden, dass ein benutzerpartizipativer Ansatz in der Anforderungsanalyse für ein datenbankbasiertes Informationssystem eine gute Möglichkeit bietet, insbesondere solche Anforderungen zu erheben, die nicht oder nur ungenügend in einem Pflichtenheft erfasst werden können. Der gezielte Einsatz strukturierter, unstrukturierter und fokussierter Interviews sowie Beobachtungstechniken ermöglicht es, zum Teil implizites Wissen zu externalisieren und in ein entsprechendes Daten- und Systemmodell einfließen zu lassen.
Im vorgestellten Projekt hat sich dabei herausgestellt, dass es nicht ausreicht, gängige Verfahren des SE zur Wissensakquisition in der Anforderungsanalyse anzuwenden. Vielmehr war es die Orientierung an Prinzipien der agilen Softwareentwicklung und des Extreme Programming, die unterstützend auf eine positive Gesprächskultur und die Überwindung von Kommunikationsbarrieren einwirkte.
Benutzer sollten deshalb in Projekten der vorgestellten Art nicht nur in ausgewählten sondern in allen Projektphasen einbezogen werden. Auf diese Weise kann eine hohe Qualität bei der Datenmodellierung garantiert werden und es entsteht ein vollständiges Modell, das alle Benutzerbedarfe befriedigt und eine Grundvoraussetzung für die Erstellung der Benutzerschnittstellen darstellt.
Hohe Kommunikationsdichte im Entwicklungsprozess als ein Wert des Extreme Programming sowie die synergetische Integration von Software Engineering und HCI-Methodik versprechen neben einer erfolgreichen Anforderungsanalyse ein Informationssystem, das sich durch Usability und daraus resultierende hohe Nutzerakzeptanz auszeichnet.
Wie sich in der Retrospektive zeigte, hat insbesondere die Bereitschaft des Entwicklerteams, bei Einbauuntersuchungen teilzunehmen, wesentlich zur offenen Kommunikationskultur und zur Akzeptanz im Gesamtteam beigetragen. Mindestens genau so wichtig wie die Wissensakquisition und das resultierende Verständnis für die tatsächlichen Aufgaben der Experten war somit die neue Erfahrung auf Seiten der Experten und zukünftigen Systembenutzer, dass dem Entwicklerteam sehr an der Einhaltung des ersten Basiskonzepts des Agile Manifesto lag: „Menschen und Zusammenarbeit vor Prozessen und Werkzeugen.“

Literatur und Onlinequellen

[Ambler 2006] Ambler, S.W. (2006): Agile Modeling (AM) Values v2: http://www.agilemodeling.com/values.htm (Validierungsdatum: 23.9.2006)

[Beck 2003] Beck, K. (2003): Extreme Programming. Das Manifest. Addison-Wesley: New York.

[Beck et al. 2001] Beck, K. et al. (2001): Agile Manifesto : http://agilemanifesto.org/ (Validierungsdatum: 23.9.2006)

[Faulkner & Culwin 2000] Faulkner, X. & Culwin, F. (2000): *Enter the usability engineer: integrating HCI and software engineering*. In: ITiCSE '00: Proceedings of the 5th annual SIGCSE/SIGCUE ITiCSEconference on Innovation and technology in computer science education. ACM Press: New York. S. 61-64.

[Gabriel 1992] Gabriel, R. (1992): *Wissensbasierte Systeme in der betrieblichen Praxis*. McGraw-Hill Publishing: New York et al.

[Juristo & Ferre 2006] Juristo, N. & Ferre, X.: *How to integrate usability into the software development process*. In: ICSE '06: Proceeding of the 28th international conference on Software engineering. ACM Press: New York. S. 1079-1080.

[Nikolopoulos 1997] Nikolopoulos, C. (1997): *Expert Systems. Introduction to First and Second Generation and Hybrid Knowledge Based Systems*. Marcel Deckker Inc.: New York/Basel/Hong Kong.

[Pekkola et al. 2006] Pekkola, S., Kaarilahti, N., Pohjola, P. (2006): *Towards formalised end-user participation in information systems development process: bridging the gap between participatory design and ISD methodologies*. In: PDC '06: Proceedings of the ninth conference on Participatory design, ACM Press: New York. S. 21-30

[Quint 2003] Quint, G. (2003): *Benutzerzentriertes Design bei der Implementierung eines web- und datenbankbasierten Konfigurationssystems für die Blaupunkt GmbH*. Magisterarbeit, Universität Hildesheim, Fachbereich III - Informations- und Kommunikationswissenschaften.

[Rauterberg 1992] Rauterberg, M. (1992): Partizipative Modellbildung zur Optimierung der Softwareentwicklung. In: R. Struder (Hrsg.): *Informationssystem und Künstliche Intelligenz*. Proceedings. 2. Workshop Ulm 24.-26. Februar 1992. Springer Berlin et al. S. 113-128.

[Schienmann 2002] Schienmann, B.(2002): *Kontinuierliches Anforderungsmanagement. Prozesse-Techniken-Werkzeuge*. Addison-Wesley: München.

[Schneider 1994] Schneider, E. (1994): *Der Prozess der Wissensakquisition und seine Integration in den Expertensystem-Entwicklungsprozess*. Universität Köln: Dissertation. Eul Verlag: Bergisch Gladbach.

[Seffah & Metzker 2004] Seffah, A. & Metzker, E.(2004): *The obstacles and myths of usability and software engineering*. In: Commun. ACM (47), ACM Press: New York. S. 71-76.

[Sharp et al. 2006] Sharp, H., Biddle, R., Gray, P., Miller, L., Patton, J. (2006): *Agile development: opportunity or fad?*. In: CHI '06: CHI '06 extended abstracts on Human factors in computing systems. ACM Press: New York. S. 32-35.

[Vossen 2000] Vossen, G. (2000): *Datenmodelle, Datenbanksprachen und Datenbankmanagement-systeme*. 4. Auflage. Oldenbourg Wissenschaftsverlag GmbH: München.

[Weichert 2003] Weichert, S. (2003): Der Knowledge Engineering Prozess bei der Entwicklung eines wissensbasierten Konfigurationssystems für die Blaupunkt GmbH. Magisterarbeit, Universität Hildesheim, Fachbereich III - Informations- und Kommunikationswissenschaften.

[Williams & Kessler 2002] Williams, L. & Kessler, R. (2002): Pair Programming Illuminated. Addison Wesley: Amsterdam.

Integration von Qualitätsdaten für Produktionsanlagen

Markus Nick[1], Sören Schneickert[1], Jürgen Grotepaß[3], Helmut Hamfeld[6], Thomas Rose[2], Torsten Sander[5], Michael Stöhr[4], Werner Stumpe[5], Horst Winterberg[7]
[1]Fraunhofer IESE, Kaiserslautern; [2]FH Münster/Steinfurt;
[3]Freudenberg GmbH, Weinheim; [4]Human Solutions, Kaiserslautern;
[5]PSI Penta GmbH, Berlin; [6]SAC GmbH; [7]Steinbichler GmbH, Neubeuern

Abstract

In automatisierten Produktionsanlagen werden mehr und mehr Sensorsysteme eingesetzt, um die produzierte Qualität zu überwachen und auf Basis gesammelter Prozessdaten sicherzustellen. Die Heterogenität der an unterschiedlichen Stellen im Prozess integrierten Sensoren erfordert einen Ansatz zur einfachen Integration. Ziel der Integration ist die für verschiedene Rollen aufbereitete Qualitätssicht, die auch ein Feedback zur Fehlerdeduktion beinhaltet. In diesem Erfahrungsbericht wird der im Projekt BridgeIT[1] entwickelte Ansatz zur syntaktischen und semantischen Integration von Qualitätsdaten vorgestellt. Der Ansatz ermöglicht insbesondere eine einfache Anbindung neuer Sensorsysteme.

1 Einleitung

Automatische, prozessintegrierte Systeme zur Qualitätskontrolle gewinnen in nahezu allen Industriezweigen vor dem Hintergrund der Null-Fehler-Forderung zunehmend an Bedeutung. Während in diesem Kontext bis vor kurzem nationale bzw. Forschungsschwerpunkte im 4. und 5. EU Rahmenprogramm (IST, GROWTH) die Entwicklung optischer Technologien und Sensorprinzipien bestimmt haben, die zum Einsatz von lokalen (punktuellen) Bildverarbeitungs- bzw. Oberflächeninspektionssystemen zur Qualitätskontrolle (Stahl-, Papier, und Textilindustrie) geführt haben, definieren die damit jetzt nun eröffneten erweiterten Rahmenbedingungen, Impulse zur Softwareentwicklung für die Integration dieser heterogenen Systeme. Die Integration und intelligente Auswertung der Datenströme heterogener Sensorsysteme eröffnet ganz neue Möglichkeiten Rückschlüsse auf defektverursachende Bedingungen zu ziehen. Insbesondere werden auch prozessschrittübergreifende Schlussfolgerungen so erst möglich.

Systeme, die exakte Problemursachen analysieren und in Korrelation zu bereits in der Vergangenheit aufgetretenen Fehlermustern setzen können, gibt es aufgrund der Heterogenität unterschiedlicher Messsysteme und auch oft fehlender Prozesstransparenz derzeit noch keine [1].

In BridgeIT wird diesem Bedarf an objektiver Qualitätserfassung und Deduktion der defektverursachenden Bedingung im Prozess Rechnung getragen. BridgeIT entwickelt einen Portalansatz mit dem Ziel, die in unterschiedlichen Prozesspunkten über unterschiedliche Sensorsysteme erhobenen Qualitätsdaten zusammenzuführen, zu visualisieren und über die Deduktion inverser „Fehlerfortpflanzungsregeln“ geeignete Steuerungsgrößen zur Prozessoptimierung zu generieren. Eine derartige Integration von Inspektionssystemen der unterschiedlichen Fertigungsstufen führt zur Integration von derzeit noch meist konkurrierenden, lokalen oder stufenbezogenen Qualitätsrichtlinien, mit der Folge, dass Qualität an den Prozessgrenzen transparent wird und schlupfbedingte Mehrkosten reduziert werden.

Die Entwicklungen in BridgeIT spiegeln u.a. den Bedarf der deutschen Industrie gemäß der Fraunhofer-Studie „Wissen und Information 2005“ wieder [2]. In dieser Studie wurde die syntaktische und semantische Integration von Daten als eine der wesentlichen Herausforderungen identifiziert, die in den kommenden Jahren von der Industrie anzugehen ist. Weiterhin sind objektive Qualitätsdaten als Basis für die kontinuierliche Verbesserung notwendig, z.B. für Projekte im Rahmen von Six Sigma-Programmen [3]. Die Nutzung integrierter Qualitätsdaten, d.h. Sensordaten für Teilequalität und Anlagenzustand, angereichert mit Informationen zum Anlagenstatus und zum Kontext (Artikel, Auftrag, …), wird zukünftig in Kombination mit Erfahrungswissen die Grundlage zur Realisierung einer möglichst automatisch geregelten Anlage bilden können.

Die in BridgeIT entwickelte *Bridge* zur Integration von Qualitätsdaten von Sensorsystemen und anderen Quellen stellt eine Lösung dar, die diesen Herausforderungen gerecht wird. Die Bridge integriert die Daten syntaktisch und semantisch. Durch ein Modul, das die Bridge-Datenbank beim Ankoppeln neuer Sensoren automatisch auf Basis der Sensordatenbeschreibung in XML erweitert, wird eine einfache Anbindung neuer Sensoren gewährleistet. Die semantische Integration dient der Zuordnung von

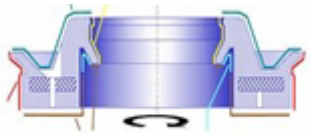


Abbildung 1: Dichtungsring (technische Zeichnung)

Sensordaten zu produzierten Teilen, um Zusammenhänge zwischen Anlagenzustand und Qualität der produzierten Teile erkennen zu können. Die Lösung wurde zunächst als Demonstrator aufgebaut und getestet und dann im Rah-

[1] Ein Teil der Arbeiten wurde durch das BMBF im Rahmen des Projektes BridgeIT im Programm Software Engineering 2006 gefördert. Förderkennzeichen 01ISC22 http://www.BridgeIT.de/

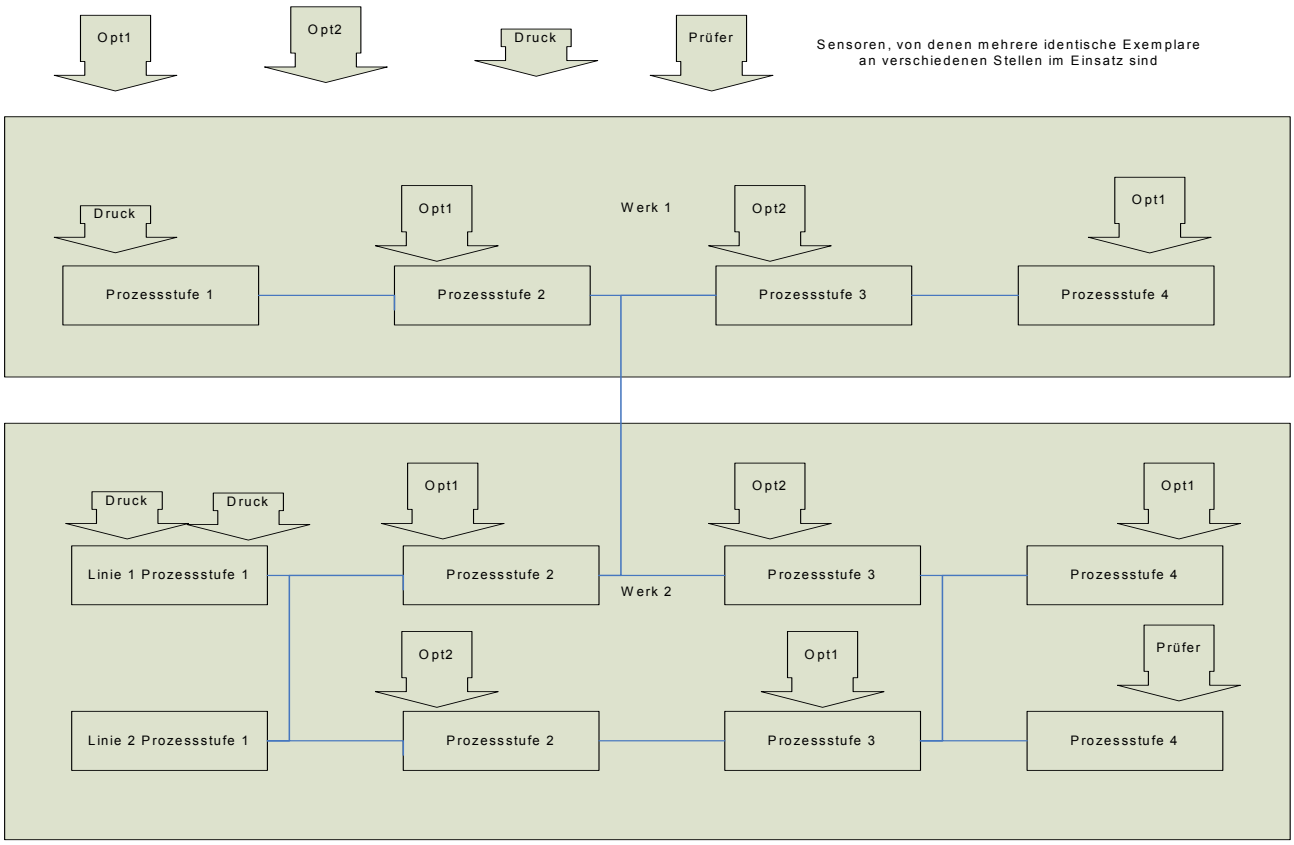


Abbildung 2: Fertigungsablauf mit mehreren Standorten und Linien, deren Zwischenprodukte teilweise von einer Linie zur anderen übergehen können

men eines Piloten an einer Produktionsanlage für Dichtungsringe (Abbildung 1) bei Freudenberg erprobt.

Im Folgenden wird zunächst in Abschnitt 2 eine für den Einsatz der Bridge typische Struktur von Fertigungsabläufen und Anlagen vorgestellt. In Abschnitt 3 wird die Bridge-Architektur als Integrationsplattform vorgestellt. Abschnitt 4 beschreibt den Ansatz für die semantische Integration der Daten. Abschnitt 5 gibt eine Übersicht über die Evaluation. Abschnitt 6 fasst zusammen und gibt einen Überblick über künftige Arbeiten.

2 Struktur von Fertigungsabläufen und Anlagen

Die Bridge soll die Integration von Daten über Produktionsstufen, -linien und -werke hinweg möglich machen. Zum leichteren Verständnis solcher Strukturen ist in Abbildung 2 ein beispielhafter Fertigungsablauf skizziert, der auch im Rest des Artikels zur Illustration des Ansatzes verwendet wird. Der Beispiel-Fertigungsablauf besteht aus 4 Prozessstufen. Es gibt 2 Werke an unterschiedlichen, evtl. weit entfernten Standorten. In jedem Standort gibt es unterschiedlich viele Fertigungslinien, einmal 1 und einmal 2 Linien. Zwischenprodukte werden zwischen den Linien und Werken ausgetauscht.

Es sind viele Sensoren im Einsatz, zum Teil unterschiedliche Exemplare des gleichen Typs, hier charakterisiert als Optischer Sensor 1, Optischer Sensor 2, Drucksensor und menschlicher Prüfer.

Die im Rahmen von BridgeIT betrachteten Sensoren bzw. Datenquellen liefern qualitätsbezogene Daten der folgenden Art:

Als Daten von Sensorsystemen werden Integer-, Float- und Double-Werte, Kurz- und Langtexte, Datumsangaben, sowie Binärdaten verarbeitet. Das sind neben den Kenndaten, direkte Messdaten des jeweiligen Sensors oder aufbereitete Messergebnisse des Sensorsystems.

Die Daten menschlicher Prüfer gehen einerseits als Parameter für den Prozess und die Fehlerklassifikation und -deduktion ein, andererseits als Klassifikationsergebnisse von z.B. Sichtkontrollen.

Die Daten des allgemeinen Anlagenbetriebes und die auftragsbezogenen Daten liefern den Kontext für die Deduktion prozessualer Zusammenhänge und für eine angepasste Visualisierung der Qualitätsdaten.

3 Bridge-Architektur

Ziel der Bridge-Architektur ist es Qualitätsdaten semantisch und syntaktisch zu integrieren und so eine produktionsstufenübergreifende Analyse und Deduktion zu ermöglichen. Durch eine einfache Anbindung vorhandener und neu hinzukommender Datenlieferanten und –verarbeiter soll die Akzeptanz des Systems gesteigert und Integrationszeiten niedrig gehalten werden. Durch die Nutzung von offenen Standards wird eine hohe Transparenz der Lösung erreicht, durch die Möglichkeit des Einsatzes von Open-Source-Produkten sind die Kosten der angebotenen Lösung auf technischer Seite niedrig zu halten.

Um die genannten Ziele zu erreichen, wurde die Definition einer allgemeinen Referenzarchitektur erarbeitet, die zum Zweck des einfachen Zugriffs und einer leichten Verarbeitung auf eine möglichst generische Strukturierung und Speicherung heterogener Qualitätsdaten unter-

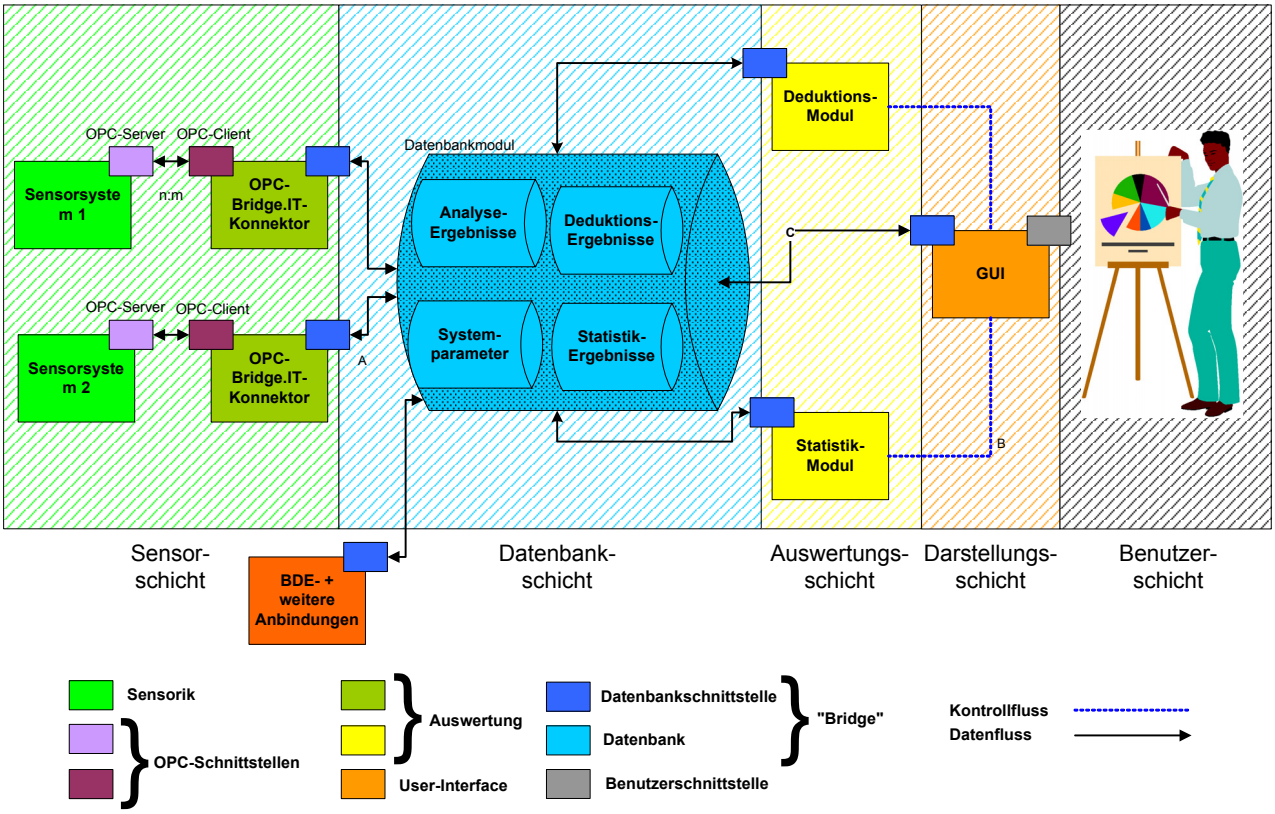


Abbildung 3: Schichten der Bridge-Architektur und schematische Darstellung des Datenflusses

schiedlicher Herkunft in einer Datenbank aufbaut. Die Referenzarchitektur beinhaltet auch die Entwicklung einer Kommunikationsarchitektur, d.h. die Kommunikation zwischen der eingebundenen Datenbank, den verschiedenen Sensorsystemen, dem Betriebsdatenerfassungssystem, den Deduktionsmechanismen und den Statistik- und Visualisierungsmodulen.

Abbildung 3 gibt einen Überblick über die Architektur der Bridge. Hier sind die verschieden Schichten der Architektur und der Datenfluss schematisch dargestellt.

Aus der Sensorschicht werden von heterogenen Sensorsystemen Qualitätsdaten in die Datenbankschicht weitergegeben. Für über OPC[2] angebundene Sensoren existiert dazu ein in BridgeIT entstandener OPC-BridgeIT-Konnektor. Die zu einer Auswertung benötigten Betriebsdaten (in Abbildung 3 mit BDE bezeichnet) und sonstige Systemparameter können ebenfalls über die Schnittstelle in die Datenbank integriert werden. Die Auswertungsschicht nutzt die vorhandenen Daten um Analysen und Statistiken zu erstellen und schreibt ggf. Ergebnisse in die Datenbank zurück. Eine GUI steuert die Module der Auswertungsschicht und holt sich die zur Steuerung und Visualisierung notwendigen Daten aus der Datenbank.

Kernstück der Architektur ist die generische Schnittstelle zur Datenbank, die sowohl lesende als auch schreibende Zugriffe auf unterschiedlichste Datenstrukturen aus einer heterogenen Umgebung gestattet. Die Zugriffsvereinbarung (Datenstruktur) erfolgt über eine Referenz in XML-Notation. Auf die so referenzierte Datenstruktur erfolgt automatisch im laufenden Betrieb eine eindeutige Abbildung in Datenbanktabellen. Datenbankkenntnisse des Nutzers sind so nicht mehr notwendig, stattdessen wird ein offener, bereits validierter Standard in der Beschreibung von Datenstrukturen verwendet. Das Schreiben und Lesen von Daten der vereinbarten Struktur ist über die Referenz auf ein zu füllendes bzw. zu lesendes Datenobjekt realisiert. Die Filterung von Lesedaten wird über Spezifikationen in SQL-Notation erreicht.

Die Bridge-Architektur lässt also eine semantische und syntaktische Integration der Daten zu. Die Integration erfolgt in eine transparente Datenbankstruktur bei automatisierter Anpassung der Datenbankstrukturen im Betrieb. Durch die Schaffung einer generischen Schnittstelle unter Nutzung von Standards wie XML und SQL ist eine einfache Anbindungsmöglichkeit für heterogene Systeme gegeben.

4 Semantische Datenintegration

Der einfache Anschluss beliebiger Sensoren mit heterogenen Datenformaten erfordert die semantische Integration der Daten. Das Kernproblem stellt hierbei die Zuordnung von Sensordaten zu den produzierten Teilen dar, d.h. Sensordaten verschiedener Stufen, Linien und Werke sollen einem produzierten Teil zugeordnet werden können. Der Umgang mit unterschiedlichen Datenformaten erfordert weiterhin eine entsprechende Flexibilität seitens der Module in der Auswerteschicht. Dies liegt jedoch außerhalb des Fokus dieses Artikels.

Im Folgenden wird zunächst das Daten-Integrations-Verfahren zur Lösung des Kernproblems vorgestellt und durch ein Beispielszenario illustriert. Dann wird zu Machbarkeit der Zuweisung eindeutiger IDs zu den Sensoren bzw. Datenquellen verschiedener Art diskutiert und

[2] OPC steht für Openness, Productivity, Collaboration und symbolisiert die Verbindung von Automatisierungskomponenten mittels eines standardisierten, herstellerunabhängigen Zugriffsverfahrens. Siehe auch [7], [8], [9]

entsprechende Erfahrungen aus dem BridgeIT-Projekt berichtet. Weiterhin werden verschiedene Möglichkeiten der Identifikation der produzierten Teile diskutiert.

4.1 Datenintegrationsverfahren

Um sinnvolle Darstellungen von Abläufen und Trendanalysen zu Produktionsprozessen zu ermöglichen, ist es notwendig die Betriebsdaten zu einem Produkt und die von Sensoren und Werkern gelieferten Daten zu dessen Produktionsprozess zusammenführen zu können. Über eine eindeutige Kennzeichnung aller am Prozess beteiligter „Datengeber“ und einer exakten Kenntnis der Konfiguration aller prozessbeteiligten Komponenten zu jedem Zeitpunkt ist diese Datenintegration möglich.

Zur Sicherung der Datenintegration müssen also folgende Daten festgelegt und erreichbar sein:

Die *Konfigurationen aller beteiligten Produktionseinheiten*. Die Konfiguration enthält notwendige Angaben zum Aufbau eines Konzerns, seiner Werke, deren Linien und deren Stufen (inkl. der integrierten Sensoren) im betrachteten Zeitraum.

Die *Artikelkennungen* und Angaben zu den Artikeln eines Konzerns, die für die Qualitätsdatenauswertung notwendig sind.

Die eindeutigen *Kennungen der Aufträge* (oder Chargen) und Angaben zu den Aufträgen, die für die Qualitätsdatenauswertung notwendig sind. Sie sind u.a. das Bindeglied zu den Artikeldaten und zum jeweiligen Kunden mit seinen speziellen Qualitätskriterien und der beauftragten Menge.

Die eindeutigen *Kennungen der Teilaufträge* (Unterchargen) aller Aufträge und Angaben zu den Teilaufträgen, die für die Qualitätsdatenauswertung notwendig sind. Sie sind das Bindeglied zu den Auftragsdaten und enthalten ebenfalls spezielle Qualitätskriterien und beschreiben implizit den Produktionsablauf.

Bei der Datenerhebung seitens der Sensorik müssen folgende Daten übermittelt werden:

Pro (Teil)Auftrag ist die eindeutige Kennung des (Teil)Auftrages, dessen realer Startzeitpunkt und Endzeitpunkt zu übertragen. Mit der Übertragung dieser Daten ist der zeitliche Rahmen für Messdaten festgelegt und die Zuordnung zum produzierten Artikel geschaffen.

Pro Messung ist die Kennung des datenerzeugenden Sensors, erhobene Messdaten, notwendige Zusatzdaten und der Zeitstempel der Messdatenaufnahme zu übertragen. Über die Kennung und den Zeitstempel ist mit Hilfe der Konfigurationsdaten eine genaue räumliche, auftrags- und ablaufbezogene Zuordnung der gelieferten Daten zu treffen.

Bei der Verknüpfung der Sensordaten mit den Produktdaten sind nunmehr folgende Fälle zu unterscheiden:

1) Das Sensorsystem überträgt die (Teil-)Auftragskennung

Beim Einsatz eines Sensors sind bei Produktwechsel in aller Regel auch andere Parametrisierungen nötig. Die Parametrisierung hängt unmittelbar mit dem (Teil-)Auftrag zusammen, da auch bei gleichem Produkt unterschiedliche Anforderungen (etwa bzgl. der Toleranzen) seitens des Kunden vorliegen können. Überträgt das Sensorsystem mit jeder Messung die (Teil-)Auftragskennung lassen sich Produkt- und Messdaten direkt zusammenführen.

2) Das Sensorsystem überträgt keine (Teil-)Auftragskennung

Häufig ist es aus praktischen Erwägungen heraus nicht sinnvoll lösbar die (Teil-)Auftragskennung mit den Sensordaten zu übertragen. Es gibt dennoch zwei Möglichkeiten Mess- und Produktdaten zusammenzuführen:

a) Die Daten aller Sensorsysteme einer Linie werden zentral verwaltet und „angereichert“ mit den (Teil-)Auftragskennungen an die Bridge übertragen. (z.B. Sensorsysteme als OPC-Server, Einstellung der Daten in die Bridge zentral über einen OPC-Client, Anreicherung der im Client administrierbaren Daten mit den Produktionsdaten).

b) Können die Sensordaten nicht unmittelbar mit den zugehörigen (Teil-)Auftragskennungen versehen werden, kann der Rückschluss auf das Produkt auch über den Vergleich der Zeitstempel des (Unter)Auftrags und der Messdaten gezogen werden.

Durch die Übertragung der o.a. Daten lassen sich also alle qualitätsrelevanten Daten zusammenführen. Da in Fall 2b die Verknüpfung der Daten nur implizit vorliegt, sind die Varianten 1 und 2a möglichst vorzuziehen.

Welche Daten zusätzlich übertragen werden müssen, hängt davon ab, welche Qualitätssichten entstehen, oder welche Analysen gemacht werden sollen. Dies können z.B. Auswertungen bzgl. Artikeln, Zeitphasen, Produktionslinien, etc. sein.

4.2 Beispielszenario

Das Szenario illustriert am Beispiel der Struktur von Abbildung 2 das Verfahren zur semantischen Datenintegration bzgl. folgender Punkte: (1) korrekte Zuordnung von produzierten Teilen und zugehörigen Prüfdaten für ein Teil, das an Tag x in Werk 2 auf Linie 2 Stufe 1, dann auf Werk 2 Linie 1 Stufe 2, dann an Tag x+1 auf Werk 1 Linie 1 Stufe 3, und dann an Tag x+2 auf Werk 1 Linie 1 Stufe 4 gefertigt wurde; (2) eindeutige Bezeichnung und Erkennung der Sensoren.

Die Abfolge des Produktionsprozesses ist in den Daten des Auftrages und seiner Unteraufträge festgehalten. Im Beispiel besteht der Auftrag A1 zu Produkt P1 aus den 3 liniengebundenen Unteraufträgen A1_U1 (Werk2, Linie2, Stufe1), A1_U2 (Werk2, Linie1, Stufe2) und A1_U3 (Werk1, Linie1, Stufe3+4).

Für alle Sensoren sei konzernweit eine eindeutige Kennung festgelegt. Für die Sensorsysteme mit den Kennungen W2_L2_S2_Opt1_1, W1_L1_S3_Opt1_1 und W1_L1_S4_Opt2_1 ist der Einbauort zum Zeitpunkt der Messdatenerfassung in den Konfigurationsdaten erfasst.

Die Sensorsysteme übertragen bei jeder Messung ihre Kennung, die Messdaten, die Messungsnummer und einen Zeitstempel. Die Prüfdaten der im Beispiel beteiligten Sensorsysteme der Typen Opt1 und Opt2 seien der Einfachheit halber auf die Unterscheidung „in Ordnung“ (IO) und „nicht in Ordnung“ (NIO) beschränkt.

Beispiel: In einem Testlauf von n Prüfteilen soll herausgefunden werden, wie viele Teile, die in vorangehenden Stufen des Prozesses von der Sensorik als NIO bewertet werden auch in den nachfolgenden Stufen mit NIO bewertet werden.

Voraussetzung:

Die Reihenfolge des Teileflusses zwischen allen Stufen des Prozesses muss sichergestellt sein

Vorgehen:

Die Auswertung wird gestartet, wenn alle Teile alle Stufen durchlaufen haben. Jetzt liegt von jedem Sensor ein Messdatensatz nach dem Muster (sensorID, unter-

auftragsID, messungNr, IO_NIO, zeitstempel) zu jedem Prüfteil in der Datenbank und zu jedem Unterauftrag ein Eintrag nach dem Muster (unterauftragsID, startzeitpunkt, endzeitpunkt, werkID, linieID, startstufeID, endstufeID) vor. Der Analysator sucht unter der Auftragsnummer A1 die zugehörigen Unteraufträge A1_U1, A1_U2 und A1_U3 und arbeitet anhand deren Reihenfolge die Sensoreinträge der Unteraufträge ab. Über die Konfigurationsdaten kann er die Reihenfolge der Sensoren innerhalb einer (Teil)Linie feststellen. Anhand der Messungsnummern lassen sich nun die Messdatenergebnisse einzelner Teile zusammenführen, die gewünschte Analyse durchführen und die Visualisierung steuern.
Alternativ kann mit der Visualisierung schon begonnen werden, wenn das erste Teil alle Stufen durchlaufen hat.

4.3 Zuweisung eindeutiger IDs

Im Folgenden werden Verfahren zur Zuweisung eindeutiger IDs zu Sensoren bzw. Datenquellen und die Machbarkeit dieser Verfahren diskutiert und mit Erfahrungen aus dem BridgeIT-Projekt untermauert.

Zuweisung von IDs zu Sensoren: Wenn der Sensorhersteller oder Sensorsystemhersteller weltweit eindeutige Kennungen seiner Sensoren vergibt, kann diese Kennung für die Identifikation des Sensors übernommen werden. Falls nicht, muss der Anlagenbetreiber dem Sensor eine konzernweit eindeutige Identifikation des Sensors zuweisen. Sensoren mit einer OPC-Schnittstelle erhalten über ihre DCOM-Basis keine ID im o.g. Sinne, da es sich bei der AppID in der Praxis letztlich um eine auf den Sensortyp bezogene Kennzeichnung handelt. Sensoren des gleichen Typs erhalten dieselbe AppID und sind ohne Zusatzinformation nicht unterscheidbar.

Eigentlich sollte ein Anlagenbauer bzw. -betreiber von den Sensoranbietern eine weltweit eindeutige Kennzeichnung der Sensoreinheiten einfordern (wie etwa die MAC-Adresse im Netzwerkbereich). Die Forderung erscheint auch im Hinblick auf die Vision einer digitalen Fabrik sinnvoll. Die Kennzeichnung wird dann bei der Datenspeicherung durch den Sensor mit übertragen.

Da dies aus heutiger Sicht nicht möglich ist, muss vom Anlagenbauer bzw. -betreiber gefordert werden, dass er für seine Produktionsbetriebe eine konzernweite Kennung aller Daten erzeugenden Komponenten sicherstellt, die dann bei der Einbindung eines Sensors in das Bridge-System mit übertragen wird. Dies kann verteilt geschehen, indem in den Betriebsdaten die Konfigurationen aller Anlagen zu jedem Zeitpunkt festgehalten sind und die Sensoren ihre lokalen Kennungen in das Bridge-System übertragen. Die Konfigurationen beschreiben dabei die exakte Anordnung der Komponenten einer Produktionseinheit zu jedem beliebigen Zeitpunkt.

Zuweisung von IDs zu menschlichen Prüfern: Bei menschlichen Prüfern stellt eine konzernweite Ortsbestimmung und die Personalnummer des Prüfers ein eindeutiges Kürzel sicher. Die Ortsbestimmung, wo der Prüfer in den Prozess eingreift, ist in den o.g. Konfigurationsdaten enthalten.

Zuweisung von IDs zu Artikeln (Produkten): Es muss eine konzernweit eindeutige Artikelkennung vorliegen, die es gestattet die verschiedenen (evtl. weltweit verteilten) Produktionsergebnisse eines Artikels zusammenzuführen. In der Regel liegen eindeutige Artikelkennungen vor. Über die Artikelkennung stehen Daten zur Artikelbeschreibung zur Verfügung.

Zuweisung von IDs zu Aufträgen und Unteraufträgen: Es muss eine konzernweit eindeutige Auftragskennung (evtl. aufgeteilt in mehrere Unteraufträge) vorliegen, die es erlaubt die verschiedenen (evtl. weltweit verteilten) Produktionsstufenergebnisse eines Auftrages zusammenzuführen. Ein Auftrag ergibt sich dabei aus der Bestellung eines Artikels in bestimmter Stückzahl für einen Kunden mit bestimmten Qualitätsforderungen. In der Regel liegen eindeutige Auftragskennungen vor. Über die Auftragskennung stehen Daten zu Qualitätsanforderungen, Prozessparametern und Prozessverläufen sowie zu Prozessverteilungen zur Verfügung.

4.4 Identifikation produzierter Teile

Es gibt zwei Möglichkeiten, um ein Teil eindeutig identifizieren zu können:

1. Single-Piece-Flow
2. Markierung des produzierten Teils

Der sog. Single-Piece-Flow garantiert den ordnungserhaltenden Durchlauf der produzierten Teile durch die Stufen und Linien. Hiermit ist es möglich, die Daten anhand von Reihenfolge und Zeitstempel den produzierten Teilen eindeutig zuzuordnen, wie oben dargestellt. Außerdem können Auswirkungen von Werkzeugabnutzungen direkt im Fertigungsabschnitt kontrolliert und behoben werden. Oft muss neben einer Single-Piece-Flow-Fertigung in der Produktion auch die Blockfertigung berücksichtigt werden. Bei der Blockfertigung werden Artikelströme aufgelöster Teileordnungen über Puffersysteme, wie z.B. Wendelförderer, an definierten Prozessschnittstellen zugeführt. Sofern diese Puffersysteme lediglich der Vereinzelung von gemeinsam in einem Nest gefertigten Teilen dienen, geht hiermit lediglich die exakte Zuordnung zu einem Platz im Nest (bei Formgebungswerkzeugen ist dies die Kavität) verloren. Sobald in das Puffersystem jedoch Teile aus mehreren Produktionsschichten landen, kann keine korrekte Zuordnung mehr getroffen werden.

Die Markierung des produzierten Teils ermöglicht somit eine sichere Zuordnung zu den Sensordaten, sofern die Markierung innerhalb eines jeden Anlagenteils mit Reihenfolgeerhaltung ausgelesen und der Bridge zugeführt wird.

Die Markierung kann z.B. durch eine aufgedruckte oder strichcodierte Nummer oder bei größeren Teilen durch ein RFID-Tag erfolgen. Wird die Nummer oder das Tag dauerhaft auf dem Teil angebracht, können auch noch im Betrieb des Teils Bezüge zu den Sensordaten aus der Produktion hergestellt werden. Diese Nachverfolgbarkeit kann z.B. bei sicherheitskritischen Teilen relevant werden.

Die eindeutige Identifikation produzierter Teile hängt in der Praxis insbesondere von der Anlagenstruktur ab.

5 Evaluation

Die Integration und Auswertung der Daten soll zur Unterstützung bei prozessoptimierenden Entscheidungen dienen. Hierfür können im Rahmen zukünftiger Arbeiten auch KI-Verfahren eingesetzt werden.
Als Kenngröße in der Evaluierung der Projektlösung wird die Fähigkeit des Deduktionsmoduls bewertet, Prozesszusammenhänge aus der Statistik von Fehlerbildern der Sensoren zu schließen. Dieses Modul liefert den verschiedenen Rollen Informationen, die die aktuell produzierte Qualität erkennen lassen und die Optimierung der Produktion unterstützen.

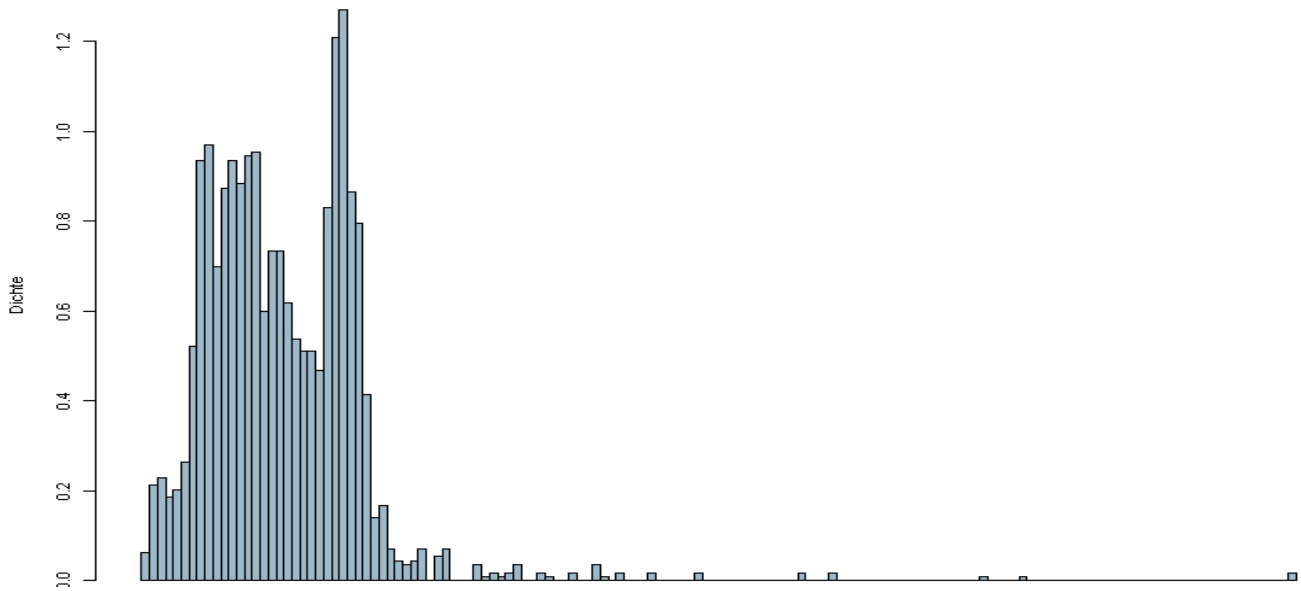


Abbildung 4: Verteilungsdichtefunktion von Fehlergravitäten

Später soll dann ein KI-Verfahren diese Unterstützung automatisieren. Die Evaluationsphase in BridgeIT muss daher zeigen, dass die Integration durch die Bridge für die automatisierte Unterstützung durch ein KI-Verfahren geeignet ist. Dabei wird davon ausgegangen, dass Mensch und KI-Verfahren ihre Entscheidungen auf der gleichen Datenbasis treffen.

Die Lösung wurde zunächst als Demonstrator aufgebaut und getestet (Abschnitt 5.1) und dann im Rahmen eines Piloten an einer Produktionsanlage für Dichtungsringe bei Freudenberg erprobt (Abschnitt 5.2). Weiterhin wird die Performance der Bridge getestet, um zu analysieren, in welchem Umfang Sensoren angebunden werden können (Abschnitt 5.3).

5.1 Demonstrator

Ziel des Demonstrators war der Machbarkeitsnachweis unter Laborbedingungen bzgl. der Anbindung verschiedener Sensoren, der Funktion der Datenflusskette und der Auswertbarkeit der gesammelten Daten. Angebunden wurden ein Sensor zur Oberflächeninspektion eines Dichtungsrings und ein ABIS II Sensor zur Oberflächeninspektion von Freiformteilen in der Automobilindustrie.
Die Datenflusskette konnte gemäß der Architektur realisiert werden. Die Auswertungen (Als Beispiel diene die Verteilungsdichte von Fehlergravitäten in Abbildung 4) mit dem Statistikmodul waren möglich. Damit konnte der Demonstrator die Anwendbarkeit der Lösung unter Laborbedingungen zeigen.

5.2 Pilot

Ziel des Piloten war der Machbarkeitsnachweis unter realen Bedingungen bzgl. der Datenmenge für eine Anlage. Für diesen Zweck wurde die Bridge in einem Pilotprozess an einer Produktionsanlage für Dichtungsprodukte bei Freudenberg evaluiert. In ausgewählten Produktionsläufen des 8-wöchigen Pilotbetriebs wurden hierbei Bildverarbeitungs- und Betriebsdaten aus produzierten Teilen gesammelt. Die Integration der Daten war erfolgreich. Der Pilot zeigte somit die Anwendbarkeit der Lösung unter realen Produktionsbedingungen für eine Anlage mit einem Sensorsystem zur Oberflächeninspektion einer Familie unterschiedlicher Dichtungsprodukte.

5.3 Performance der Bridge

Um abschätzen zu können, wie viele Sensorsysteme und Anlagen bedient werden können, wird die Performance der Bridge noch evaluiert.

Die Performance wird im Wesentlichen durch die Performance der Datenbank für Ablage und Abruf von Sensordaten bestimmt. Exemplarische Messungen werden in Kürze vorgenommen.[3]

6 Verwandte Arbeiten

Die Bridge mit ihrer Bridge-Datenbank zeigt eine gewisse Verwandtschaft zu Datawarehouses [4]. Datawarehouses dienen ebenfalls der Integration von Daten aus verteilten und unterschiedlich strukturierten Datenbeständen, um eine globale Sicht auf die Quelldaten und damit übergreifende Auswertungen zu möglich zu machen. Dazu werden i.d.R. ebenso Datenbanken verwendet. Allerdings sollen mittels Datawarehouses auch Daten die für das operative Geschäft genutzt werden, von solchen Daten separiert werden, die im Datawarehouse z.B. für Aufgaben des Berichtswesens, der Entscheidungsunterstützung, der Geschäftsanalyse sowie des Controllings und der Unternehmensführung verwendet werden. Dies bedeutet, dass die Daten im Datawarehouse nicht realzeit-aktuell sind, d.h. z.B. nur einmal am Tag aktualisiert werden. Für die Zwecke der Produktionssteuerung bei BridgeIT ist dies jedoch nicht akzeptabel, da auf Basis der Qualitätsdaten eine Steuerung in Realzeit erfolgt. Dies gilt insbesondere auch für die im Rahmen des Projektes CheckMATE geplante weitergehende Automatisierung dieser Steuerungsaufgaben.

In den letzten Jahren hat sich mehr und mehr die Abkehr von turnusmäßiger Beladung hin zum sog. Real-Time-Data-Warehousing vollzogen [5][6].
In diesem Sinne könnte man die Bridge und ihre Datenbank als ein solches Real-Time-Data-Warehouse für Qua-

[3] Ergebnisse sollten Anfang bis Mitte September vorliegen.

litätsdaten von Produktionsanlagen betrachten. Während aber Datawarehouses i.d.R. Daten von Datenbanken integrieren, die gewöhnlich eine Standard-Schnittstelle mit SQL anbieten, stand hier die Einfachheit der Anbindung der Sensoren und anderer Datenquellen an die Datenbank im Vordergrund.

7 Zusammenfassung und künftige Arbeiten

In diesem Erfahrungsbericht wurde der Bridge-Ansatz zur syntaktischen und semantischen Integration von Qualitätsdaten vorgestellt. Der Ansatz ermöglicht insbesondere eine einfache Anbindung neuer Sensorsysteme durch eine generische Schnittstelle zur angebundenen Datenbank, die Zugriffe auf Datenstrukturen aus einer heterogenen Umgebung auf Basis einer Datenbeschreibung in XML erlaubt Die semantische Integration der Daten wird durch eine systematische Vergabe und Verwendung von IDs bis hin zur Werkzeugebene und für produzierte Teile in Kombination mit Zeitstempeln für die Daten machbar. Für nicht-reihenfolgeerhaltende Fertigungsabläufe sind ggf. besondere Vorkehrungen zu treffen, um die Sensordaten den produzierten Teilen zuordnen zu können. Die Anwendbarkeit der entwickelten Lösung wurde mit einem Demonstrator unter Laborbedingungen sowie in einem Piloten an einer Produktionsanlage im regulären Betrieb gezeigt.

Im Projekt CheckMATE[4] soll nunmehr die Nutzung der integrierten Qualitätsdaten in Kombination mit Erfahrungswissen zur Realisierung des für den „kleinsten Qualitätssicherungskreises der Welt“ genutzt werden. Dies bedeutet, dass unter Verwendung der Sensordaten für Teilequalität und Anlagezustand sowie Anlagenstatusinformation und Kontextinformationen (Artikel, Auftrag) die Anlage möglichst automatisch geregelt wird. Rückfragen an Werker oder Produktionssteuerer erfolgen nur dann, wenn die Erfahrungen zur Regelung im aktuellen Kontext noch nicht hinreichend valide sind. Das Erfahrungswissen wurde bereits im beschriebenen Pilot akquiriert und wird derzeit formalisiert.

8 Literatur

[1] Jürgen Grotepaß. *Vision Inspection Systems as Integral Elements for Continuous Improvements of Production Lines*. EMVA, 3rd European Machine Vision Business Conference; Palermo; April 29-30, 2005 http://www.emva.org/

[2] Fraunhofer Gesellschaft. *Studie Wissen und Information 2005*. Fraunhofer IRB Verlag, 2006.

[3] Mc Kinsey. *Analyse des in der Automobilzulieferindustrie erreichten Qualitätsstandards*. SIS „Surface Inspection Summit, Aachen, 2003.

[4] William H. Inmon, Richard D. Hackathorn. *Using the Data Warehouse*. John Wiley & Sons, ISBN 0-471-05966-8

[5] Colin White. *Intelligent Business Strategies: Real-Time Data Warehousing Heats Up.* DM Review Magazine; August 2002.

[6] Wikipedia. *Data-Warehouse*. http://www.wikipedia.de/, Stand 10. Juli 2006.

[7] Frank Iwanitz, Jürgen Lange. *OPC. Grundlagen, Implementierung und Anwendung*. 3. Auflage, Hüthig

[8] OPC Foundation. http://www.opceurope.org/

[9] Thuan L. Thai. *Learning DCOM. Distributed Components on Windows*. O'Reilly Media; 2000

[4] http://www.checkmate-online.de/

Knowledge Search within a Company-WIKI

Stephanie Müller[1], Nils Kritzler[1], Alexander Tartakovski[1], Ralph Bergmann[1], and Ralph Traphöner[2]

[1]University of Trier,
Department of Business Information Systems II,
54286 Trier, Germany
{muel4102|krit4101}@uni-trier.de
{Alexander.Tartakovski|bergmann}@wi2.uni-trier.de

[2]Ralph Traphöner
empolis GmbH
Europaallee 10, 67657 Kaiserslautern, Germany
ralph.traphoener@empolis.com

Abstract

The usage of Wikis for the purpose of knowledge management within a business company is only of value if the stored information can be found easily. The fundamental characteristic of a Wiki, its easy and informal usage, results in large amounts of steadily changing, unstructured documents. The widely used full-text search often provides search results of insufficient accuracy. In this paper, we will present an approach likely to improve search quality, through the use of Semantic Web, Text Mining, and Case Based Reasoning (CBR) technologies. Search results are more precise and complete because, in contrast to full-text search, the proposed knowledge-based search operates on the semantic layer.

1 Introduction

The concept of Wiki [Baeza-Yates, 1999] provides a simple and efficient way of creating knowledge and making it accessible. It is suited especially for the purpose of knowledge management within a company because of the great acceptance by employees.

However, the authoring simplicity results over time in a large amount of steadily changing, unstructured documents. Consequently, the users often lose the overview of available content. Full-text search, which is usually implemented within Wiki-systems, does not help sufficiently to overcome that problem [Cesarano *et al.*, 2003]. It does not take the relationships between concepts and objects, synonyms, and multilingualism among other things into account and therefore often provides insufficient search-results [Cesarano *et al.*, 2003; Money and Turner, 2004]. In this situation user acceptance decreases, since the desired information may often only be found after several attempts using full-text search.

The same situation could be observed at empolis GmbH[1] after the introduction of a Wiki for the purpose of knowledge management. Since its implementation in February 2004, the amount of pages increased up to 5,500 in November 2004. Following this considerable increase the user acceptance began to decrease.
empolis GmbH and the Department of Business Information Systems II, University of Trier launched a project with the objective of overcoming the explained difficulties by developing a knowledge-based search function to enable improved access to the information filed in the Wiki.

In this paper, we present the concept and the realisation of the search function using a combination of following technologies: Semantic Web, Text Mining, and Case Based Reasoning (CBR) [Davenport and Prusak, 1998; Davenport and Grover, 2001; Leuf and Cunningham, 2001]. A domain specific ontology provides a vocabulary for the semantic annotation of the content. The annotation is constructed automatically with the help of text mining technology. Similarity-based search on the semantically observed content is done using CBR-retrieval technology.

The approach to use semantic information to improve the search functionality within a Wiki has also been followed in the Semantic MediaWiki project[2]. There, the authors of the Wiki article enter semantic information themselves. In contrast, following our approach, semantic information is allocated automatically to each article.

Section 2 describes the application of Wiki within a company in terms of knowledge management. While section 3 introduces the concept of knowledge-based search, section 4 demonstrates its realisation. The last section concludes the paper with summary and discussion.

2 Wiki for emphasising knowledge-sharing

"Increasingly, knowledge is recognized as an organization's most valuable resource and the best

[1] empolis is an arvato AG subsidiary, an international media service company and part of Bertelsmann AG. It is supplier of enterprise content and knowledge management solutions.

[2] http://wiki.ontoworld.org/index.php/Semantic_MediaWiki

foundation of sustained competitive advantage" [Maedche, 2002]. Knowledge management is rapidly becoming an integrated business function as companies realise that effective management of intellectual resources is connected to competitiveness [Abecker and Elst, 2004]. The difficulty lies in gathering knowledge as well as its creation, allocation, storage and location.

One instrument to organise and cross-link knowledge is a Wiki [Baeza-Yates, 1999]. This concept offers a forum for its users to share knowledge and look up information. It simplifies and encourages knowledge sharing, as its usage is simple and quick. The handling of Wiki does not require conformity to many rules and there is also no need to setup specific software.

These characteristics lead to a lack of formal structure as well as a dynamic changing landscape, which makes it very difficult to keep an overview of the content. In particular the constant growth, which is anarchical and uncontrolled, makes this task more and more complicated. Additionally, many inner-company Wikis are kept in several languages, which aggravates the task.

The main aim of a Wiki is the re-usability of knowledge. Its existence is only of interest if not just the storage of knowledge is realised in an easy and uncomplicated way but also the location of the stored information is quick and simple to reference. To reach this objective, the improvement of the search functionality is needed to enable relevant information to be found more easily and is presented in the following.

3 Knowledge-based search supported by the concept of ontology

The approach most used within a Wiki is full-text search; but it is already widely known that results are not satisfactory. Using full-text search, a result is only a 100% hit, if the title of an article corresponds exactly to the query. The problem of this method is the total ignorance of similarities between words like singular and plural or different words used for the same thing; multilingualism is not cared for either. The listing of results contains many irrelevant articles, misses out several relevant documents and the ordering of relevance does not reflect the real order of importance of the results. This insufficient search functionality leads to the decrease of usage of Wiki for knowledge sharing.

To improve the insufficient search functionality, the approach of a knowledge-based search function is presented in this paper. Following this approach, ontology provides the necessary background knowledge. Outgoing from this knowledge, text miner software automatically annotates the unstructured documents with semantic information. Then a case base reasoning suite is used to represent the achieved semantic content of the articles as cases in a case base and to perform the similarity based search.

3.1 Semantic annotation and knowledge-based search

The first step while developing knowledge-based search functionality is the creation of semantic annotations, which represent the content of every Wiki article. According to the approach presented in this paper, every single annotation consists of a set of concepts, which are identified within a Wiki article by the text mining software. Search is then performed by comparison of the query with the annotations of the several Wiki documents. This kind of search provides a faster access to the content of a Wiki. Regarding its usually large number of documents, which is also constantly increasing, this methodology is most appropriate in that context. If annotations are constructed accurately by the text mining software it is also possible to offer better finding of relevant information.

The main problem that has to be solved is the ambiguity of natural language; it manifests itself in the synonym and polysemy phenomenon [Money and Turner, 2004]. The synonym phenomenon refers to the problem that the same concept can be represented in many different ways. The fact that words can have different meanings in different contexts is defined by polysemy [Cesarano *et al.*, 2003]. To provide a fast and reliable knowledge-based search, the knowledge of the language use within a Wiki is essential. In particular, the problem of imprecise interpretation of the search-query and the consequential need of "processes to ‚interpret' the query, to retrieve the expanded query condition according to the interpretation, and to evaluate the closeness of the result to the original query" [Liu, 2001] require the knowledge of the language use within the Wiki. Without this interpretation based on the knowledge, satisfactory results for the user query are not possible.

As the Wiki landscape is changing constantly, manual annotation of the various articles is not appropriate in this context. For automatically annotating documents, its content has to be classified by the text mining software without human intervention. Consequently, the text mining software requires a knowledge base including domain specific language knowledge in order to use it for the annotation purposes.

3.2 Ontologies to provide the necessary background knowledge

The domain specific knowledge is represented by the usage of ontologies. It provides a common understanding of things of the world and for that reason is means to bridge the 'semantic gap' existing between the actual syntactic representation of information and its conceptualisation [Davenport and Grover, 2001].

Ontologies are the key means to annotate unstructured documents with semantic information, to integrate information and to generate specific views that make knowledge access easier [Davenport and Grover, 2001]. They provide the domain knowledge for the realisation of knowledge-based search.

Before mapping Wiki articles according to the domain ontology, the latter one demonstrating the conceptual model of the Wiki has to be created. This scheme represents the set of concepts, instances and relationships which map the content of the Wiki. A thesaurus completes that model; synonyms, pseudo-synonyms as well as acronyms are included to enhance semantic understanding [Cesarano *et al.*, 2003].

After the ontology is created, the metadata of various articles can be produced, which means annotating the documents. The extracted words of an article are mapped to the concepts of the ontology. The annotation resulting from this process consists of ontology-concepts, which

present the content of each article. It is stored together with the corresponding article and is available for the search function from then on. To support the search functionality it is appropriate to annotate each article after creation or editing.

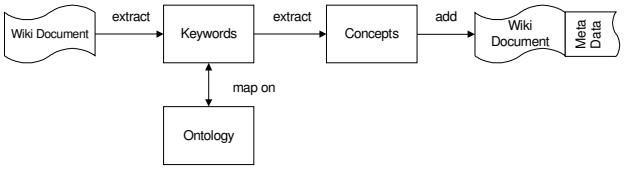


Figure 1 Process of the creation of metadata

A document retrieval process can be carried out in the following way. First, a query has to be annotated with metadata in the same manner as every Wiki article. During the search process, the annotation of the query is compared with the metadata of the articles. Afterwards, the articles having metadata with a high similarity to the metadata of the query are presented to the inquirer, ordered with respect to similarity and diversity.

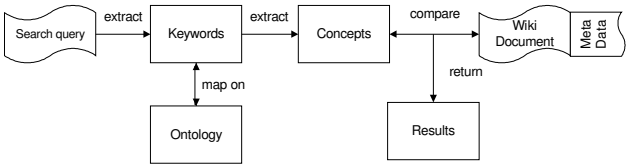


Figure 2 Query processing for retrieval

4 Realisation

The realisation of the above explained theoretical approach can be divided into two parts: The creation of the ontology and the development of the search functionality itself.

4.1 Identification of the specific domain knowledge of the Wiki

As the ontology needs to model the content of the Wiki, the domain knowledge of the Wiki has to be detected.

The starting point for this procedure is the collection of all articles contained in the Wiki; this set is named ‘corpus’ in the following text. To find out which words of the corpus reflect the content of the Wiki, word frequency lists are used. The word frequency list is built with the help of concordance programs. A concordance can be described as an alphabetical index of all the words in a text or corpus of texts, showing every contextual occurrence of a word.

The first step is sorting out useless words that do not describe the content of the articles. These words are called ‘stop words’ and are so common that they are worthless in giving any information about the essence of an article. The result is the concordance of the remaining words. These are ordered according to their frequency. Next, it is necessary to define how often a word has to be present to be important enough for assimilation into the ontology. Words with a lower frequency are deleted accordingly. That choice is dependent on the total number of words on the list.

After this proceeding, the word frequency list still contains several words that do not imply any relevance for the ontology. As language is ambiguous, it is not possible to eliminate every useless word during the concordance process. This has to be done manually. There is also a need to remove words that might be significant but are used in several contexts and thus their meaning is not definite. After the removal of these insignificant words, the content of the resulting list reflects the environment of the Wiki. The list at this stage is the initial point in creating the ontology.

4.2 Ontology creation process

Starting with the modified word frequency list, the manual creation process of the ontology can take place. To keep the overview, a visualisation tool is used.

“There is no ‘correct’ way or methodology for developing ontologies” [Maletic and Marcus, 2001], several approaches exist, depending on the application that one has in mind. One possible way is to start with a rough first pass, which is then refined in an iterative process [Maletic and Marcus, 2001]. As this methodology fits into the given context we decided to use it.

This is done by allocation of a concept to every word of the list. The resulting concepts have to be ordered by the ‘kind-of’ relation, which is well known from the oo-modelling. Thus, while filling the ontology it has to be decided where to put every concept into the hierarchical scheme. Following this procedure, additional facts have to be taken into account: sometimes more than one place exists where the concept belongs or there is the necessity to create additional concepts to merge several concepts. The last step within this process is the consideration of whether there is a need to create additional concepts, which fit into the given context and therefore enrich the ontology.

Apart from the ’kind-of’ relation, which is already contained in the hierarchical scheme, other sorts of relations between several concepts are created as well. Because the CBR-based search functionality is intended to be performed it is sufficient to define solely the strength of connection between concepts, instead of complete and explicit definition of all relations. The strength of a connection can be evaluated to find relevant search results. This is realised by similarity weights, which range from 0 for ‘no relation’ to 1, which means ‘equals’. This provides a model that is applicable for any CBR-suite and enables it to retrieve documents that contain similar keywords to those formulated in the query.

The last step is the creation of a thesaurus for every concept. This feature supplies a base of keywords, which stands for the respective concept and serves for the semantic annotation executed by the text mining functionality. The figure below shows an extract of an example ontology.

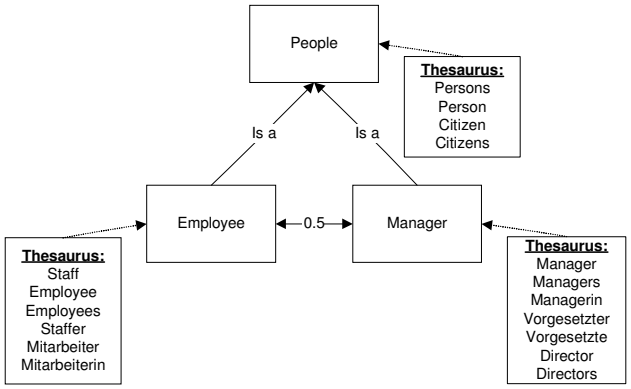


Figure 3 Extract of an example ontology

4.3 Embedding the ontology into CBR-Suite

To execute the search it is necessary to transform the ontology into a data model of a CBR-suite.

Most CBR-suites adopt a classic structural CBR approach, where each case is described by a finite and structured set of attribute-value pairs. The data model defines attributes allowed to be used for case description, and a global similarity function, which is used to compare queries with cases. Attribute names and respective types representing feasible value ranges define attributes. The global similarity function is usually defined according to the local-global principle. First, local similarity functions are defined for every attribute. Such a function compares two values from a certain attribute type; i.e. it compares a query and a case only with respect to a chosen attribute. To achieve the global similarity measure, local similarities are aggregated by an aggregation function, which provides a possibility to compare any query and any case to each other.

Unfortunately, an ontology that encodes an object oriented data model cannot be mapped one-to-one to the attribute-value structural CBR model. In the following the transformation approach is presented, which has been developed during realisation of the knowledge based search functionality.

1. At the beginning, the first proportion of attribute names has to be defined. Candidates for these attribute names are the names of those concepts that are located at the top of the inheritance hierarchy within the ontology. According to the extract of the ontology displayed above, the corresponding case model would include the attribute 'People'. A final version of the case model which was developed for the empolis Wiki includes also attributes: 'Software', 'Hardware', 'Companies', 'Platform', 'Service', and so on. Within the final ontology these are the names of the concepts at the top of the hierarchy.
2. The next task is the definition of attribute types. Since the attribute-value structural CBR approach does not support inheritance explicitly, a slightly unnatural modelling manner has to be chosen. All the names of subconcepts, which are located within the inheritance tree of the ontology under a certain top-level concept, can be understood as symbols, which build the type of the attribute originated from that top-level concept. According to the extract of the ontology, the type of the attribute 'People' is {Employee, Manager}, since the concepts 'Employee' and 'Manager' are both subconcepts of the top-level concept 'People'. An improvement of this approach, in order to reflect the Wiki articles more precisely, could be achieved by the usage of power sets instead of simple sets of symbols. Consequently, if within some Wiki article both concepts 'Employee' and 'Manager' are found, the value of the attribute 'People' could be {Employee, Manager}.
3. In order to reflect the inheritance relation, the symbols have to be ordered with a taxonomic relation in exactly the same way as the 'kind-of' relationship. The taxonomic relation provides important information for the calculation of the local similarity. Based on the location of two symbols within taxonomy, a similarity to each other can be expected.
4. The next step is the transformation of non-inheritance relations, which are implicitly defined within the ontology by the strength of connection between concepts. For every relation, an additional attribute within the case model should be introduced. An attribute type has to include all symbols that originate from the names of concepts affected by the relation. The connection strength between concepts from the ontology can be directly taken over as similarity between appropriate symbols.
5. The last step is the accomplishment of the similarity function. The local similarities are already modelled within the previous steps. For the first part of attributes defined in the steps 1-3 the local similarity is given by taxonomy. It can be further adjusted depending on the concrete realisation within the CBR-suite. For the second part of attributes, which encode non-inheritance relations, the local similarity measure is explicitly given by the connection strength between concepts. There is no further necessity to adjust this measure. In order to get the global similarity measure, the local similarities are usually aggregated by the normalised, weighted sum. The weights can be adjusted during the tuning of the search function.

After creation of the case-model, all available Wiki articles should be analysed with text mining software. Hereby, the text miner applies the thesaurus defined within the ontology in order to identify a set of concepts for every Wiki article. The resulting sets of concepts are saved as cases according to the case-model. The following figure shows some example documents with corresponding cases.

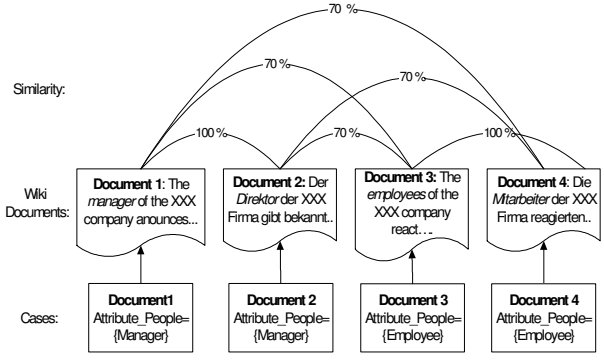


Figure 4 Creation of the case base

The attribute values of cases, which are mapped to Document 1 and Document 2, are equal. The reason for that is the fact that the words 'manager' and 'director' are included in the thesaurus and are mapped to the same concept 'Manager' which is transferred to the value 'Manager' of the attribute 'People' within both cases. As the cases of both documents contain exactly the same attribute values, they have a similarity of 100 %. To compare document 2 and document 3, their cases show attributes with similar values. The reason for that is the fact that attribute values 'Manager' and 'Employee' are similar with respect to the taxonomy in the case model. As a result these cases are rated by similarity of 70 %.

The performance of the similarity based search starts with the input of the search query. With the usage of the text miner and the case model, a new case outgoing from the query is created. Next, the similarity function calculates the similarity of this new case to the ones in the case base and retrieves the set of most similar cases. They are represented to the user in order of relevance.

To realise the search functionality we used the open retrieval engine orenge, which has been developed by the project partner empolis GmbH. It has been created specifically to execute intelligent case-based and knowledge-based searches and for that reason it was useful for our purposes.

5 Outlook and Conclusion

This paper addresses the problem of decreasing acceptance of inner-company Wikis, which occurs when the content becomes large and chaotic. The acceptance by employees decreases on the one hand because of loss of an overview over the available information and on the other hand because of lack of search functionality. Full-text search, which is usually implemented within Wiki-systems, does not support the users sufficiently since the quality of the search-results is low.

The intention of the project described in this paper is to make the application of Wiki for the purpose of knowledge management within companies more attractive by development of improved search functionality.

We introduced a realisation approach to knowledge-based search functionality, which is likely to outperform full-text search. Several characteristics of this approach indicate better search results. But, this statement still has to be proven by a following evaluation. Another interesting idea for a future evaluation is the comparison of the developed search functionality according to other information retrieval approaches.

According to the realised approach, the domain knowledge of the Wiki is represented via ontology, which is created in the semiautomatic manner. For this purpose, the whole document corpus is analysed using concordance programs and, after manual validation, the remaining data can be taken over into the ontology as concepts and relations. Following, after manual validation and extension, the ontology is embedded in a CBR-suite.

Each document from the corpus and each query is semantically annotated with text mining software contained in the CBR-suite, which has access to the constructed domain ontology. Each semantic annotation of any document or any query is regarded as a single CBR-case. The search for the relevant documents is then carried out as a CBR retrieval process.

Based on the meaningful domain model, the quality of search results is expected to increase to a high extent. The provided knowledge guarantees that the annotation is of a high quality and matches the content of the articles. Knowledge-based search copes easily with the weaknesses of full-text search such as "the gap between the user's information need and the actual query strings they specify" [Cesarano *et al.*, 2003]. It finds relevant articles regardless of which synonym is used to formulate the query. Another advantage is the support of multilingualism and different word forms. Results are represented according to relevance; that means not only 100 % hits are displayed, but also articles with related content.

However, it has to be considered that good search results depend exclusively on a good data model. The richer it is the better are the results. As it is of such importance, attention has to be paid that the model is extensive and correct. Furthermore, the search only operates sufficiently if the data model spans the whole context of the provided database. This makes a continuous improvement of the model extremely important. But it has to be considered that this process is extremely time-consuming as well as costly. One approach that can be investigated in a further attempt is the utilisation of the cross-linking characteristic of a Wiki to automatically build and maintain the data model. Generally, effective maintenance is extremely important in order to achieve good results. If done continuously, this guarantees a good search functionality that works within unstructured documents and outranges full-text search to a great extend.

References

[Abecker and Elst, 2004] A. Abecker and van L. Elst van. Ontologies for Knowledge Management. In: Staab, S., Studer, R. (Editors), Handbook on Ontologies, Berlin 2004, Pages 435-454.

[Baeza-Yates, 1999] R. Baeza-Yates. Modern information retrieval. Addison-Wesley, New York 1999.

[Cesarano *et al.*, 2003] C. Cesarano, A. Acierno d', A. Picariello. An Intelligent search Agent System for Semantic Information Retrieval on the Internet: In: Proceedings of the 5th ACM international workshop on Web information and data management. New Orleans, Louisiana, USA, 2003, Pages 111-117.

[Davenport and Grover, 2001] T. Davenport and V. Grover. General Perspectives on Knowledge Management: Fostering a Research Agenda. In: Journal of Management Information Systems, Volume 18, Issue 1, 2001.

[Davenport and Prusak, 1998] T. H. Davenport and L. Prusak, Working Knowledge: How Organisations manage what they know, Harvard Business School Press, Boston, Mass. 1998.

[Leuf and Cunningham, 2001] B. Leuf and W. Cunningham. The Wiki Way Quick Collaboration on the Web, 1. Print, Addison-Wesley, Boston, Mass. [and others] 2001.

[Liu, 2001] H. Liu. Intelligent Search techniques for large software systems. Thesis. Ottawa-Carleton Institute for Computer Science, School of Information Technology and Engineering, University of Ottawa 2001.

[Maedche, 2002] A. Maedche. Ontology Learning for the Semantic Web. 1. Edition, Kluwer Academic, Dordrecht 2002.

[Maletic and Marcus, 2001] J.I. Maletic and A. Marcus. Supporting Program Comprehension Using Semantic and Structural Information. In: Proceedings of 23rd ICSE, Toronto 2001, Pages 103-112.

[Money and Turner, 2004] W. Money and A. Turner. Application of the Technology Acceptance Model to a Knowledge Management System. In: Proceedings of the 37th Hawaii International Conference on System Sciences, 2004.

Flexible Workflows for Knowledge Management in the Digital Design

Mirjam Minor, Daniel Schmalen, Ralph Bergmann
University of Trier
D-54286, Trier, Germany
{minor,schmalen,bergmann}@uni-trier.de

Andreas Koldehoff
sci-worx GmbH, Garbsener Landstr. 10
D-30419, Hannover, Germany
andreas.koldehoff@sci-worx.com

Abstract

This paper presents work in progress on an adaptive workflow management tool for digital design projects. The chip design follows standardized default processes which are adapted during an ongoing project by changing requirements from both design and application uncertainties. Our approach focuses on flexible monitoring and case-based authoring support of adaptive workflows in order to support the knowledge management in a real-world application.

1 Introduction

"If EDA tools* are to assist the semiconductor industry at the 90nm† and 65nm nodes, there must be profound changes to existing tools, and the introduction of new technologies that allow designers to consider and optimize for manufacturing at each stage of the design, verification, tape-out and test process" demands Janusz Rajski, Mentor Graphics [Rajski, 2006]. The chip design in the nano era operates near the physical and technological limits – Infineon is about to develop even 45nm technology until midyear 2008. When starting projects with a new, smaller technology new types of errors may emerge. It is uncertain whether the old algorithms will work well under the new conditions. The requirements for the chip design process are high: A very tight time to market gets in conflict with the need for first time right delivery. The first layout that is manufactured has to be error free in order to avoid respins as the return to the design process and the way back to the chip foundry again are expensive and time consuming. A delay of some weeks leads to a high risk to loose the market. Like the technological imponderabilities, the uncertainty of the customer requirements may cause adaptations of the ongoing chip design process. This complex task requires a careful knowledge management.

This article presents concepts for a tool that supports the repeated reconsideration and adaptation of the ongoing design process by means of flexible workflow technology. It focuses on monitoring and authoring support capabilities for adaptive workflows. Our work is a result of the close collaboration of the University of Trier – as a subcontractor – with the microelectronics company sci-worx in the URANOS project.

Section 2 describes the digital design domain and how workflow instances evolve during the design projects by late and lean modeling. Section 3 deals with a model of context factors for authoring and monitoring purposes. In Section 4, we sketch our ideas for a monitoring that supports risk management. Finally, Section 5 contains a discussion of related work and the steps that we will do next.

*EDA = Electronic Design Automation

†nm = nanometer

2 Incremental and flexible workflow modeling

We aim to support the digital design process by a new workflow tool. It allows an incremental and flexible modeling of design processes. Initial workflow instances following a default workflow definition are adapted during the ongoing project.

2.1 Initial workflow instances

The **design flow** is a standardized description of the step by step design process for all digital design projects of a company. The initial workflow instances are derived from a workflow definition that follows that design flow. *SciWay 2.0* the design flow of sci-worx describes the four high-level phases of a project that consist of several sub-phases each:

1. Specification
2. Implementation and verification
3. FPGA synthesis and validation
4. ASIC synthesis

The specification is more than the customer requirements specification (CRS) and can take up to two month or even longer. The main work after the negotiation of the CRS with the customer is to write the specifications of the implementation, of the verification, and of the validation. The second phase, implementation and verification, includes in parallel the implementation in a Hardware Description Language (HDL), the HDL verification, and the implementation of the validation software. The synthesis is the generation of a layout that goes to the chip foundry. The verification is the functional check against the specification, while the validation (also called 'testing') is more 'hardware-related' and checks, for instance, the resistance to heat or whether a processor boots. Field Programmable Gate Arrays (FPGA's) are designed as a pre-stage of the Application Specific Integrated Circuits (ASIC's). FPGA's follow the building-block concept to be thoroughly validated before the prototype is transformed into an ASIC. Some of the testing tasks within the third and fourth phases can be automated. The literal synthesis is strongly supported by tools as well.

Sometimes, sci-worx needs to execute only a part of the design flow. Some customers come with already finished

specifications; some projects end already after the FPGA synthesis; some customers even perform the FPGA validation on their own.

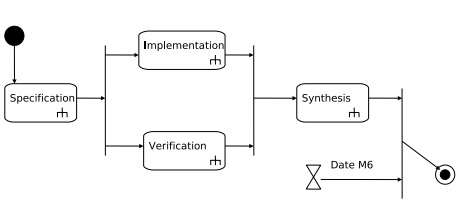


Figure 1: The workflow definition following SciWay 2.0.

Figure 1 shows a sample workflow definition in UML 2.0 notation [Ambler, 2006] following the phases and subphases of the design flow in SciWay 2.0. In terms of the workflow patterns of Aalst et al. [van der Aalst *et al.*, 2003], our workflow modelling language consists of the five basic control flow patterns sequence, parallel split, synchronization, exclusive choice, and simple merge, as well as of structured cycles (loops) and placeholder tasks for sub-workflows. The rounded boxes describe the tasks; the arrows model the sequence of tasks. Some tasks can be performed in parallel like the implementation and verification. The fork symbol in a task hides the sub-diagrams not to be confused with sub-workflows that have an own context (see Section 3). The sand-glass stands for the date of the milestone number 6 that includes the final delivery. Dates are valid for all workflow instances that belong to a project.

The initial workflow instances of a project are generated from a list of modules by means of a standard workflow definition which is more complex than the sample in Figure 1. Modules are sets of functional elements that can be tested as a unit.

2.2 Evolution of workflow instances

Modifications of the workflow instances are mainly triggered by change requests either from the customer or from the designers themselves. Besides changing a date, they may concern three types of structural modifications:

1. add or delete a task
2. split or bundle workflow instances or sub-diagrams
3. reschedule a workflow instance

Modifications within loops are valid for all future iterations. Figure 2 provides an example of a workflow instance that has been modified by an additional task 'Concretisize CRS' namely to select one of two open alternatives from 2-level or 3-level motion estimation. The reason for this modification was a change request from the designers to specify an open feature in the CRS. We decided to model an own task for it rather than a loop backwards to the specification task as the CRS document is part of the contract with the customer and can not be changed. Instead, a change request document is stored in addition to the CRS.

A workflow instance or sub-diagram can be split into several instances and sub-diagrams respectively when the implementation specification is refined or when a change request requires a finer degree of granularity. For instance, when the customer requires a simplified version of a functional element earlier than the sophisticated version, the workflow sub-diagram on this functional element is split

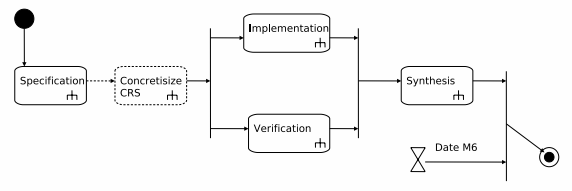


Figure 2: Extend the workflow instance on 'motion estimation' by a new task.

into two sub-diagrams with different milestones. A clone operation supports the workflow modeller in splitting both sub-diagrams or whole workflow instances. In opposite to a normal copy the clone operation skips tasks that have already been completed by the master.

A sample of rescheduling is given in Figure 3 and 4. The adaptations are triggered by a sequence of change requests from the customer. The workflow instance is on the module 'motion vector prediction'.

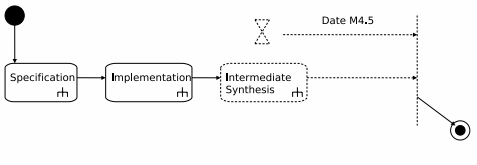


Figure 3: Remove verification and synthesis of 'motion vector prediction'.

The first change request (as depicted in Figure 3) is a result of the fact that the customer requires an accelerated schedule. An intermediate delivery M4.5 with reduced functionality has been arranged. The features will be implemented but not verified until this intermediate delivery. When the verification tasks for the module 'motion vector prediction' have to be finished is still in negotiation.

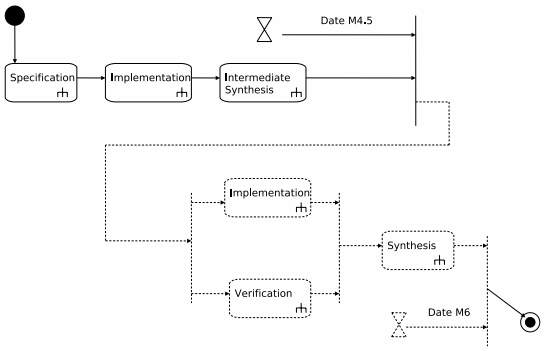


Figure 4: Reschedule verification and synthesis of 'motion vector prediction'.

Later on, a second change request is created that defines that the module shall be fully verified for the final delivery M6. That means that the workflow instance has to be extended by the verification tasks as shown in Figure 4. In parallel, there has to be done some additional implementa-

tion at least for the verification software. And afterward, the synthesis tasks follow.

2.3 Late and lean modeling

At the very beginning of a project, the functional elements of the workflow instances are only rough features from the CRS. During the implementation specification, the workflow instances are refined according to the modules of the future implementation. Later on, change requests trigger adaptations of the particular workflow instances as decribed above.

Due to this *late modeling*, the complexity of the workflow instances increases during the life cycle of a project. The granularity of the functional elements depends on the degree of agility that the project has reached. A rule of thumb is to model only things that have to be done by different persons or that deviate from the standard design flow. We do this in the roughest possible granularity and call it *lean modeling*.

The overall set of workflow instances of a project is organized within a top-level workflow instance that contains the sub-workflows for the modules.

2.4 Suspension of workflow instances

The execution of a workflow instance may take a long time. It has to be suspended during modeling activities by the user. The user may even withdraw an ongoing task from a worklist. We have developed a lock mechanism with breakpoints that are set and released by the user. Setting a breakpoint stops the ongoing tasks that lie within the focus of the breakpoint and suspends the execution of further tasks in the locked area. We have specified rules for each implemented workflow pattern how locks are propagated and released again.

A suspension may endure several weeks, e.g. if a decision of a customer is awaited. When the breakpoint has been lifted, the workflow engine determines where to continue the execution. As we do not model the data flow explicitely this is quite easy.

3 The context model

We extend the workflow model from Section 2 by a context model for authoring support and monitoring purposes. The workflow instances are embedded into a set of partially interdependent context factors. Besides factors from the design context, e.g. the employed EDA tools, we consider also the application context, e.g. the risk that the functionality might change due to the development of the end user requirements and the market. Table 1 shows some samples of context factors.

Context factors are represented as attributes with one or several attribute fields. Each attribute field has exactly one value type defined. The following value types are allowed:

- Boolean
- Single integer value
- Multiple integer value
- Percentage
- Single float value
- Multiple float value
- Date
- Time in hours
- Time in weeks
- Single value from a given pickup list
- Multiple value from a given pickup list
- Free-text
- Risk factor (has two slots: single value from a given pickup list for the *degree of damage*, *probability of occurrence* in percentage)

The value types of the sample context factors in Table 1 including the pickup lists are specified in Table 2. The underlined values are default values. For example, the algorithms risk contains an estimation how many algorithms are still risky and how important this is. An algorithm is risky when it has not yet been implemented with a certain technology. Then the performance and sometimes even the implementability is unknown in advance. The default value for this context factor is 'Cleared'. Interdependencies between units may be specified by means of binary dependencies between context factors.

The context factors are remembered for the following purposes:

- We give *authoring support* for the manual adaptation of workflow instances by retrieving similar workflow instances and presenting their subsequent adaptation steps. The case base consists of pairs of subsequent revisions of past workflow instances while the first revision of the two forms the problem part and the second, with the modification operations that have been performed, the solution part of the case. The context factors contribute to the similarity values for past workflow instances in addition to the structure and the state of the workflow instance. For example, the employed EDA tools are important to identify workflow instances with a similar context.
- Some of the context factors are *risk factors* that may be explicitly monitored by means of the system. An example is the verification grade of the specification that has to be monitored especially when external devices are employed or own IP's are reused. Furthermore, the context acquisition tool provides the utily of *risk analysis* for the current state of a project.
- Before a modification to a sub-workflow is applied the modeller may use the utility of *dependency analysis* that infers the modules (and sub-workflows) that are depending on the module either directly or indirectly.
- Some rather administrative context factors like the dates of milestones are used for *monitoring the state of a design project* (see Section 3).

Following the classification schema for workflow contexts of Maus [Maus, 2001], the context factors for the *authoring support* including the *risk analysis* and the *dependency analysis* belong to the information dimension as well as the explicitly monitored *risk factors* while the factors used for *monitoring the state of a design project* belong to the history dimension.

As for the workflow modeling, the principle of late and lean modeling holds also for the context factors: Only factors are acquired at the beginning of a project that have an important impact on the success of the whole project. Later on, they might be extended by context factors for single workflow instances in order to improve the authoring support. The values of context factors can change when the project progresses. Context factors that are not yet acquired have either empty values or pre-defined default values. We

are planning to implement a tool for the context acquisition with a configurable set of factors. The users may even wish to add context factors during runtime like specifying further milestones. New context factors are restricted to previously well-defined issues.

The monitoring of agile workflow instances is a challenging task. It allows the users to observe the state of the ongoing project and supports the risk management of crucial context factors.

The requirements for the monitoring are:

- to browse the hierarchy of the workflow instances, i.e. the module level and the level of functional elements,
- to allow the user to navigate within one instance, i.e. between tasks and sub-tasks as well as in the temporal dimension,
- to distinguish finished and open tasks, and
- to present the current state of the risk factors.

The *project managers* can use the monitoring to survey the status of the design flow including the change requests. That means the tool is able to support change request management. The annotation of context factors provides support for the risk management of projects. The *designers* can use the monitoring when joining a running project.

4 Related and future work

The CAKE system [Freßmann *et al.*, 2005] handles short term workflows and provides the dynamic assignment of sub-workflows. It uses sets of context factors to perform a case-based retrieval of appropriate sub-workflows. We are planning to extend CAKE for long-term workflows in the URANOS project.

CBRFlow [Weber *et al.*, 2005] is a conversational CBR tool for workflow management that captures the reasons for an adaptation in a dialog with the user. The workflow system ADEPT [Reichert *et al.*, 2003] allows the adaption of processes at both the process instance and the process type level. The implementations of ADEPT and CBRFlow are about to be integrated. In opposite to ADEPT, we do not model the data flow explicitely. Furthermore, the correctness of an adapted instance is not (yet) checked and our workflow definition does not yet evolve. We use the workflow definition for starting a default process that is adapted only at the instance level with both standard variations as illustrated in Figure 2 and ad-hoc changes. Potentially, some standard variations may emerge during the usage of the system and give hints for adapting also the workflow definition in future work. Like CBRFlow, URANOS will employ CBR to support the adaptation of workflow instances. URANOS will use structural CBR rather than conversational CBR, as our users are used to work with complex software tools and are supposed to be faster in specifying attribute values than in formulating questions. An authoring support agent will present previous modifications of similar workflow instances concerning the structure and the context factors of the workflow instance. This aims to decrease the manual effort for adapting workflow instances by reuse. Both CBRFlow and ADEPT do not support long-term suspension of certain areas of a workflow as it is required for the URANOS project.

FRODO task man [van Elst *et al.*, 2003] deals with late modeling of workflows, i.e. the hierarchy of sub-tasks is extensible after enacting a workflow. In URANOS, the functional split of a workflow instance is similar to FRODO's model. In contrast of FRODO, the late modeling will be supported by CBR.

The MTCT system[Bassil *et al.*, 2004] applies ADEPT for the processing of client requests for container transportation. It is related to our approach as it uses templates but in a simpler manner than we do: The templates are only for activities; the overall set of templates is fix. The focus of MTCT lies on time optimization by automatic rescheduling. In our approach, the focus lies on assisting the user who determines the scheduling by means of closed interaction with the customer.

The case handling approach[van der Aalst *et al.*, 2005] is slightly related to our work. It introduces three roles for users: the execute, the redo, and the skip role. Both redo and skip do not make structural changes of ongoing workflow instances. They are possible in our approach as well as in ADEPT, CBRFlow and FRODO.

The main issues of our future work is to implement and evaluate our authoring support concepts in a case study, and to develop a concept for the system's monitoring capabilities.

5 Acknowledgment

The authors acknowledge the Federal Ministry for Education and Science (BMBF) for funding URANOS under grant number 01M3075. We acknowledge the assistance we have received from both Stefan Pipereit, sci-worx and Marko Höpken, sci-worx as well.

References

[Ambler, 2006] Scott W. Ambler. UML 2 Activity Diagrams. Internet: http://www.agilemodeling.com/artifacts/activityDiagram.htm, 2006. [Last visited: February 2006].

[Bassil *et al.*, 2004] Sarita Bassil, Rudolf K. Keller, and Peter G. Kropf. A workflow-oriented system architecture for the management of container transportation. In Jörg Desel, Barbara Pernici, and Mathias Weske, editors, *Business Process Management: Second International Conference, BPM 2004, Potsdam, Germany, June 17-18, 2004. Proceedings*, LNCS 3080, pages 116 – 131. Springer, 2004.

[Freßmann *et al.*, 2005] Andrea Freßmann, Rainer Maximini, and Thomas Sauer. Towards collaborative agent-based knowledge support for time-critical and business-critical processes. In Klaus-Dieter Althoff, Andreas Dengel, Ralph Bergmann, Markus Nick, and Thomas Roth-Berghofer, editors, *Professional Knowledge Management: Third Biennial Conference, WM 2005*, LNAI 3782, pages 420 – 430, Kaiserslautern, Germany, April 2005. Springer-Verlag Berlin Heidelberg 2005.

[Maus, 2001] Heiko Maus. Workflow context as a means for intelligent information support. In Varol Akman, Paolo Bouquet, Richmond H. Thomason, and Roger A. Young, editors, *Modeling and Using Context, Third International and Interdisciplinary Conference, CONTEXT, 2001, Dundee, UK, July 27-30, 2001, Proceedings*, LNCS 2116, pages 261 – 274. Springer-Verlag, 2001.

[Rajski, 2006] Janusz Rajski. Shifting Perspectives on DFM, keynote talk at the International Symposium on Quality Electronic Design 2006. Internet: http://www.isqed.org/, 2006. [Last visited: January 2006].

[Reichert *et al.*, 2003] Manfred Reichert, Stefanie Rinderle, , and Peter Dadam. Adept workflow management system: Flexible support for enterprise-wide business processes (tool presentation). In W. M. P. van der Aalst et al., editor, *Proc. International Conf. on Business Process Management (BPM '03), Eindhoven, The Netherlands, June 2003*, LNCS 2678, pages 370 – 379. Springer Verlag, 2003.

[van der Aalst *et al.*, 2003] Wil M. P. van der Aalst, Arthur H. M. ter Hofstede, Bartek Kiepuszewski, and Alistair P. Barros. Workflow patterns. *Distributed and Parallel Databases*, 14(1):5 – 51, 2003.

[van der Aalst *et al.*, 2005] Wil M. P. van der Aalst, Mathias Weske, and Dolf Grünbauer. Case handling: a new paradigm for business process support. *Data Knowl. Eng.*, 53(2):129 – 162, 2005.

[van Elst *et al.*, 2003] Ludger van Elst, Felix-Robinson Aschoff, Ansgar Bernardi, Heiko Maus, and Sven Schwarz. Weakly-structured workflows for knowledge-intensive tasks: An experimental evaluation. In *12th IEEE International Workshops on Enabling Technologies (WETICE 2003), Infrastructure for Collaborative Enterprises, 9-11 June 2003, Linz, Austria*, pages 340 – 345. IEEE Computer Society, 2003.

[Weber *et al.*, 2005] Barbara Weber, Stefanie Rinderle, Werner Wild, and Manfred Reichert. CCBR-Driven Business Process Evolution. In Héctor Muñoz-Avila and Francesco Ricci, editors, *Case-Based Reasoning, Research and Development, 6th International Conference, on Case-Based Reasoning, ICCBR 2005, Chicago, IL, USA, August 23-26, 2005, Proceedings*, LNAI 3620, pages 610 – 624. Springer, 2005.

Table 1: Sample specification of context factors.

Context category	Attributes					
Risk estimation	End user and market risk			Algorithms risk		
	Technology risk			Tools risk		
Functionality	Clearness of functional specification (list of open formulations)			Uncertainty of customer requirements		
Validation	...					
Verification	Degree of verification concerning specification			Verification capability		
Technology	...					
Tools	EDA tools			Support tools		
Pins and registers	Pins/ports			Registers		
SW/HW-Codesign	Interaction software/hardware					
External devices	External IP			External software		
Design rules	...					
Milestones	M1	M2	M3	M4	M5	M6

Table 2: Sample value types for Table 1.

Attribute name	Data type	Range of values
End user and market risk	Risk	{will be fixed immediately, will cause a short loop will cause a long loop, has fatal consequences}, [0...100]
Algorithms risk	Risk	{will be fixed immediately, will cause a short loop will cause a long loop, has fatal consequences}, [0...100]
...		
EDA tools	multiple value from a pickup list	{Mentor Graphics LeonardoSpectrumTM, Mentor Graphics PrecisionTM RTL Synthesis, Synplicity, Synplify, Synplify Pro, Synopsys FPGA Compiler IITM, ...}
...		
Pins/ports	single integer value	{1, 2, ..., n}
Registers	single integer value	{1, 2, ..., n}
...		

Content Aggregation on Knowledge Bases using Graph Clustering

Christoph Schmitz and **Andreas Hotho** and **Robert Jäschke** and **Gerd Stumme**
Knowledge and Data Engineering Group, Universität Kassel
{lastname}@cs.uni-kassel.de
http://www.kde.cs.uni-kassel.de

Abstract

Recently, research projects such as PADLR and SWAP have developed tools like Edutella or Bibster, which are targeted at establishing peer-to-peer knowledge management systems. In such a system, it is necessary to obtain brief semantic descriptions of peers, so that routing algorithms or matchmaking processes can make decisions about which communities peers should belong to, or to which peers a given query should be forwarded.

This paper provides a graph clustering technique on knowledge bases for that purpose. Using this clustering, we can show that our strategy requires up to 58% fewer queries than the baselines to yield full recall in a bibliographic peer-to-peer scenario.

1 Introduction: Ontology-Based P2PKM

Recently, a lot of effort has been spent on building peer-to-peer systems using semantic web technology [Tane et al., 2004; Ehrig et al., 2003; Bonifacio et al., 2002; Nejdl et al., 2002], based on a notion of peer-to-peer based, personal knowledge management (P2PKM for short). In such a scenario, users will model their knowledge in personal knowledge bases, which can then be shared with other users via a peer-to-peer network.

Many use cases for P2PKM have been implemented recently. In the PADLR and ELENA projects[1], a P2P infrastructure is established for the exchange of learning material; Bibster[2] is a tool for sharing BibTeX entries between researchers; the SCAM tool[3] for knowledge repositories connects to a P2P network. In these systems, each peer builds a knowledge base on top of a common ontology such as LOM and ACM CCS.

One crucial point in such a P2P network is that query messages need to be *routed* to peers which will be able to answer the query without flooding the network with unnecessary traffic. Several proposals have been made recently as to how the network can self-organize into a topology consisting of communities around common topics of interest, a structure which is beneficial for routing, and how messages can be routed in this topology [Schmitz, 2004; Schmitz et al., 2004; Haase and Siebes, 2004; Tempich et al., 2004]. All of these are based on the idea of routing indices [Crespo and Garcia-Molina, 2002]. In a routing index, peers store an aggregated view of their neighbors' contents, enabling them to make content-based routing decisions.

One missing link towards these self-organized network topologies is the extraction of expertises – semantic self-descriptions – of peers from the peers' knowledge bases. In this paper, a method of extracting these expertises using a clustering technique on the knowledge base is proposed and evaluated.

The remainder of this paper is structured as follows: After a brief review of an ontology-based P2P knowledge management scenario and related work, we will introduce technical preliminaries in Section 2. In Section 3 the automatic generation of self-descriptions of peers' knowledge bases through the use of graph clustering will be demonstrated. Section 4 presents evaluation results for a bibliographic P2PKM scenario. Section 5 concludes and discusses future work.

This paper has first been published at ESWC 2006 [Schmitz et al., 2006].

1.1 Related Work

To the best of our knowledge, the exact problem discussed in this paper has not been treated before. There are, however, related areas which touch similar topics.

Knowledge-rich approaches from the text summarization area [Hovy and Lin, 1999; Hahn and Reimer, 1999] use algorithms on knowledge representation formalism to extract salient topics from texts in order to generate summaries. We compare our approach to the one in [Hovy and Lin, 1999] in Section 4.

In semantic P2P overlays, peers need some means of obtaining a notion of other peers' contents for routing tables and other purposes. [Löser et al., 2005] and others rely on observing the past behavior of peers – queries sent and answered – to guess what kind of information peers contain, including some fallback strategies to overcome the bootstrapping problem. In [Haase and Siebes, 2004], peers publish their expertise containing *all* topics they have information about without any aggregation, which will be a resource consumption problem for larger knowledge bases and networks.

Keyword-based P2P information retrieval systems can make use of the bag-of-words or vector-space models for IR. [Reynolds and Vahdat, 2003] proposes the use of Bloom filters to maintain compact representations of contents for routing purposes. These techniques, however, do not provide a semantically aggregated view of the contents,

[1] http://www.l3s.de/english/projects/projects_overview.html
[2] http://bibster.semanticweb.org
[3] http://scam.sourceforge.net/

but rather a bitwise superposition of keywords which loses semantic relationships between related keywords.

Much work has been done on graph clustering (e. g. [Pothen, 1997]) in a variety of areas. Most of these algorithms, though, do not readily yield representatives such as the centroids from the k-modes algorithm used in Section 3, and/or may not be naturally adapted to the shared-part/personal-part consideration used in Section 2.3.

2 Basics and Definitions

2.1 P2P Network Model

As in [Schmitz, 2004], the following assumptions are made about about peers in a P2PKM network:

- Each peer stores a set of *content items*. On these content items, there exists a *similarity function* called sim. We assume $sim(i, j) \in [0, 1]$ for all items i, j, and the corresponding *distance function* $d := 1 - sim$ shall be a metric. For the purpose of this paper, we assume *content items* to be entities from a knowledge base (cf. Section 2.2), and the metric to be defined in terms of the ontology as described in Section 2.4.
- Each peer provides a self-description of what its knowledge base contains, in the following referred to as *expertise*. Expertises need to be much smaller than the knowledge bases they describe, as they are transmitted over the network and used in other peers' routing indices. A method of obtaining these expertises is outlined in Section 3. Formally, an expertise consists of a set $\{(c_i, w_i)|i = 1 \dots k\}$ of pairs mapping content items c_i to real-valued weights w_i.
- There is a relation $knows$ on the set of peers. Each peer knows a certain set of other peers, i. e., it knows their expertises and network address (e. g. IP address, JXTA ID, ...). This corresponds to the routing index as proposed in [Crespo and Garcia-Molina, 2002]. In order to account for the limited amount of memory and processing power, the size of the routing index at each peer is limited.
- Peers query for content items on other peers by sending query messages to some or all of their neighbors; these queries are forwarded by peers according to some *query routing strategy*, which uses the sim function mentioned above to decide which neighbors to forward messages to.

2.2 Ontology Model

For the purpose of this paper, we use the view on ontologies proposed by the KAON framework [Ehrig et al., 2002]. Following the simplified nomenclature of [Ehrig et al., 2002], an *ontology* consists of *concepts* with a subclassOf partial order, and relations between concepts. A *knowledge base* consists of an ontology and *instances* of concepts and relations. Concepts and instances are both called *entities* (for details cf. [Ehrig et al., 2002]).

Another important feature of KAON is the inclusion mechanism for knowledge bases, enabling the implementation of the shared and personal parts of knowledge bases as introduced in the next section.

2.3 Shared and Personal Parts of the Knowledge Bases

Based on the use cases mentioned in Section 1, all peers $P_i, i = 1 \dots n$, in the system are assumed to *share* a certain part O of their ontologies: in the case of e-learning, this could be the Learning Object Metadata (LOM)[4] standard plus a classification scheme; when exchanging bibliographic metadata as in Bibster, this would be an ontology reflecting BIBTEX and a classification scheme such as ACM CCS[5], etc.

Additionally, the knowledge base KB_i of each peer P_i contains *personal* knowledge PK_i which is modeled by the user of the peer and is not known a-priori to other peers. Querying this knowledge efficiently and sharing it among peers is the main task of the P2PKM system. Formally, we can say that for all i, $KB_i = O \cup PK_i$.

In Figure 1, the ontology used in the evaluation in Section 4 is shown. In this case, the shared part O comprises the concepts Person, Paper, Topic, and their relations, as well as the topics of the ACM CCS. The personal knowledge PK_i of each peer contains instantiations of papers and persons and their relationships to each other and the topics for the papers of each individual author in DBLP with papers in the ACM digital library (cf. 4.1 for details).

For the purpose of this paper, an agreement on a shared ontology O is assumed. The problem of ontologies emerging in a distributed KM setting [Aberer et al., 2003], of ontology alignment, mapping, and merging [de Bruijn et al., 2004], are beyond the scope of this work.

2.4 Ontology-Based Metrics

An ontology of the kind we use is a labeled, directed graph: the set of nodes comprises the entities, and the relations between entities make up the set of edges. An edge between entities in this graph expresses relatedness in some sense: the instance `paper37` may have an instanceOf edge to the concept `Paper`, `Paper` and `Topic` would be connected by an edge due to the `hasTopic` relation, etc.

On this kind of semantic structure, [Rada et al., 1989] has proposed to use the distance in the graph-theoretic sense (length of shortest path) as a semantic distance measure.

Metric Used in the Evaluation

We follow this suggestion and apply it to the abovementioned graph as follows:

- To each edge, a length is assigned; taxonomic edges (instanceOf, subclassOf) get length 1, while non-taxonomic edges are assigned length 2. This reflects the fact that `subclassOf(PhDStudent, Person)` is a closer link between these concepts than, say, `rides(Person, Bicycle)`.
- Edge lengths are divided by the average distance of the incident nodes from the root concept. This reflects the intuition that top-level concepts such as `Person` and `Project` would be considered less similar than, e.g., `Graduate Student` and `Undergraduate` farther from the root.

Similarity, Relatedness, and Semantic Distances – Why Edge Counting?

The notions of semantic similarity (things having similar features) and relatedness (things being associated with each other) have long been explored in various disciplines such as linguistics and cognitive sciences. Discussions about these phenomena and their respective properties have lasted for decades (cf. [Tversky, 1977; Gentner and Brem, 1999]).

[4] http://ltsc.ieee.org/wg12

[5] http://www.acm.org/class

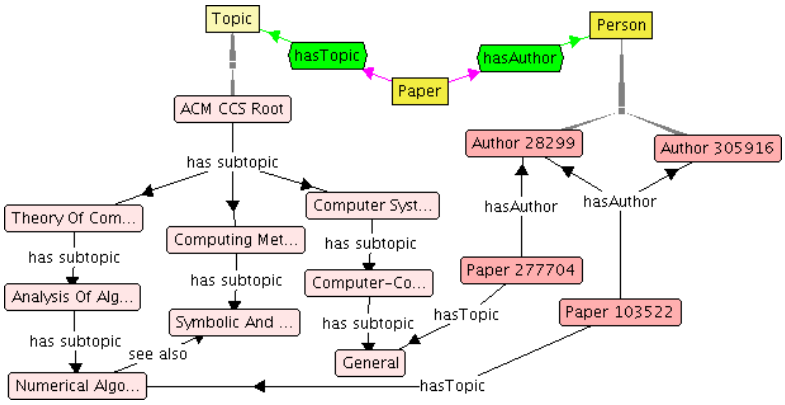


Figure 1: Example Knowledge Base

While most of this discussion is outside the scope of this paper, some key points [Gentner and Brem, 1999] are worth mentioning: Thematic relatedness and similarity are distinct phenomena, but both can get mixed up or influence each other.

In the context of this paper, where the goal is to provide self-descriptions of knowledge in a P2PKM system, some more influences on the choice of the semantic distance should be noted:

- The ontologies to be used in P2PKM will be engineered specifically for KM purposes. Thus, regarding a relation between two concepts as an indication that these two have something to do with each other reflects the intention of a knowledge engineer to express relatedness.
- In a P2PKM system, domain specific ontologies will be used. These represent a conceptualization of a small part of the world which is relevant for the given domain, so that stray associations such as *lamp – round glowing object – moon –* ..., which might occur in a "world ontology", will be avoided.
- Modeling idiosyncrasies of certain tools and formalisms such as described in the next section need to be anticipated. This can be done by allowing for flexible weighting and filtering strategies.

Various constraints are present on other kinds of metrics which have led to the use of an edge-counting metric for the purpose of this paper. Approaches such as [Resnik, 1995] or [Tversky, 1977] assume the presence of full text or linguistic background knowledge; others such as [Maedche and Staab, 2002] only use concepts and an instanceOf relationship, neglecting instances and non-taxonomic relationships altogether. To yield maximum flexibility and to use as much of the modeled content as possible, an edge counting approach was chosen for this paper.

Keeping this discussion in mind, one needs to be aware of what kinds of similarity and/or relatedness should be expressed in modeling the ontology and parameterizing the metric.

Pitfalls on Real-World Ontologies

While the edge-counting metric seems straightforward, applying it to real-world ontologies turned out to be non-trivial:

Noise and Technical Artifacts. Often not all of the content of a knowledge base is used to model a certain domain as such; e. g., in KAON, lexical information is represented as first-class entities in the knowledge base. This yields entities which are not relevant for the semantic distance computation. There is also a root class which every entity is an instance of, which would render our approach to calculating distances useless.

Modeling Idiosyncrasies. Engineering an ontology implies design decisions, e. g. whether to model something as an instance or as a concept [Welty and Ferrucci, 1994]. These decisions carry implications for the weighting of edges, e. g. when taxonomic relationships are expressed by a relation which is not one of `instanceOf`, `subclassOf`.

To overcome these problems, we have implemented extensive entity filtering and weighting customization strategies which are applied prior to the metric computation itself.

Choice of Parameters

One obvious question is where the parameters, weighting schemes and filtering rules necessary for this kind of metric should come from. These can be agreed upon just like the ontology to be used itself. When stakeholders deside that there should be a "see also" relation between topics, they could also agree on its importance or non-importance for retrieval tasks (cf. the discussion about the value of non-taxonomic relations in [Rada et al., 1989]).

Secondly, this kind of semantic metric will not primarily be used to reflect human judgment of similarity or relatedness directly, but to structure a network topology. For this type of use, optimal parameters can be determined in simulation experiments or might be learned over the lifetime of the system.

2.5 k-Modes Clustering

In Section 3, we will use an extension of k-modes clustering [Huang, 1998] to obtain aggregations of knowledge bases. The basic version of k-modes clustering for partitioning a set S of items into k clusters $S_1, \ldots, S_k$ such that $S = \bigcup_i S_i$ works as follows:

1. Given k, choose k elements $C_i, i = 1 \ldots k$ of S as *centroids*
2. Assign each $s \in S$ to the cluster S_i with $i = \arg\min_j d(C_j, s)$
3. For $i = 1 \ldots k$, recompute C_i such that $\sum_{s \in S_i} d(C_i, s)$ is minimized.
4. Repeat steps 2 and 3 until centroids converge.

This algorithms yields (locally) optimal centroids which minimize the average distance of each centroid to its cluster members. A variation we will use is *bi-section k-modes clustering*, which produces k clusters by starting from an initial cluster containing all elements, and then recursively splitting the cluster with the largest variance with 2-modes until k clusters have been reached.

As the algorithm is randomized, it may happen that a cluster cannot be split although k clusters have not been reached. In that case, we retry a fixed number of times before accepting the clustering.

3 Graph Clustering for Content Aggregation

As mentioned in the motivation, a peer needs to provide an expertise in order to be found as an information provider in a P2PKM network. From the discussion above, the following requirements for an expertise can be derived:

- The expertise should provide an aggregated account of what is contained in the knowledge base of the peer, meaning that using the similarity function, a routing algorithm can make good a-priori guesses of what can or cannot be found in the knowledge base. More specifically, the personal part PK_i should be reflected in the expertise.
- The expertise should be much smaller than the knowledge base itself, preferably contain only a few entities, because it will be used in routing indices and in computations needed for routing decisions.

With these requirements in mind, we propose the use of a clustering algorithm to obtain an expertise for each peer.

3.1 Clustering the Knowledge Base

We use a version of bi-section k-modes clustering for the extraction of such an expertise. As mentioned before, k-modes clustering yields centroids which are locally optimal elements of a set regarding the average distance to their cluster members.

Using the semantic metric, these centroids fulfill the abovementioned requirements for an expertise: We can compute a *small* number of centroids, which are – on the average – *semantically close* to every member of their respective clusters, thus providing a good *aggregation* of the knowledge base.

In order to apply this algorithm in our scenario, however, some changes need to be made:

- The set S to be clustered has to consist only of the *personal parts* PK_i of the knowledge bases. Otherwise, the structure of the shared part (which may be comparatively large) will shadow the interesting structures of the personal part.
- The *centroids* C_i will not be chosen from the whole knowledge base, but only from the shared part O of the ontology. Otherwise, other peers could not interpret the expertise of a peer.

The expertise for each knowledge base is obtained by clustering the knowledge base as described, obtaining a set $\{C_i \mid i = 1 \dots k\} \subseteq O$ of entities from the ontology as centroids for a given k. The expertise then consists of the pairs $\{(C_i, |S_i|) | i = 1 \dots k\}$ of centroids and cluster sizes. Because we restricted the choice of centroids to be from O, we get expertises that other peers can interpret from clustering the elements of KB_i.

3.2 Determining the number of centroids

One problem of the k-modes algorithm is that one needs to set the value of k beforehand. As the appropriate number of topics for a given knowledge base may not be known a-priori, we use the *silhouette coefficient* [Kaufman and Rousseeuw, 1990], which is an indicator for the quality of the clustering. In short, it determines how well clusters are separated in terms of the distances of each item to the nearest and the second nearest centroid: if each item is close to its own centroid and far away from the others, the silhouette coefficient will be large, indicating a good clustering.

4 Experimental Evaluation

In the following sections, we will try to verify three hypotheses:

1. Extracting a good expertise from a knowledge base is harder for large knowledge bases.
2. With larger expertises, the retrieval results improve.
3. The clustering strategy extracts expertises which are useful for retrieval.

The intuition is as follows: Extracting a good expertise from a large knowledge base is harder than from a small one, as the interests of a person interested in many areas will be more difficult to summarize than those of someone who has only few fields of interest. With larger expertises, the retrieval results improve, because if we spend more space (and processing time) for describing someone's interests, we can make better guesses about what his knowledge base contains. As the clustering strategy tries to return the centroids which are as close as possible to all cluster members, we assume that it gives a good approximation of what a knowledge base contains.

4.1 Setup

To evaluate the usefulness of the expertise extraction approach from the previous sections, we consider a P2PKM scenario with a self-organized semantic topology as described in [Schmitz, 2004; Haase and Siebes, 2004; Tempich et al., 2004]: the expertises of peers are stored in routing tables, where similarity computations between queries and expertises in the routing indices are used to make greedy routing decisions when forwarding queries.

If the routing strategy of this network works as intended, the peers which published an expertise closest to a given query will be queried first. In the following experiment, the quality of the expertises is evaluated in isolation based on that observation: An expertise was extracted for each peer. All of the shared entities of the ontology were used in turn as queries. For each query, the authors were sorted in descending similarity of the closest entity of the expertise to the query. Ties were resolved by ordering in decreasing weight order.

The evaluation is based on the bibliographic use case mentioned in Section 1: there are scientists in the P2P network sharing bibliographic information about their publications. An ontology according to Figure 1 is used. Only the top level concepts (Person, Topic, Paper) and the ACM classification hierarchy are shared among the peers. Each user models a knowledge base on his peer representing his own papers.

We instantiated such a set of knowledge bases using the following data:

- For 39067 papers from DBLP which are present in the ACM Digital Library, the topics were obtained from the ACM website. There are 1474 topics in the ACM Computing Classification System. Details on the construction of the data set and the conversion scripts can be found on http://www.kde.cs.uni-kassel.de/schmitz/acmdata.
- To yield non-trivial knowledge bases, only those authors who wrote papers on at least 10 topics were considered. This left 317 authors. A discussion of this pruning step can be found in Section 4.3.

For each of the summarization strategies described below, we show the number of authors which had to be queried in order to yield a given level of recall. This is an indicator for how well the expertises capture the content of the authors' knowledge bases: the better the expertises, the fewer authors one needs to ask in order to reach a certain level of recall.

This is a variation of the usual precision-against-recall evaluation from information retrieval. Instead of precision – how many of the retrieved documents are relevant? – the relative number of the queried authors which are able to provide papers on a given topic is measured.

4.2 Expertise Extraction Strategies

In comparison with the clustering technique from Section 3, the following strategies were evaluated. The expertise size was fixed to be 5 except where noted otherwise.

Counting (#5): The occurrences of topics in each author's knowledge base were counted. The top 5 topics and counts were used as the author's expertise.

Counting Parents (#P5): As above, but each topic did not count for itself, but for its parent topic.

Random (R5): Use 5 random topics and their counts.

Wavefront (WFL7/WFL9): Compute a wavefront of so-called *fuser concepts* [Hovy and Lin, 1999]. A fuser concept is a concept many descendants of which are instantiated in the knowledge base. The intuition is that if many of the descendants of a concept occur, it will be a good summary of that part of the knowledge base. If only few children occur, a better summarization would be found deeper in the taxonomy.

There are two parameters in this computation: a threshold value between 0 and 1 for the *branch ratio* (the lower the branch ratio, the more salient the topic), and a minimal depth for the fuser concepts. There are some problems in comparing this strategy with the other strategies named here:

- It is not possible to control the number of fuser concepts returned with the parameters the strategy offers.
- Leaves can never be fuser concepts, which is a problem in a relatively flat hierarchy such as ACM CCS, where many papers are classified with leaf concepts.
- All choices of parameters yielded very few fuser concepts.

The expertise consisted of the fuser concepts as returned by the wavefront computation with the inverse of the branch ratio as weights. If the number of fuser concepts was less than 5, the expertise was filled up with the leaf concepts occurring most frequently. We examined thresholds of 0.7 (WFL7) and 0.9 (WFL9) with minimal depth 1.

Clustering (C5/C37): The expertise consisted of centroids and cluster sizes determined by a bisection-k-modes clustering as described in Section 3. C5 used a fixed k of 5, while C37 selected the best $k \in \{3, \ldots, 7\}$ using the silhouette coefficient. 20 retries were used in the bi-section k-means computation.

4.3 Results

In this section, results are presented for the different strategies. The values presented are averaged over all queries (i. e. all ACM topics), and, in the cases with randomized algorithms (C5, C37, R5), over 20 runs.

Note that all strategies except C37 returned expertises of size 5, while in C37, the average expertise size was slightly larger at 5.09. Table 3 shows the distribution of expertise sizes for C37.q

Pruning of the Evaluation Set

In order to yield interesting knowledge bases to extract expertises from, we pruned the ACM/DBLP data set as described in Section 4.1. Thus, only the knowledge bases of authors which have written papers on at least 10 topics were considered.

Table 1: Full vs. pruned data: Fraction of authors (%) queried to yield given recall, C5 strategy

Recall	full data	pruned data
10%	0.01	4.09
30%	0.04	4.93
50%	0.07	6.43
70%	0.16	12.53
90%	0.55	18.73
100%	3.45	22.88

Table 1 presents a comparison of the full and the pruned dataset for the C5 strategy. It can be seen that the full data require querying only a fraction of the authors which is one or two orders of magnitude *smaller* than the pruned data. This indicates that the first hypothesis holds; the pruning step yields the "hard" instances of the problem.

Influence of the Expertise Size

Intuitively, a larger expertise can contain more information about the knowledge base than a smaller one. In the extreme case, one could use the whole knowledge base as the expertise.

To test the second hypothesis, Figure 2 and Table 2, show the influence of the expertise size on retrieval performance for the C5 clustering strategy.

Table 2: Percentage of Authors Queries against Expertise Size (C5 Strategy)

	Expertise Size				
Recall	1	3	5	7	10
10%	15.06	6.80	4.09	3.38	3.03
30%	17.66	8.16	4.93	4.12	3.69
50%	21.79	10.59	6.43	5.35	4.82
70%	33.37	19.79	12.53	10.21	9.18
90%	44.57	28.20	18.73	15.44	14.15
100%	49.07	33.04	22.88	19.10	17.67

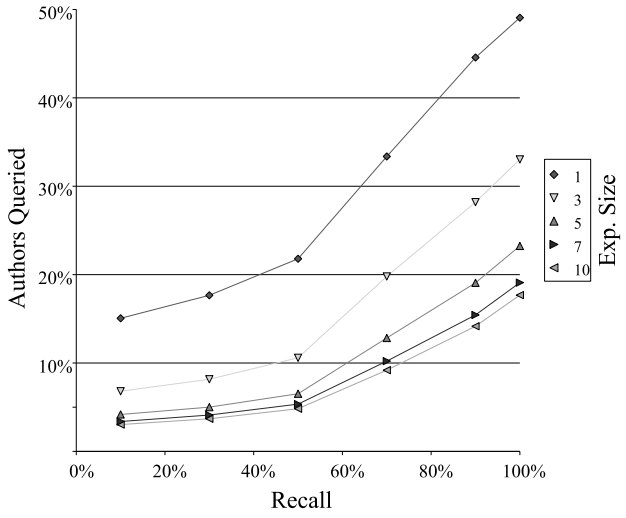


Figure 2: Influence of Expertise Size (C5 Strategy)

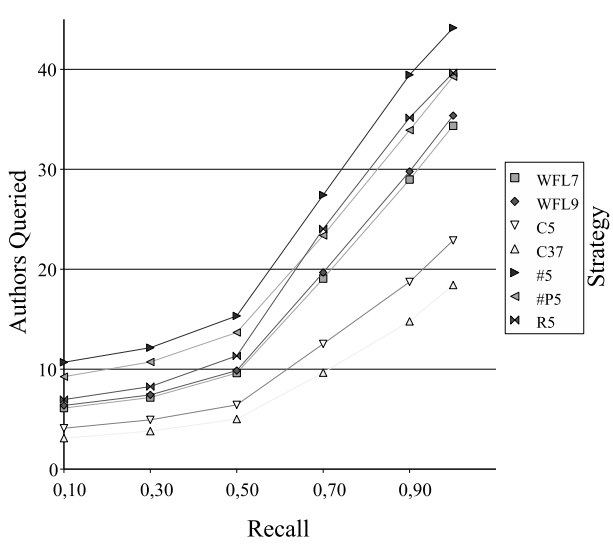


Figure 3: Percentage of Authors Queried against Recall

Table 3: Distribution of Expertise Sizes for C37

Exp. Size	Percentage of Authors
3	20%
4	15%
5	21%
6	23%
7	21%
Avg.: 5.09	

While the small number of data points for each recall level do not lend themselves to a detailed quantitative analysis, it is clear that the expertise size has the expected influence in the clustering technique: the larger the expertise is, the more detail it can provide about the knowledge base, and the better the retrieval performance is.

Note that the resources a peer would be willing to spend on storing routing tables and making routing decisions are limited, so that a trade-off between resources set aside for routing and the resulting performance must be made, especially as network and knowledge base sizes grow larger.

Influence of the Summarization Strategy

Finally, we evaluate the performance of the clustering strategies against the other strategies mentioned above.

Table 4 and Figure 3 show that the k-modes clustering compares favorably against the other strategies: fewer authors need to be asked in order to find a given proportion of the available papers on a certain topic. This is an indication that the clustering technique will yield expertises which can usefully be applied in a P2PKM system with a forwarding query routing strategy based on routing indices. For example, to yield 100% recall, 58% fewer (18.42% vs. 44.15%) peers would have to be queried when using C37 instead of the #5 strategy. With C37 and a routing strategy that contacted best peers first, $100\% - 18.42\% \approx 81\%$ of the peers could be spared from being queried while still getting full recall.

The standard deviations σ of the randomized strategies given in Table 4 show that while the actual results of the C5, C37, and R5 runs may vary, the quality of the results for querying is stable.

To get an impression about why the clustering strategies work better than the others, consider one author whose papers are labelled with the following topics[6]: B.5, B.6, B.6, B.6.1.a, B.6.1.a, B.6.3.b, B.7, B.7.1.c, B.8, B.8, C.0.d, C.3.e, C.5.3.f, D.3.2, G.1, I.5.4.g, J.

The different strategies delivered the results shown in Table 5. It can be seen that the clustering strategies find the best balance between spreading the expertise over all occuring topics, and on the other hand generalizing so that many occuring topics are subsumed under one expertise entry. This happens due to the way the clustering strategy spreads the clusters over the ontology graph, maxminizing the coherence within clusters. Most other strategies, for example, did not consider any of the topics outside the B and C parts of ACM CCS.

5 Summary and Outlook

5.1 Conclusion

In this paper, an algorithm which can be used to extract semantic summaries – called *expertises* – from knowledge bases is proposed. A motivation for the necessity of this kind of summary is given, namely, that such summaries are needed for routing tables in semantic P2P networks.

We demonstrate that the clustering method outperforms other strategies in terms of queries needed to get a given recall on a set of knowledge bases from a bibliographic scenario. We also show qualitatively that larger knowledge bases are harder to summarize, and that larger expertises are an advantage in determining which peers to query.

5.2 Outlook and Work in Progress

Evaluation in Context. This paper provides evidence that the clustering procedure extracts suitable expertises for a P2PKM setting. The next step will be combining the clustering with self-organization techniques for P2PKM networks as described in [Schmitz, 2004]. Note that usually the value of aggregations or summaries is measured by evaluating it against human judgment. In our case, however, the aggregations will be evaluated with regard to their contribution to improving the performance of the P2P network.

[6]Note that the fourth level topics do not have names of their own originally; we attached artificial IDs to distinguish them

Table 4: Percentage of Authors Queried against Recall; σ: Standard Deviation

	Authors Queried						
Recall	WFL7	WFL9	C5 (σ)	C37 (σ)	#5	#P5	R5 (σ)
10%	6.11	6.37	4.09 (.28)	**3.10** (.18)	10.69	9.25	6.96 (.48)
30%	7.16	7.43	4.93 (.28)	**3.80** (.19)	12.15	10.72	8.26 (.52)
50%	9.61	9.86	6.43 (.32)	**5.01** (.21)	15.33	13.67	11.33 (.61)
70%	19.06	19.67	12.53 (.52)	**9.65** (.33)	27.43	23.38	24.04 (.82)
90%	28.97	29.78	18.73 (.64)	**14.78** (.47)	39.45	33.91	35.16 (.93)
100%	34.35	35.37	22.88 (.75)	**18.42** (.48)	44.15	39.27	39.65 (.83)

Table 5: Sample Results for Different Strategies

#P5	#5	R5	WFL7	WFL9	C5	C37
B. (6)	B.6.1.a (2)	B.6.1.a (2)	C. (3)	C. (3)	B. (11)	B.6 (10)
B.6.1 (2)	B.6 (2)	B.6.3.b (1)	B.6.1.a (2)	B.6.1.a (2)	C. (3)	C. (3)
B.6.3 (1)	B.8 (2)	D.3.2 (1)	B. (2)	B. (2)	I.5.4.g (1)	J. (1)
C.0 (1)	B.6.3.b (1)	C.5.3.f (1)	B.6 (1.5)	B.6 (1.5)	D.3.2 (1)	G.1 (1)
B.7.1 (1)	B.5 (1)	B.7 (1)	B.6.3.b (1)	B.6.3.b (1)	G.1 (1)	D.3.2 (1)
						I.5.4.g (1)

Scalability Issues. Computing the metric as described above is very expensive, as it needs to compute all-pairs-shortest-paths. For large ontologies having tens or hundreds of thousands of nodes, this is prohibitively expensive. In the current evaluation, the shortest paths needed are computed on the fly, but for a real-world P2PKM implementation, some faster solution needs to be found. The obvious idea of pre-computing the metric does not mitigate the problem very much, because maintaining the shortest path lengths requires $O(n^2)$ storage.

On possible direction of investigation is to look at the actual usage of the metric in a P2PKM system. If the community structure of the network leads to a locality in the use of the metric, caching and/or dynamic programming strategies for the metric computation may be feasible.

Test Data and Evaluation Methodology. Other than in Information Retrieval, for example, there are neither widespread testing datasets nor standard evaluation methods available for Semantic Web and especially P2PKM applications. In order to compare and evaluate future research in these areas, standardized data sets and measures need to be established.

Acknowledgement. Part of this research was funded by the EU in the Nepomuk project (FP6-027705).

References

[Aberer et al., 2003] Aberer, K., Cudré-Mauroux, P., and Hauswirth, M. (2003). The Chatty Web: Emergent Semantics Through Gossiping. In *Proc. 12th International World Wide Web Conference*, Budapest, Hungary.

[Bonifacio et al., 2002] Bonifacio, M., Cuel, R., Mameli, G., and Nori, M. (2002). A peer-to-peer architecture for distributed knowledge management. In *Proc. 3rd International Symposium on Multi-Agent Systems, Large Complex Systems, and E-Businesses MALCEB'2002*, Erfurt, Germany.

[Crespo and Garcia-Molina, 2002] Crespo, A. and Garcia-Molina, H. (2002). Routing indices for peer-to-peer systems. In *Proc. International Conference on Distributed Computing Systems (ICDCS)*, Vienna, Austria.

[de Bruijn et al., 2004] de Bruijn, J., Martin-Recuerda, F., Manov, D., and Ehrig, M. (2004). State-of-the-art survey on ontology merging and aligning (SEKT project deliverable 4.2.1). http://sw.deri.org/~jos/sekt-d4.2.1-mediation-survey-final.pdf.

[Ehrig et al., 2003] Ehrig, M., Haase, P., van Harmelen, F., Siebes, R., Staab, S., Stuckenschmidt, H., Studer, R., and Tempich, C. (2003). The SWAP data and metadata model for semantics-based peer-to-peer systems. In Schillo, M., Klusch, M., Müller, J. P., and Tianfield, H., editors, *Proc. MATES-2003. First German Conference on Multiagent Technologies*, volume 2831 of *LNAI*, pages 144–155, Erfurt, Germany. Springer.

[Ehrig et al., 2002] Ehrig, M., Handschuh, S., Hotho, A., et al. (2002). KAON - towards a large scale Semantic Web. In Bauknecht, K., Tjoa, A. M., and Quirchmayr, G., editors, *Proc. E-Commerce and Web Technologies, Third International Conference, EC-Web 2002*, number 2455 in LNCS, Aix-en-Provence

[Gentner and Brem, 1999] Gentner, D. and Brem, S. K. (1999). Is snow really like a shovel? Distinguishing similarity from thematic relatedness. In Hahn, M. and Stoness, S. C., editors, *Proc. Twenty-First Annual Meeting of the Cognitive Science Society*, Mahwah, NJ.

[Haase and Siebes, 2004] Haase, P. and Siebes, R. (2004). Peer selection in peer-to-peer networks with semantic topologies. In *Proc. 13th International World Wide Web Conference*, New York City, NY, USA.

[Hahn and Reimer, 1999] Hahn, U. and Reimer, U. (1999). Knowledge-based text summarization: Salience and generalization operators for knowledge base abstraction. In Mani, I. and Maybury, M. T., editors, *Advances in Automatic Text Summarization*. MIT Press.

[Hovy and Lin, 1999] Hovy, E. and Lin, C.-Y. (1999). Automated text summarization in SUMMARIST. In Mani, I. and Maybury, M. T., editors, *Advances in Automatic Text Summarization*. MIT Press.

[Huang, 1998] Huang, Z. (1998). Extensions to the k-means algorithm for clustering large data sets with categorical values. *Data Min. Knowl. Discov.*, 2(3):283–304.

[Kaufman and Rousseeuw, 1990] Kaufman, L. and Rousseeuw, P. J. (1990). *Finding Groups in Data: An Introduction to Cluster Analysis*. John Wiley.

[Löser et al., 2005] Löser, A., Tempich, C., Quilitz, B., Staab, S., Balke, W. T., and Nejdl, W. (2005). Searching dynamic communities with personal indexes. In Gil, Y., Motta, E., Benjamins, V. R., and Musen, M. A., editors, *Proc. 4th International Semantic Web Conference, ISWC 2005*, Galway, Ireland.

[Maedche and Staab, 2002] Maedche, A. and Staab, S. (2002). Measuring similarity between ontologies. In *Proc. Of the European Conference on Knowledge Acquisition and Management - EKAW-2002. Madrid, Spain, October 1-4, 2002*, volume 2473 of *LNCS/LNAI*. Springer.

[Nejdl et al., 2002] Nejdl, W., Wolf, B., Qu, C., Decker, S., Naeve, A., Sintek, M., Nilsson, M., Risch, T., and Palmér, M. (2002). Edutella: A P2P networking infrastructure based on RDF. In *Proc. 11th International World Wide Web Conference (WWW 2002)*, Honolulu, Hawaii.

[Pothen, 1997] Pothen, A. (1997). Graph partitioning algorithms with applications to scientific computing. In Keyes, D. E., Sameh, A., and Venkatakrishnan, V., editors, *Parallel Numerical Algorithms*, pages 323–368. Kluwer.

[Rada et al., 1989] Rada, R., Mili, H., Bicknell, E., and Blettner, M. (1989). Development and application of a metric on semantic nets. *IEEE Transactions on Systems, Man and Cybernetics*, 19(1):17–30.

[Resnik, 1995] Resnik, P. (1995). Using information content to evaluate semantic similarity in a taxonomy. In *Proc. Fourteenth International Joint Conference on Artificial Intelligence, IJCAI 95*, Montreal, Canada.

[Reynolds and Vahdat, 2003] Reynolds, P. and Vahdat, A. (2003). Efficient peer-to-peer keyword searching. In Endler, M. and Schmidt, D. C., editors, *Middleware*, volume 2672 of *Lecture Notes in Computer Science*. Springer.

[Schmitz, 2004] Schmitz, C. (2004). Self-organization of a small world by topic. In *Proc. 1st International Workshop on Peer-to-Peer Knowledge Management*, Boston, MA.

[Schmitz et al., 2006] Schmitz, C., Hotho, A., Jäschke, R., and Stumme, G. (2006). Content aggregation on knowledge bases using graph clustering. In *Proceedings of the 3rd European Semantic Web Conference*, Budva, Montenegro.

[Schmitz et al., 2004] Schmitz, C., Staab, S., and Tempich, C. (2004). Socialisation in peer-to-peer knowledge management. In *Proc. International Conference on Knowledge Management (I-Know 2004)*, Graz, Austria.

[Tane et al., 2004] Tane, J., Schmitz, C., and Stumme, G. (2004). Semantic resource management for the web: An elearning application. In *Proc. 13th International World Wide Web Conference*, New York.

[Tempich et al., 2004] Tempich, C., Staab, S., and Wranik, A. (2004). Remindin': Semantic query routing in peer-to-peer networks based on social metaphors. In W3C, editor, *Proceedings of the 13th International World Wide Web Conference (WWW 2004)*, pages 640–649, New York, USA. ACM.

[Tversky, 1977] Tversky, A. (1977). Features of similarity. *Psychological Review*, 84(4):327–352.

[Welty and Ferrucci, 1994] Welty, C. A. and Ferrucci, D. A. (1994). What's in an instance? Technical Report #94-18, RPI Computer Science Dept.

The FLOSSWALD Information System on Free and Open Source Software

Meike Reichle & Alexandre Hanft

University of Hildesheim
Institute of Computer Science
Intelligent Information Systems Lab
D-31141, Hildesheim, Germany
meike.reichle|alexandre.hanft@uni-hildesheim.de

Abstract

We propose the implementation of an intelligent information system on free and open source software. This system will consist of a case-based reasoning (CBR) system and several machine learning modules to maintain the knowledge base and train the CBR system thus enhancing its performance. Our knowledge base will include data on free and open source software provided by the Debian project, the FLOSSmole project, and other public free and open source software directories. We plan to enrich these data by learning additional information such as concepts and different similarities. With this knowledge base, we hope to be able to create an information system that will be capable of answering queries based on precise as well as vague criteria and give intelligent recommendations on software based on the preferences of the user.

1 Introduction

In the beginning of free and open source software, these programs were mainly written for their developers' own needs or those of their communities. This has produced a large and diverse range of software wich often offers numerous alternatives for the same task, such as text editors, e-mail clients or web browsers. Also because of this existing project descriptions are mostly technically phrased and focus on the project's technological features.

However, in order to chose from a range of available software especially less experienced computer users mainly ask for qualitative attributes such as usability, stability and an agreeable look. Already existing software directories also offer mainly technically oriented search possibilities. What's missing here is the link between the user's qualitative expectations and the technical attributes of a project.

We believe it is possible to learn to translate these qualitative attributes into a set of technical features. Our plan is to design and implement an information system on free and open source software, FLOSSWALd, that offers searches by technical as well as qualitative attributes. The system's knowledge base will consist of software descriptions, improved with tags and user feedback on the results.

First, we illustrate our motivation to launch FLOSSWALD. Section 2 introduces the idea behind FLOSSWALD, including a first analysis of the available data sources. Section 3 presents related work that we have examined in the course of the project's creation. This paper closes with a conclusion and an outlook in section 4.

1.1 Motivation

Since its first public emergence, free and open source software has steadily gained more popularity. What first was an elaborated hobby among computer scientists, has today found its way to a much larger community, including less experienced computer users. With this increasing amount of users and of course also developers there has also been an increasing demand for information on that matter.

The Free and Open Source Software Community however is a complex social and technical network that consists of hundreds of thousands of individual groups and projects. Since the large majority of these projects is non-commercial, they usually don't engage in advertisement or public relations but focus on technical development and community contacts.

This has created a wide supply of free and open source software in all degrees of quality, from low class to very high grade, but most of it only known to its users, other insiders or those who know where and how to search for it. But the popularity of free and open source software is rising, and a growing number of computer users who have just recently begun using some of the most popular free and open source software such as the Firefox browser or the Thunderbird e-mail client are now considering to exchange also other programs for a free and open source alternative. These computer users usually only know very few isolated projects and don't know about the actual free and open software scene.

Free and open source software is not limited to private users though. With the increasing cost of commercial suites and the ongoing need to reduce costs, free and open source business and server applications such as the Apache HTTP server or the mysql database management system have already gained a substantial market share. The Netcraft web server survey for July 2006 [Netcraft 2006] shows a 63.09% share for the Apache web server, followed by Microsoft's IIS with 29.48%.

This rising interest in free and open source software also creates a new need for information on free and open source

software projects and their nature and quality. However, knowledge about this topic is still very much restricted to insiders who are themselves active in the Free and Open Source Software Community. While simple technical questions such as *"What database does application XY use?"* are sufficiently easy to answer using e.g. web search machines, more vague questions such as *"What is the right e-mail client for me"* or *"Which GNU/Linux distribution should I put on my small company's web-server?"* are a much harder task. Such questions don't only need technical information but also meta information such as how old or established a project is, how many users or developers it has, how mature its code base is, or how reliable it is to still be around in a few years.

Existing information services such as Freshmeat or Sourceforge rely heavily on technical criteria and language and are thus of only small use to inexperienced users. As a consequence, we plan to implement an intelligent information system that meets this need by offering more intuitive search criteria and intelligent search tools based on user preferences and learned similarities between software or user groups.

1.2 A note

In this paper we use the term "free/libre and open source software" in order to include both, the Open Source and the Free Software community. The term *free* in this text is thus not understood as *"for no cost"* but in the sense of freedom, meaning

> *[...] the users' freedom to run, copy, distribute, study, change and improve the software.* [FSF 2004]

2 The FLOSSWALD Project

First we introduce the project and describe the investigated data sources for our knowledge base: Debian packages, DebTags, Debian changelogs, the Debian bug tracking system and the FLOSSmole project followed by a disscussion of their adequacy. The last part of this section presents the planned implementation as an instatiation of a more general framework for knowledge-based systems.

2.1 Concept

FLOSSWALD, the **F**ree/**L**ibre **O**pen **S**ource **S**oft**W**are and **A**pp**L**ication **D**irectory, is a project proposal that aims to use a case-based reasoning system that includes information about the individual softwares in its knowledge base. We decided to us a case-based system, because we are dealing with vague criteria and use a large collection of individual information entities. The system is further equipped with several machine learning components that are meant to improve system performance by creating additional knowledge in the form of concepts, e.g. user groups or similarities (such as *"of a similar kind"* or *"do the same task"*), from the provided data.

2.2 The Knowledge Base

The knowledge base will be developed in three stages, each integrating a new data source. Our first data source will be the Debian GNU/Linux package repository. This repository offers several sources of information on software, which will be elaborated in the following sections. Secondly, we will extend our knowledge base to also include data from other free and open source software directories, such as Freshmeat, Sourceforge or Savannah. For these data it would be possible to cooperate with the FLOSSMOLE project [Howison et al. 2006] that provides raw data, mined at Freshmeat, Sourceforge, ObjectWeb and Rubyforge and also from donated data by other research teams. These data can be used to enrich the information already gathered in stage one. In the last stage, we hope that the FLOSSWALD project will have gathered enough momentum to also attract software authors, maintainers and users themselves and offer them a way to complete and update our data on their projects or enter new projects as they arise.

The Debian Project

Debian GNU/Linux is a free operating system, that is developed by more than a thousand volunteer developers and many more contributors such as package maintainers, translators and documentation writers all over the world, who collaborate mainly via the Internet. Debian is distinguished from other GNU/Linux distributions primarily by its overriding commitment to the principles of Free Software as laid down in its "Free Software Guidelines" [Debian 1997], its non-profit nature, and its open development model. Debian GNU/Linux is a binary Linux distribution, which means that it takes existing free software, "packages" it and provides those packages to be installed on the user's system. So instead of installing or compiling a piece of software a user downloads and installs the according package.

Debian GNU/Linux has, due to its age and popularity, probably the largest selection of prepackaged free and open source software of all GNU/Linux distributions. The Debian package repository [1] currently (as of July 2006) holds 15,660 binary packages and 9,053 source packages in its stable release. This number is still growing, for the upcoming release the package repository currently holds 17,583 binary packages and 10,228 source packages. And, what's most important, all of these packages come with a textual description, such as the example in Fig. 1.

These descriptions already offer a great wealth of information. The textual description can be analyzed with different information retrieval tools, extracting important terms or finding similar descriptions in other packages. The package's size, dependencies, section and priority can also provide conclusions on its suitability for an existing system, its nature or purpose. Additionally to this, the Debian package repository offers several other sources of information.

DebTags [Zini 2005] is a project started by Enrico Zini. Those tags are shown alongside package descriptions where available (Fig 1, last two lines) and give meta information on the software. The tags include numerous different ontologies representing different "perspectives" such as what the software is used for, what interfaces are used, what role the software has (server, client), its programming language, used protocols and many others. DebTags are in a machine readable format and thus allow for smarter search and navigation interfaces than the original full text Debian package search.

Debian further collects detailed anonymised usage data on its packages. Debian GNU/Linux users can voluntarily

[1] http://packages.debian.org

```
Package: 3dchess
Priority: optional
Section: games
Installed-Size: 136
Maintainer: Debian QA Group <packages@qa.debian.org>
Architecture: i386
Version: 0.8.1-12
Depends: libc6 (>= 2.3.6-6), libx11-6, libxext6, libxmu6,
libxpm4, libxt6, xaw3dg (>= 1.5+E-1)
Filename: pool/main/3/3dchess/3dchess_0.8.1-12_i386.deb
Size: 33564
MD5sum: fecee217870b621286f75e528496d3b1
SHA1: 88343e19f566cf5cd11ef099bad97fbabf4e316d
SHA256: 3601709708044f7e489a0a74dbe4aca0e04b2fe1bc
533655b268af36fb6abd2c
Description: 3D chess for X11
3 dimensional Chess game for X11R6. There are three boards,
stacked vertically; 96 pieces of which most are the
traditional chess pieces with just a couple of additions;
26 possible directions in which to move. The AI isn't
wonderful, but provides a challenging enough game to all but
the most highly skilled players
Tag: game::board:chess, interface::3d, use::gameplaying,
x11::application
```

Figure 1: The package description of a Debian package

install a program called *popularity contest*[2] (popcon) that sens anonymised reports to the Debian project indicating what packages are installed on a user's system and when they have been used the last time. These data allow a first take on e.g. the popularity of a particular software or – if analysed on a per user basis – what softwares are often used together.

Debian Changelogs and the bug tracking system [3]: Every Debian package has a changelog where all changes or updates on the package are noted, the Debian bug tracking system is used by developers, maintainers and users to report problems or bugs of a piece of software and monitor these reports and their solutions e.g. by patches or new versions. Both these sources can give information on the up-to-dateness and stability of a package.

The FLOSSmole Project
The FLOSSmole (formerly OSSmole) project is a collaborative project

> *[...] designed to gather, share and store comparable data on and analyses of free and open source software development for academic research.* [Howison et al. 2006, S.1]

Its aim is to create a trusted dataset for research communities dealing with free and open source software, such as the TREC dataset [TREC 2005] in the information retrieval community or the UCI repository in machine learning [Newman et al. 1998]. As a collaborative project FLOSSmole expects its users to give back any improvements or additional scripts that are created when using the provided data.

In order to achieve this the FLOSSmole project identified several key requirements [Howison et al. 2006], that it aims to comply with: Firstly, the collected data is required to be easily available, without a lengthy requisition procedure or having to deal with complicated repositories, such as versioning systems. This is achieved by offering simple unmonitored web downloads. Furthermore, the provided data has to be comprehensive and compatible, offering different timestamped versions in order to allow historic comparisons and also comparisons between different free and open source projects or repositories by including mappings between the different databases.

FLOSSmole gathers its data both by web spidering and also by using the project directory's database dumps where available. These datasets are then "cleaned": Where databases are available they have to be restructured, since they all use their own structure and attributes, the individual attributes have to be mapped to their respective counterparts in other repository's databases. Spidered data have to be checked and cleaned of false input, redundancies and the like. In the above mentioned repositories (Freshmeat, Sourceforge, ObjectWeb and Rubyforge) FLOSSmole mines different data, including project data (name, programming language, platform, operating systems, intended audience, project topics) and developer-related data (number of developers, their contact information and roles). Additionally FLOSSmole mines on the bug tracking systems of Sourceforge, Savannah, Freshmeat and the Apache Foundation, collecting information on bugs, such as when they were opened and closed, their priority and status over time.

All of these data are available as raw database dumps and may be used freely by scientific projects dealing with free and open source software.

Using these data we intend to map the respective projects to their developed software and thus extend the already existing cases with new attributes or create new we cases where necessary.

2.3 Adequacy of the Data Sources

We believe that the presented data sources are adequate for an information system such as the one we are planning. In order to serve as a knowledge base for a CBR system, the data provided needs to be correct with respect to content, of a sufficient supply as well as structured and in a simple format. Regarding the Debian packages, the correctness of the data may be assumed since they are taken directly from a working system and the Debian policy tells package maintainers to describe their packages in a neutral and objective way. Debian's open development model further supports their correctness. Also, all information on the Debian packages is freely available in a structured pure text file and can thus easily be worked with or stored, e.g. into a database. The same goes for the data provided by the FLOSSmole project which can be downloaded as a pure text database dump and are also collaboratively reviewed and updated.

Of course the knowledge base of our information system will be incomplete in the beginning. Based only on the Debian packages during the first stage it will miss software that e.g. only runs on Microsoft Windows or software that is not packaged for Debian for other reasons, e.g. because it has a license considered non-free by the Debian Free Software Guidelines. Also we will have to evaluate which attributes are provided by the respective sources, how they

[2] http://popcon.debian.org/
[3] http://bugs.debian.org/

can be mapped or converted, which of them are useful to us and whether we might have to create or extract additional information such as user ratings or associated user groups/categories.

2.4 Integration into a Knowledge-Based System

Due to the heterogeneity and the large amount of free and open source software, it seems to be appropriate to use case-based reasoning to give the user advice according to their vague descriptions and (soft) constraints. In such a case-based reasoning system, each software project should be represented by an individual case. Additionally, users should have the option to tag already known software with freely chosen tags to enrich the software descriptions with attributes that conform better with most users' level of abstraction than the mostly technical information we get from the afore mentioned sources. To finally create the "link" between more formal project descriptions and the vague descriptions from the users' perspectives we plan to use machine learning technology.

In order to query the system a flexible input mask is offered that allows giving information on what (type of) operating system is used or what other software is already installed. In order to not overload the user interface the according menus would be optional so that if the user choses to give a particular information e.g. *Browser* this would then open a drop down menu including a list of browsers to chose from. Further on, the user has the possibility do define precise as well as vague requirements, with the vague ones including points such as *"Runs on slow computers"*, *"Language can be switched"* or *"Easy to install"*. These more intuitive requirements would then be mapped to actual technical queries, e.g. *"Easy to install"* would be true if the project provides prebuilt binaries and an installing mechanism for the user's operating system. It should also be possible to prioritise those requirements. Additionally to these options the user can also give keywords for a free text search that will either be conducted over the full software information or only on selected fields. The user will be free to choose from these and probably also other options and combine them to create a query that best represents his or her information need.

FLOSSWALD is obviously a knowledge-based system, because most of the functionality FLOSSWALD should provide depends on knowledge and its processing. [Althoff et al. 2006] presented a framework for knowledge-based systems (KBS) that appears to be promising for realising FLOSSWALD. Their underlying idea is to once implement a highly flexible knowledge based system, that is then reconfigured for different use cases, instead of every time building a new system from scratch. To achieve this goal [Althoff et al. 2006] combine case-based reasoning, experience factory approaches, software product-lines and agent technology within one architecture for knowledge based systems.

Within this KBS architecture different components are responsible for usage and maintainance. This complies with our scenario where some users search the system for advice on a software while others update project descriptons or enter new ones. All of these tasks are associated with different roles within the system and are carried out by a software agent, possibly supervised by a human operator, depending on how easy the according task can be automated. This again corresponds to our estimation that also plans to first maintain the knowledge base and its content by hand but automate this process using machine learning technologies once the knowledge base is sufficiently large and enough correct training data has been created.

Mapping this KBS concept to our concept of the FLOSSWALD project would result in a setup as illustrated in Fig 2: FLOSSWALD, as user interface of the whole knowledge-based system, is located in the system tier and interacts with the users. It uses a CBR-Agent$_1$ for retrieval. The CBR-Agent$_1$ itself accesses the case base$_1$ inside the knowledge access tier. The maintainance which includes taking care of the case base and similarities for retrieval is done separately under the hood of a case factory represented by CFM Agent$_1$ in the maintenance and build-up tier. The case factory itself incorporates several roles like case manager or libriarian. It uses the services of a machine learning agent$_2$ for discovering the relationships between technical features and qualitative attributes. The CBR-Agent$_1$ and the machine learning agent$_2$ both process knowledge intensive tasks and are thus located in the knowledge worker tier.

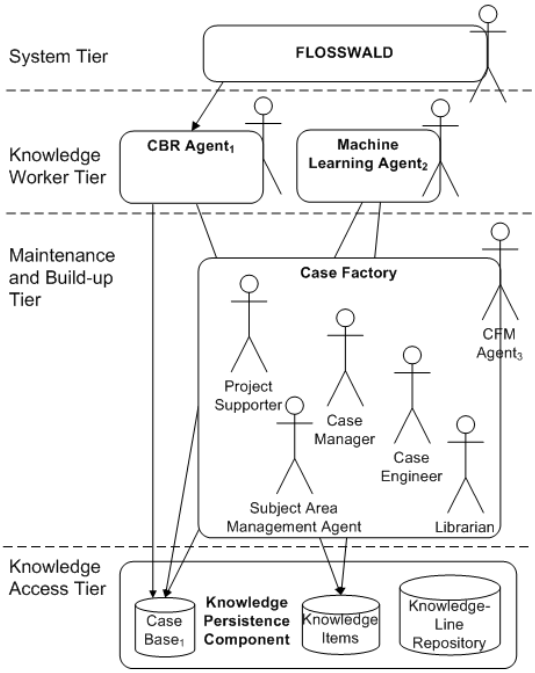


Figure 2: The FLOSSWALD concept mapped on a KBS

3 Related work

Existing information systems on free and open source software include Freshmeat, GNU Savannah,Berlios and Sourceforge. Many of these directories (e.g. Savannah, Berlios and Sourceforge) also offer development infrastructure such as web space, mailing lists, version control systems, bug tracking systems, or wikis. Those directories thus only include information on projects they also host. This has led to a certain redundancy since many projects registered with e.g. Sourceforge, so they are listed there but do not use the provided infrastructure but their own.

A system that is rather close to our concept is the FSF/UNESCO Free Software Directory which follows the same purpose. It, however, only includes software that runs on free operating systems. The FSF/UNESCO Free Software Directory is also a collaborative project, offering a web interface for users to enter and update entries. Since it did not import any data but only relies on user input the directory so far only holds around 5,000 entries. Efforts to raise this number are made, e.g. by introducing contests.

There is also a commercial information service on free and open source software, ohloh.net. It currently lists almost 4,000 open source projects. Ohloh's approach is different to that of the FSF/UNESCO Free Software Directory or ours. It mines all its data automatically and has as data source the projects' source code and version control systems. Because of this, their information is mainly focused on the projects code base. There are also a tags, that are used to display related projects. For each indexed project ohloh.net lists general information such as the age of the project, the number of developers, and its main programming language. Beside that it offers a feature called *Codebase Cost* that allows the user to calculate how much it would have cost to have that code written, based on lines of code, man years and a freely selectable yearly salary. Further on the site presents license information and several diagrams, illustrating the amount of lines of code and the activity of the projects' developers, also measured in lines of code. Because of this these numbers and diagrams only provide a measure of the effort that has been put into a particular software. Other information such as its quality or usability have to be infered from the other presented data by the user himself.

[Althoff et al. 1999] have created an information system that is designed as an experience factory and holds information on CBR technology and tools. This system is also based on a CBR system that can be queried and updated over a web interface. This system gave the original idea for the FLOSSWALD and we hope to be able to reuse some of the experiences made in the development of this system.

4 Conclusion & Outlook

We are confident that FLOSSWALD will be of great use to computer users new to free and open source software, who will most likely do vague searches, based on similarities or ratings as well as to experienced users who are searching with highly defined criteria such as required libraries or avoiding particular technologies or protocols. Our first step will be to evaluate the data provided by the Debian project and the FLOSSmole project and design a knowledge base and case structure to flexibly work with them. Then we plan to prepare and implement a CBR system based on the knowledge-based systems framework that is able to deal with the provided information and define the particular components (such as the maintenance of the knowledge base) where machine learning modules can be used to improve the system's performance.

Once the knowledge base is sufficiently large, the answer quality of the query system is satisfying and the project has hopefully gained some publicity, the last step will be to open up the project towards collaborative maintainance and implement a mechanism that allows users and software autors or maintainers to exert influence on the knowledge base itself by updating existing cases or adding new ones. To this end we will not only have to develop the technical means to actually edit the case base but also an adequate quality assurance mechanism to furtheron ensure the correctness of the data. As a basis for this we will use [Hanft and Minor 2005].

5 Sources and References

[Althoff et al. 1999] Althoff, K.-D., Nick, M., Tautz, C. (1999). CBR-PEB: An Application Implementing Reuse Concepts of the Experience Factory for the Transfer of CBR System Know-How. In Proceedings of the Seventh Workshop on Case-Based Reasoning during Expert Systems '99 (XPS-99), Wuerzburg, Germany.

[Althoff et al. 2006] Althoff, K.-D., Hanft, A. & Schaaf, M. (2006). Case Factory – Maintaining Experience to Learn. M. Göker & T. Roth-Berghofer (eds.), Proc. 8th European Conference on Case-Based Reasoning (ECCBR'06), LNCS 4106. Springer Verlag. pp 429-442.

[Debian 1997] Debian Project (1997). The Debian Free Software Guidelines (DFSG).
http://www.debian.org/social_contract#guidelines
last visited: 07/23/2006

[FSF 2004] Free Software Foundation, Inc (2004). The Free Software Definition at http://www.fsf.org/licensing/essays/free-sw.html.
last visited: 07/25/2006

[Hanft and Minor 2005] Alexandre Hanft, and Mirjam Minor. A Low-Effort, Collaborative Maintenance Model for Textual CBR. In Steffi Brninghaus (eds), ICCBR 2005 Workshop Proceedings, pages 138 – 149, August 2005, DePaul University, Chicago, USA.

[Howison et al 2006] Howison, J., Conklin, M., Crowston, K. (2006). FLOSSmole: A Collaborative Repository for FLOSS Research Data and Analyses. International Journal of Information Technology and Web Engineering. 1(3). July-September, 2006. pp 17-26.

[Netcraft 2006] July 2006 Web Server Survey at http://news.netcraft.com/archives/2006/06/28/july_2006_web_server_survey.html. last visited: 07/25/2006

[Newman et al 1998] Newman, D.J. & Hettich, S. & Blake, C.L. & Merz, C.J. (1998). UCI Repository of machine learning databases at http://www.ics.uci.edu/∼mlearn/MLRepository.html. Irvine, CA: University of California, Department of Information and Computer Science. last visited: 07/25/2006

[TREC 2005] The Fourteenth Text REtrieval Conference (TREC 2005) in Gaithersburg, Maryland, National Institute of Standards and Technology (NIST), at http://trec.nist.gov/pubs/trec14/t14_proceedings.html. last visited: 07/25/2006

[Zini 2005] Zine, E. (2005). A cute introduction to Debtags at http://debtags.alioth.debian.org/paper-debtags.html. Last visited: 07/25/2006

KDML 2006

12. Workshop der Fachgruppe Knowledge Discovery, Data Mining und Maschinelles Lernen und des Arbeitskreises Knowledge Discovery

`http://events.iis.uni-hildesheim.de/lwa06/kdml/`

Dieser Band enthält die Beiträge zum Workshop *Knowledge Discovery, Data Mining und Maschinelles Lernen* (KDML 2006, 09.-11.10.2006 in Hildesheim). Der Workshop wird 2006 zusammen von der GI-Fachgruppe *Knowledge Discovery, Data Mining und maschinelles Lernen* (FG-KDML, früher FGML) und dem Arbeitskreis *Knowledge Discovery* (AK-KD) des GI-Fachbereichs *Datenbanken und Informationssysteme* (FB DBIS) organisiert. Der Workshop wird wieder im Rahmen der Workshop-Woche *Lernen – Wissensentdeckung – Adaptivität* (LWA 2006) gemeinsam mit den Fachgruppen *Adaptivität und Interaktion* (ABIS), *Information Retrieval* (FGIR) und *Wissensmanagement* (FGWM) veranstaltet.

Wir erhielten Einreichungen zu verschiedensten Themen der Fachgruppe und des Arbeitskreises, die sich auch so im Programm des Workshops KDML 2006 widerspiegeln:

- Clustering & Subgroup Discovery
- Web-Mining & Information Retrieval
- Data Mining Anwendungen
- Bio-KDD
- Visualisierung
- Data Stream Mining
- Klassifikation
- Frequent Subgraphs
- Inductive Rule Learning
- Kernel-Based Learning

Wir danken allen Autoren für ihre spannenden Beiträge und freuen uns auf interessante Vorträge und Diskussionen. Wir danken auch den Organisatoren der LWA 2006, Martin Schaaf, Alexandre Hanft und Klaus-Dieter Althoff.

Alexander Hinneburg, Andreas Hotho und Ralf Klinkenberg
Halle / Kassel / Dortmund, September 2006

KDML 2006

12th Workshop on Knowledge Discovery, Data Mining, and Machine Learning

http://events.iis.uni-hildesheim.de/lwa06/kdml/

These proceedings contain the contribution to the workshop *Knowledge Discovery, Data Mining, and Machine Learning* (KDML 2006, October 9th-11th, 2006 in Hildesheim, Germany). In 2006, this workshop is jointly co-organized by the German Informatics Society's (GI) special interest group on *Knowledge Discovery, Data Mining and Maschine Learning* (FG-KDML, formerly FGML) and the working group *Knowledge Discovery* (AK-KD) of the GI section *Databases and Information Systems* (FB DBIS). The workshop once again is part of the workshop week *Learning – Knowledge Discovery – Adaptivity* (LWA 2006), jointly co-organized with the GI special interest groups on *Adaptivity and Interaction* (ABIS), *Information Retrieval* (FGIR), and *Knowledge Management* (FGWM).

We received submissions from a variety of topics from the special interest group and the working group, which is also reflected in the programm of KDML 2006:

- Clustering & Subgroup Discovery
- Web-Mining & Information Retrieval
- Data Mining Applications
- Bio-KDD
- Visualization
- Data Stream Mining
- Classification
- Frequent Subgraphs
- Inductive Rule Learning
- Kernel-Based Learning

We would like to thank the authors for their interesting contributions to KDML 2006 and we are looking forward to inspiring presentations and discussions at the workshop. We would also like to thank the organisers of LWA 2006, namely Martin Schaaf, Alexandre Hanft und Klaus-Dieter Althoff.

Alexander Hinneburg, Andreas Hotho, and Ralf Klinkenberg
Halle / Kassel / Dortmund, September 2006

Case-Based Characterization and Analysis of Subgroup Patterns

Martin Atzmueller
University of Würzburg,
Department of Computer Science,
Am Hubland, 97074 Würzburg, Germany
atzmueller@informatik.uni-wuerzburg.de

Abstract

In this paper, we propose a case-based approach for characterizing and analyzing subgroup patterns: We present techniques for retrieving characteristic factors and cases, and merge these into prototypical cases for presentation to the user.

In general, cases capture knowledge and concrete experiences of specific situations. By exploiting case-based knowledge for characterizing a subgroup pattern, we can provide additional information about the subgroup extension. We can then present the subgroup pattern in an alternative condensed form that characterizes the subgroup, and enables a convenient retrieval of interesting associated (meta-)information.

1 Introduction

Subgroup discovery is a powerful and broadly applicable technique aiming at discovering interesting subgroups concerning a certain target property of interest, e.g., in the subgroup of smokers with a positive family history the risk of coronary heart disease (target property) is significantly higher than in the general population. The discovered interesting subgroups denote *nuggets* or *chunks* of knowledge. A subgroup is usually easy to interpret depending on a suitable description language, e.g., using conjunctive selection expressions. In that sense the subgroup description defining the subgroup objects (cases) stands for itself. Nevertheless, methods for subgroup characterization and analysis can be very useful, e.g., [Gamberger *et al.*, 2005; 2003], since they can be used to obtain further information about the extension of the subgroup, i.e., the cases covered by the subgroup description.

In the context of experience management [Bergmann, 2002] and case-based reasoning, cases contain specific knowledge of previously experienced, concrete problem situations [Aamodt and Plaza, 1994]. Usually, a case consists of a problem description part, a solution part, and additional attached meta-information, e.g., a description of the context of the case [Bartsch-Spörl *et al.*, 1999]. For example, in the medical domain specific cases for patients are collected which do not only include the case description (given by a set of attribute values) but also additional information, e.g., images from x-ray or sonographic examinations. Then, presenting a characteristic set of cases can be used for identifying typical problem situations and contexts of a specific subgroup. Such introspective information can support the user in interpreting the discovered subgroup patterns, by presenting a subgroup in an alternative form.

In this context, we propose case-based methods providing characterization and analysis capabilities concerning the subgroup extension, i.e., the cases covered by the subgroup. First, characteristic factors of the subgroup and their respective strengths are identified. Then, typical and extreme cases characterizing the subgroup are retrieved. The obtained set of factors, the respective cases, and associated meta-information contained in the cases, can then be provided as important additional information. For example, in the medical domain meta-information such as medical images, the name of the examiner that examined or documented a case, and a typical context of a subgroup pattern can both provide important analytical information and increase the actionability of the pattern.

In this paper we show how to characterize a subgroup in terms of its characteristic factors, how we can locate relevant characteristic cases using that information, and how we can finally summarize these by generating a *prototypical pattern case* containing the characteristic cases and factors. This case is then presented to the user as a representative case for a specific subgroup pattern.

The rest of the paper is organized as follows: We first introduce subgroup discovery, subgroup patterns, and characterization techniques in Section 2. After that, we present methods for case-based characterization and analysis of subgroup patterns in Section 3: We discuss an approach for obtaining a ranked list of the characteristic factors of a subgroup pattern. Next, we show how to analyze and exemplify subgroup patterns using typical and extreme cases. Based on these techniques, we present a method for generating *prototypical pattern cases* as a condensed representation of the factors and cases characterizing a given subgroup pattern. Next, we provide two case-studies in the medical domain in Section 4. Finally, we conclude the paper with a summary in Section 5, and point out interesting directions for future work.

2 Subgroup Discovery and Subgroup Patterns

The main application areas of subgroup discovery [Klösgen, 1996; Wrobel, 1997] are exploration and descriptive induction, to obtain an overview of the relations between a (dependent) target variable and a set of (independent) explaining variables. A subgroup pattern is specified by a subgroup description language; its quality is determined by a suitable quality function and a specific target variable (concept of interest).

In the following we first introduce the used knowledge representation, before we introduce subgroup patterns and describe a method for their statistical characterization.

2.1 General Definitions

First, let us introduce some vocabulary for the used knowledge representation: Let Ω_A be the set of all attributes. For each attribute $a \in \Omega_A$ a range $dom(a)$ of values is defined; we assume $\mathcal{V}_A$ to be the (universal) set of attribute values of the form $(a = v)$, $a \in \Omega_A, v \in dom(a)$. Other common names for attribute values are *findings* and *observations*.

A *case* c is defined as a tuple

$$c = (\mathcal{V}_c, \mathcal{I}_c),$$

where $\mathcal{V}_c \subseteq \mathcal{V}_A$ is the set of attribute values observed in the case c. The set of attribute values is also often called the set of *observations* for the given case, but can also include the solution of a case, e.g., a diagnosis in the medical domain. The set $\mathcal{I}_c$ provides additional (meta-) information.

In our context, we do not explicitly consider the solution part of a case that is usually modeled for case-based reasoning applications. It is easy to see, that the solution of a case, e.g., a diagnosis in medical domains, could easily be included in either the set $\mathcal{V}_c$ or the set $\mathcal{I}_c$, depending on the requirements of the application.

The set of all possible cases for a given problem domain is denoted by Ω_C. Let $CB \subseteq \Omega_C$ be the case base containing all available cases (also often called instances).

2.2 Subgroup Patterns

A subgroup pattern is defined by a subgroup description language. A single-relational propositional subgroup description

$$sd = \{e_1, e_2, \ldots, e_n\},$$

is defined by the conjunction of a set of selection expressions (selectors) $e_i = (a_i, V_i)$, i.e., selections on domains of attributes, $a_i \in \Omega_A, V_i \subseteq dom(a_i)$. For example, the subgroup given in the introduction is defined by the selectors *smoker=yes* and *family history=positive* (with respect to the target property *coronary heart disease*). The selection expressions contained in the subgroup description are also called the *principal factors* of the subgroup. We define Ω_E as the set of all selection expressions and Ω_{sd} as the set of all possible subgroup descriptions

The interestingness of a subgroup pattern can be flexibly formalized by a (user-defined) quality function

$$q : \Omega_{sd} \rightarrow R,$$

e.g., [Klösgen, 1996], that is used in order to evaluate a subgroup description $sd \in \Omega_{sd}$. Typical criteria for ranking subgroups and for estimating their quality include the difference in the distribution of the target variable concerning the subgroup and the general population, and the subgroup size. Usually the k best subgroups and/or the subgroups with a quality above a minimum threshold are selected.

2.3 Statistical Characterization of Subgroup Patterns

Subgroups can always be characterized by the factors used to describe them, i.e., by the selectors contained in the subgroup description. However, besides these *principal factors* there are certain *supporting factors* that can also be applied in order to characterize a subgroup, c.f., [Gamberger *et al.*, 2005]: The supporting factors are given by attribute values $supp \subseteq \mathcal{V}_A$ contained in the subgroup that are identified using basic statistical analysis. The value distributions of their corresponding (supporting) attributes differ significantly comparing the subgroup and the total population with respect to the concept of interest.

Thus, given a binary target variable, a supporting attribute a of a subgroup s is defined as an attribute with a significantly different distribution comparing the true positive (target class) cases contained in the subgroup s and all the negative (non-target) cases contained in the total population.

We say, that an attribute value $(a = v)$ corresponding to the selector $e = (a, \{v\})$ of a supporting attribute a is characteristic for the subgroup, i.e., it is a supporting factor, if it is positively associated with the true positive (target class) cases contained in the subgroup compared to all the negative cases. For testing the statistical significance of an attribute and an attribute value we apply the standard χ^2-test for independence with a 0.05 significance level (i.e., with a confidence level of 95%), and the correlation- or ϕ-coefficient for binary variables, respectively.

The principal factors can be regarded as *strong* factors, while the supporting factors can be regarded as a kind of *weak* factors: The principal factors are observed in all cases of a subgroup while the supporting factors are only observed in some cases. Nevertheless, the supporting factors can provide important additional information with respect to the target cases contained in the subgroup. As discussed by Gamberger et al. [Gamberger *et al.*, 2005] presenting the supporting factors characterizing the subgroup in addition to the principal factors can be very helpful for the user: Given the principal factors the supporting factors can provide additional evidence with respect to the target concept. In this way, observing the supporting factors can facilitate an easier recognition of target cases [Lavrac *et al.*, 2002]: If a case is assigned to a subgroup based on the principal factors, then observing a supporting factor provides for some evidence that the case is potentially positive with respect to the concept of interest. Thus, the supporting factors are used to point at specific characteristics of the target space covered by the subgroup. Then, we can define a generalized set F of *characteristic factors* as the union of the principal and supporting factors.

Considering the subpopulation defined by the subgroup the principal factors are contained in all cases. The supporting factors do not occur in all cases of the subgroup but may occur in many cases. Then, their individual strength in confirming the concept of interest, i.e., their *relative importance* can be scored. We will describe such an approach in Section 3.1 below.

3 Case-Based Subgroup Characterization and Analysis

In this section we describe the methods of the proposed approach for case-based subgroup characterization and analysis: Given a specific subgroup pattern, we first obtain a set of characteristic factors (selectors) for the subgroup. Next, we rank these factors and obtain a set of exemplifying cases for the given factors. After that, we create a *prototypical pattern case* capturing the characteristic factors of the subgroup pattern, the set of characteristic and exemplifying cases, and a set of relevant additional factors. The generated prototypical pattern case contains a set of (real) cases associated with the set of factors characterizing the subgroup and a selection of relevant additional factors contained in the set of cases, besides the characteristic factors. In that sense, the prototypical pattern case provides a representative summary of the characteristic factors and the respective retrieved cases for a specific subgroup pattern.

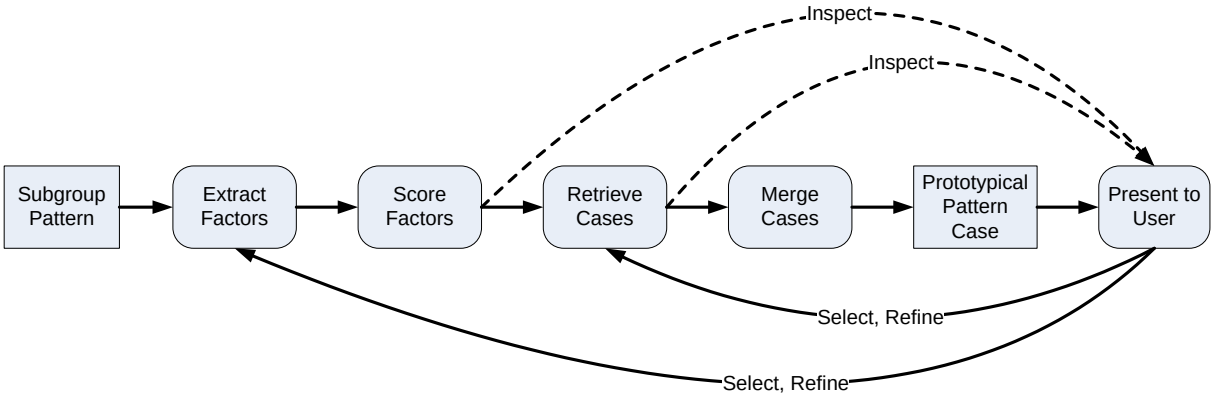


Figure 1: Process model: Case-based characterization and analysis

The approach for case-based characterization and analysis of subgroup patterns consists of the following steps shown in Figure 1:

1. Given a subgroup pattern s, we first extract the characteristic factors given by a set of selectors $F \subseteq \Omega_E$.
2. Next, we score the obtained characteristic subgroup factors F: For each selector $e \in F$ we obtain its respective (confirmation) strength with respect to the target concept. The assigned scores are then mapped to weights denoting the importance of the respective factors.
3. After that, we apply a case-based retrieval method. Concerning the cases contained in the subgroup we retrieve either typical or extreme cases with a high coverage of the characteristic factors F – as exemplifying cases for the subgroup pattern. In the retrieval method the factors can be weighted according to their relative importance, i.e., according to the assigned weights, depending on the requirements of the user.
4. Finally, we merge the retrieved cases in a virtual prototypical pattern case and present this case to the user to facilitate an easier interpretation.

This process is shown in Figure 1. It is incremental and can include user feedback: The user can optionally inspect, select and refine the set of characteristic factors that are considered in the scoring and the retrieval step. Furthermore, the user can also optionally inspect a preview of the retrieved cases before the prototypical pattern case is generated, and can refine or extend this set as well, if needed.

A prototypical pattern case contains both the set of the (scored) characteristic factors, the set of the relevant (retrieved) *subcases*, and other selected factors obtained from the set of subcases. The prototypical pattern case representation serves several purposes:

- The user usually first considers the different factors (with assigned confirmation strengths) of the prototypical pattern case. The case contains the most important factors that characterize the subgroup pattern reflected by the collection of subcases. In that sense, the prototypical pattern case can be regarded as an extended representative case: It can either contain a summary of the typical problem setting of the subgroup, or a range of the extreme settings of the subgroup pattern.
- Furthermore, the set of the typical or extreme cases of the subgroup can be inspected in detail by the user: The prototypical pattern case also contains a mapping from each contained subcase to the set of the most similar subcases in order to identify clusters representing related situations in a specific context.
- Each factor contained in the prototypical pattern case is also linked to the originating (real) cases contained in the case base. Then, the different real world situations in which the factors occurs can be inspected by the user. Furthermore, these links provide the opportunity to locate other relevant meta-information.

In the following sections, we first show how we score and rank the characteristic factors: For each factor we measure the individual importance for confirming the target concept in the subgroup. After that, we describe the case-based techniques for characterizing and exemplifying subgroup patterns in terms of cases, utilizing methods from case-based reasoning. Finally, we describe how to generate prototypical pattern cases.

3.1 Scoring Subgroup Factors

After the set of characteristic factors has been determined, it can already be used for characterizing a subgroup pattern. However, by analyzing these factors further, we can additionally estimate the strength of each supporting factor with respect to the target concept.

In the following we describe a technique for computing confirmation strengths (weights) for the set of characteristic factors F of a given subgroup. To facilitate an easier interpretation by the user, we focus on a restricted set of symbolic categories. We essentially measure the individual strength of a factor $e \in F$ with respect to the evidence it provides for the target concept in the subgroup. It is easy to see that the principal factors will always obtain the strongest confirmation category, while often weaker categories will be assigned to the supporting factors.

For rating the subgroup factors concerning their confirmation strengths, we compare two populations: The true positives contained in the subgroup and the false positives of the total population. In this way we identify how significantly a selector can discriminate between the cases containing the target concept in the subgroup, and all remaining non-target class cases. For example, in the medical domain we would like to identify factors that are characteristic for a subpopulation of all the patients with a certain disease compared to all the healthy patients.

For scoring the characteristic subgroup factors we rely on an adaptation of a method presented in [Atzmueller *et al.*, 2006b]: Given a subgroup, a characteristic factor, and the target concept, we construct a 2×2 contingency table similar to the technique for identifying the supporting factors. We then compare the distribution of the factor of the true positives in the subgroup, i.e., the target class cases, to all negative cases. By definition, this association is always significant concerning the characteristic factors. Next, we compute a score $s \in [0; 1]$ according to the strength of the association using the ϕ-coefficient for binary variables (c.f., [Atzmueller *et al.*, 2006b]), utilizing the generated contingency table.

Next, there are two options for utilizing the score: First, we can map the obtained score to a symbolic confirmation category $sc \in \{+, ++, +++\}$ that specifies confirming symbolic categories in ascending order using a suitable conversion table. The symbolic category sc expresses the strength or the relative importance of a given selector e. For each factor (selector) $e \in F$ we construct a scoring selector $e' = (e, sc)$ assigning the respective confirmation category sc. Then, we can present the scored selectors to the user for an intuitive overview of the important factors and their corresponding strength for confirming the target concept of the subgroup. Second, we can utilize the obtained scores for the case-based retrieval method described below: Since the confirmation categories denote the strength of the association between an individual factor and the target concept of the subgroup, we can directly map the individual categories to weights denoting the relative importance of the factors. The weights can then be applied in the retrieval method when estimating the similarity of cases.

3.2 Identifying Exemplary Cases for Subgroup Patterns

As a first step for analyzing a specific subgroup pattern we retrieve a set of exemplary cases of the pattern: In this way, we aim to utilize the implicit experiences contained in the cases of the case base as explaining examples. Given a set of characteristic factors F of the subgroup or a user-selected subset of these, either typical or extreme cases with a high coverage of the set of factors F can be retrieved. By inspecting these sets of cases 'as is' the user is already able to obtain a view on the general 'problem setting' of the subgroup. The next step combines these cases and the factors into a prototypical case as an intuitive alternative form. In the next section we describe how the factors and the cases are merged into a prototypical pattern case as a condensed representation.

For exemplifying a subgroup pattern, a naive solution retrieves all the target class cases contained in the subgroup. However, this approach suffers from two shortcomings: First, the set of cases can be quite large for a comprehensive overview. Furthermore, a subset of F is not accounted for very precisely, i.e., the supporting factors: The target class cases contained in the subgroup are determined by the set of principal factors contained in the subgroup, and the target concept only. In contrast to only considering the subgroup description, the set of supporting factors might cover quite a diverse set of cases, since they are not contained in all of the cases. During the retrieval step, we can take the individual strengths of the factors into account utilizing the learned weights. Additionally, we can also include other background knowledge, e.g., partial similarities between attribute values, if available.

Case Retrieval We aim to retrieve a set of (target-class) cases contained in the subgroup that have a high coverage with the set $F \subseteq \Omega_E$ containing the characteristic factors (selectors). Then, we have two options to characterize the set F: First we can retrieve *typical* cases that are most similar to F while the individual cases can also be very similar to each other. These cases can then be used to exemplify the most common factors contained in F. Second, we can retrieve *extreme* cases, i.e., cases that are very similar to F but not to each other. This set of diverse cases is discriminative and can be used in order to obtain a comprehensive view on the setting of extreme factor combinations concerning the set F.

For the retrieval step we use retrieval techniques adapted from case-based reasoning methods [Aamodt and Plaza, 1994]. Given a query case q, we aim to retrieve the k most similar cases $\{c_1, \ldots, c_k\}, c_i \in CB$. The attribute values contained in the query case are commonly called the *problem description*. We consider a *virtual* query case q and define its problem description as the set of characteristic factors F_i obtained from a given subgroup s_i. Optionally, the user can modify and tune F_i interactively to fit the analysis requirements. For example, a subset F' of the factors F_i can be selected, e.g., the most interesting factors. Furthermore, the analysis can also be extended to the non-target class cases contained in the subgroup. Thus, specific queries can be easily formulated by the user.

For assessing the similarity of a (generated) query case q and a retrieved case c, we can use the well-known *matching features* similarity function $sim(q, c)$ given in Equation 1:

$$sim(q, c) = \frac{|\{e \in F' : \pi_e(q) = \pi_e(c)\}|}{|F'|}, \qquad (1)$$

for which we consider the factors $F' \subseteq F_i$ contained in the query case q; $\pi_e(c)$ returns the value corresponding to selector $e = \big(a, \{v\}\big)$, i.e., v for a (virtual) query case c, and the value of the corresponding attribute a otherwise.

Additionally we can apply a weighted similarity measure given in Equation 2 by taking the learned weights of the factors into account, if these factors should not be equally weighted. Additionally, we can apply partial similarities between attribute values, if these are available:

$$sim(c, c') = \frac{\sum\limits_{e \in F'} w(e) \cdot sim\big(\pi_e(c), \pi_e(c')\big)}{\sum\limits_{e \in F'} w(e)}, \qquad (2)$$

where $w(e)$ is the weight of the factor e. If we do not consider partial similarities between attribute values and the weights of factors, then it is easy to see that the formula simplifies to the standard similarity measure given in Equation 1. If partial similarities are not available, then we can define a default similarity of 1 if the factors are equal, and 0 otherwise.

The diversity of a set of retrieved cases $\mathcal{RC} = \{c_i\}_k$ of size k is computed according to the measure $diversity(\mathcal{RC})$, defined as follows:

$$diversity(\mathcal{RC}) = \frac{\sum\limits_{i=1}^{k-1} \sum\limits_{j=i+1}^{k} \big(1 - sim(c_i, c_j)\big)}{k \cdot \frac{(k-1)}{2}}, \qquad (3)$$

where the similarity of two cases is estimated with respect to the attributes in the constructed query case q, as described above.

To retrieve the set of the most extreme cases with respect to a subgroup pattern we apply techniques that obtain a set of most similar but diverse cases regarding to the query case. There are several methods to retrieve a set of diverse cases as described, e.g., in [McSherry, 2002]. We apply the *Bounded Greedy (BG)* algorithm introduced by Smyth and Mc Clave [Smyth and McClave, 2001]: BG starts with a retrieval set initially containing the most similar case to the query case. In each iteration of the algorithm the case in the set of the $2k$ most similar cases is selected which maximizes both the product of its similarity to the query case and its relative diversity with respect to the cases that have been selected for the retrieval set so far.

The relative diversity $relDiversity(c, RC)$ of a case c with respect to the retrieval set $\mathcal{R}C = \{c_i\}_m$ of size m is defined as

$$relDiversity(c, RC) = \frac{\sum_{i=1}^{m} 1 - sim(c, c_i)}{m} . \quad (4)$$

BG stops if the retrieval set reaches its pre-specified size k. To obtain a smaller number of diverse (extreme) cases, we can optionally select the smallest subset $R' \subseteq R$, for which the coverage between the problem description of a query case q and the union of the problem descriptions contained in R' is maximized.

The retrieved set of typical (or extreme) cases can be seen as a set of explaining examples for the given set of factors characterizing a specific subgroup. Thus, a subgroup can be inspected in a different view by considering specific exemplary cases. By presenting typical or extreme cases the user gets a detailed and intuitive impression about the objects (cases) contained in the subgroup. In the next section, we describe how to merge the retrieved cases into a prototypical pattern case for a convenient presentation to the user.

3.3 Generating Prototypical Pattern Cases

In order to create a representative of the retrieved typical or extreme cases of a subgroup pattern, we construct a *prototypical pattern case*. The prototypical pattern case is created by merging the set of the retrieved subcases or a user-selected subset of these. Basically, we then need to combine the contained attribute values and meta-information of the individual subcases.

A *prototypical pattern case*

$$cp = (\mathcal{V}_{cp}, C_{cp}, \sigma_{cp}, \delta_{cp})$$

consists of a set of subcases $C_{cp} \subseteq CB$ of a given case base CB, a set of attribute values $\mathcal{V}_{cp}$ generated using the subcases, a mapping function from an attribute value of the constructed prototypical to its set of (originating) subcases

$$\sigma_{cp} : \mathcal{V}_{cp} \rightarrow 2^{C_{cp}} ,$$

and a selection function

$$\delta_{cp} : C_{cp} \times \mathbb{N} \rightarrow 2^{C_{cp}} .$$

The selection function δ_{cp} retrieves a set of the most similar l subcases for a specific subcase of the prototypical pattern case cp, for which $l \in \mathbb{N}, l \leq k = |C_{cp}|$.

When combining the parts of the subcases, we can simply merge the contained meta-information $\mathcal{I}_c$ of each subcase c. The set of attribute values $\mathcal{V}_{cp} \subseteq \mathcal{V}_A$ of the prototypical pattern case is basically created by joining the attribute values of the subcases:

$$\mathcal{V}_{cp} = \bigcup_{c \in C_{cp}} \mathcal{V}_c .$$

It is easy to see that we can transfer all the attribute values included in the query case to the prototypical pattern case: These factors are given by the set of characteristic (principal and supporting) factors (or a user-selected subset of these) and should therefore always be contained in the prototypical pattern case. For the remaining attributes not included in the set of characteristic attributes we need to select a discriminative set of *additional attribute values* contained in the set of subcases. However, when combining the problem descriptions, i.e. sets of attribute values, conflicts can arise if two cases contain different values for the same attribute. Therefore, we need to apply a conflict resolution step for competing attribute values for a specific attribute, i.e., if $\mathcal{V}_{cp}$ contains more than one attribute value for an attribute.

The following algorithm implements such a conflict resolution strategy for determining the set of additional attribute values during the merge step:

1. We choose the value contained in the query case if included in one subcase.
2. Otherwise, we either draw a majority vote or we can apply background knowledge, if available:
   (a) Generally, we select the most frequently occurring value v from the set of the respective attribute values V contained in the subcases, i.e.,
   $$v = \arg\max_{v_i}(freq\{v_i \in V\}) .$$
   In the case of ties, we select the value that is associated most positively with the target concept utilizing the technique described in Section 3.1.
   (b) Alternatively, we can apply background knowledge, if available: Utilizing partial similarities between attribute values we can select the value which is most similar to the value included in the query case.
   Additionally, we can utilize *abnormality knowledge* (e.g., [Atzmueller *et al.*, 2005b]) which is quite common in some domains, e.g., in the medical domain. Abnormality knowledge specifies which attribute values represent a normal or an abnormal state of their corresponding attribute, e.g. the value *pain=none* is normal, whereas *pain=high* is abnormal for a certain attribute/symptom. If abnormalities are defined, then we select the value with the highest abnormality. This approach is motivated by the heuristic that often especially the abnormal values are interesting, e.g., in the medical domain. For example, if we consider two patients with two (different) diseases, then it seems to be reasonable that the more severe attribute value (finding) will be selected, e.g. *pain=high* from one diagnosis rather than *pain=none* from another one. This is especially helpful when considering a set of extreme cases characterizing the subgroup, since the abnormal values indicate extreme conditions.

The set of attribute values of a generated prototypical pattern case is then given by the set of principal factors and supporting factors of a given subgroup pattern, and by additional factors contained in a set of exemplifying cases.

We model the mapping function σ_{cp} of a prototypical pattern case cp by creating a link from each attribute value of the case cp to the set of the original subcases containing the value, when merging the set of attribute values.

Both the selection and the mapping function enable a 'drill-down' approach when further analyzing a set of factors or a set of cases: The user can easily inspect a related set of subcases, and can also inspect each originating case for a specific attribute value.

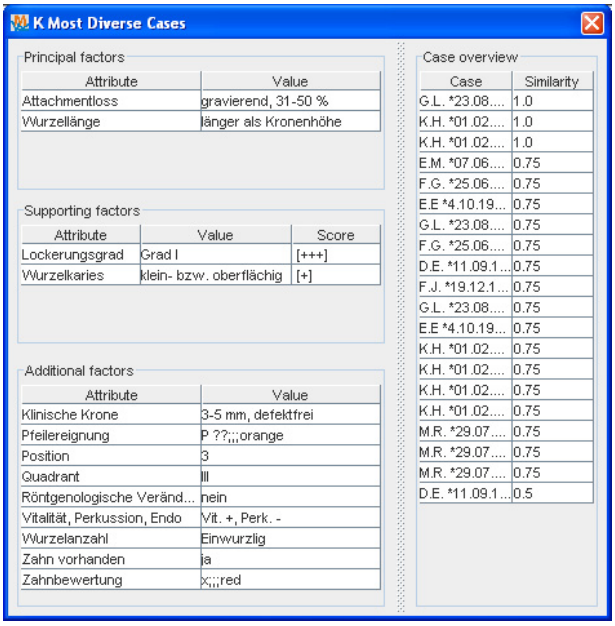


Figure 2: A Prototypical Pattern case for the subgroup *attachmentloss=strong AND root length=longer than crown length* (with respect to the target concept *incorrect tooth extraction*). The left pane contains the principal factors, the supporting factors (*toothlax=minor, root caries=minor*) and their associated scores, and the additional factors of the prototypical pattern case. The right pane shows the retrieved subcases, i.e., in this example the 20 most diverse cases for the given subgroup pattern.

Figure 2 shows an exemplary screenshot of a prototypical pattern case for the domain of dental medicine: The figure depicts the subgroup *attachmentloss=strong AND root length=longer than crown length*, and shows the principal factors, the supporting factors and their strengths, other additional factors, and the set of subcases of the generated prototypical pattern case.

3.4 Discussion

Characterizing subgroup patterns by a set of supporting factors has been proposed by Gamberger et al. [Gamberger *et al.*, 2005; 2003]. The methods for obtaining the supporting factors and for ranking these can be regarded as being related to correlation-based methods for relevance analysis of attributes and attribute values. However, in comparison to such approaches for estimating the importance or the relevance of attribute values (e.g., [Hall, 2000]) and for learning weights of attributes (e.g., [Aha, 1992]) in a case-based reasoning context, the supporting factors focus on descriptive aspects of a subgroup pattern. Thus, the importance of the attributes is estimated with respect to a pattern and a specific target concept, and not concerning the class only: The supporting factors can characterize the subgroup in a different way, orthogonal to the subgroup description.

In contrast to only obtaining the supporting factors (and thus also a subset of the characteristic factors), we further rank these in order to obtain their confirmation strength for the target concept. The obtained confirmation strengths are given by symbolic categories in order to enable an intuitive interpretation for the user. Furthermore, we can directly map these to weights (denoting their relative importance) for the similarity measure used in the case-based retrieval method.

Using prototypical cases has been introduced early in the field of case-based reasoning, e.g., [Bareiss, 1989], and is often applied in medical domains [Schmidt and Gierl, 2001]. In contrast to the existing approaches, we do not just aim at summarizing or describing a set of cases. Instead, we focus on characterizing subgroup patterns: First, we obtain characteristic factors using statistical analysis. Using these we retrieve sets of exemplifying cases. After that, we combine both into a prototypical pattern case, a process for which we can include background knowledge, if available. This prototypical pattern case then provides a comprehensive and condensed alternative representation of a subgroup pattern in the form of a single case.

The different components of a prototypical pattern case $cp = (\mathcal{V}_{cp}, C_{cp}, \sigma_{cp}, \delta_{cp})$ can then be applied in order to fulfill the requirements sketched above in Section 3:

- The set $\mathcal{V}_{cp}$ provides for a condensed form of the characteristic factors and a summary of the remaining factors contained in the subcases. In this way, the generated case can be seen as an alternative view of a subgroup pattern.
- The links between the factors contained in the problem description $\mathcal{V}_{cp}$ of the prototypical pattern case and its set of subcases C_{cp} provide an easy approach for inspecting the important factors in their specific context, i.e., embedded in their originating cases. Furthermore, associated meta-information can be conveniently identified.
- The set C_{cp} and the selection function δ_{cp} facilitate an easy inspection and traversal of the neighborhood of exemplifying cases with respect to the given subgroup pattern. Then, also relevant meta-information can be located quite easily.

So, a prototypical pattern case provides for a concise, easy to interpret, and transparent representation for analyzing, summarizing and characterizing a specific subgroup pattern. Starting with the problem description of a prototypical pattern case, the user can always apply 'drill-down' techniques in order to obtain additional information.

4 Application – Case Studies

The presented approach has already been successfully applied in medical domains. In the following we sketch two case studies: The first case study is given by an application in the domain of sonography utilizing cases from the SONOCONSULT system: Subgroup discovery is applied as a technique for knowledge discovery and for quality control. Then, the discovered subgroup patterns could be conveniently analyzed using the case-based techniques.

The second case study was performed with respect to a knowledge refinement setting applying subgroup discovery techniques in the domain of dental medicine. The goal was to improve a given knowledge-base by analyzing subgroup patterns denoting patterns with a high share of erroneous diagnoses. Then, the knowledge base could be extended by modifying and adding new relations (rules) as needed.

4.1 Characterizing Subgroup Patterns in the Context of Knowledge Discovery

For the first case study, we applied cases acquired using the SONOCONSULT [Huettig *et al.*, 2004] system. SONOCONSULT is a medical documentation and consultation system for sonography which has been developed with the knowledge system D3 [Puppe, 1998].

SONOCONSULT is in routine use in the DRK-hospital in Berlin/Köpenick and in the Würzburg University Hospital. The documented cases contain detailed descriptions of findings of the examination(s), together with the inferred diagnoses, and additional meta-information. The derived diagnoses of a case are usually correct as shown in a medical evaluation, c.f. [Huettig *et al.*, 2004], resulting in a high-quality case base with detailed case descriptions. Currently, the collected SONOCONSULT case base consists of about 11,000 cases. Due to the structured data gathering strategy and the high quality of the case descriptions the system and the collected case base provide excellent opportunities for data analysis and knowledge discovery.

We already utilized parts of the collected case base of SONOCONSULT for knowledge discovery and for data analysis using subgroup mining methods, e.g., [Atzmueller *et al.*, 2005b; 2005c]. The methods were applied in order to discover interesting clinical relations between different organ systems since the inter-organ relations are usually known in the domain of sonography. Furthermore, we applied subgorup mining for quality control with respect to the documentation habits of the sonographic examiners. Then, novel relations between different organ systems could be discovered and documentation profiles for certain examiners could be obtained. Both the relations and the profiles are represented by interesting subgroup patterns.

However, after performing knowledge discovery, the demand for a deeper inspection and characterization of the discovered subgroup patterns in terms of real cases and the further need for identifying related meta-information contained in the cases motivated the development of the presented techniques. Concerning these, the proposed methods for characterization and analysis of subgroup patterns provide powerful opportunities: The medical experts could directly locate interesting contexts, i.e., exemplary cases of specific patients, and typical case descriptions for a specific subgroup pattern. The generated prototypical cases were applied in order to obtain a summary of the typical problem setting of a subgroup pattern, and for subsequently identifying relevant meta-information contained in the characteristic set of cases. Concerning the case-studies the users could easily discover relevant meta-information, e.g., certain examiners and images associated with a given subgroup pattern using the prototypical case; the location of specific images of sonographic situations proved especially interesting for the medical clinicians.

4.2 Analyzing Subgroup Patterns in the Context of Interactive Knowledge Refinement

The second case study concerns the domain of dental medicine were we used subgroup mining for interactive knowledge refinement of a knowledge-based system. The case study was performed in the domain of dental medicine implemented with a consultation and documentation system for dental findings regarding any kind of prosthetic appliance. The system has been developed in cooperation with the department of prosthodontics at the Würzburg University Hospital.

The system aims to decide about a diagnostic plan using the clinical findings: The cases always contain the standard anamnestic findings and additional findings from x-ray examinations, e.g., abnormal x-ray findings (apical, periradicular), grade of tooth lax, endodontic state (root filling, pulp vitality), root quantity, root length, crown length, level of attachment loss, root caries, tooth angulation and elongation/extrusion. For decision support the system derives two distinct diagnosis *EX* and *IN* that either indicate the teeth that could be conserved (IN) or should be extracted (EX).

We successfully applied a method for knowledge-refinement using subgroup mining methods, in order to improve the correctness of the knowledge base that initially was in an earlier state. Therefore, we were able to improve the knowledge base significantly by adding and modifying relations that were identified using a subgroup mining approach, c.f., [Atzmueller *et al.*, 2005a; 2006a]. Subgroup mining was applied for pointing at certain subgroups corresponding to 'hot spots' of the knowledge base, i.e., specific factor combinations for which the error rate of the system increased significantly. These subgroups were then analyzed by the domain specialists in order to perform refinement operators on the knowledge base, e.g., modifying relations or adding new ones.

However, the experiences obtained throughout the earlier parts of the case study motivated the development of further methods for subgroup characterization, introspection and analysis: Often small 'hot spots', i.e., very specific subgroup patterns, needed to be analyzed in detail, either statistically or by viewing the detailed cases.

The presentation of characterizing subgroup factors and a set of exemplifying cases merged to prototypical pattern cases was a key feature for the domain specialist, who performed the analysis. Figure 2 in Section 3.3 shows an example of such a case. The method allowed for a comprehensive overview on the sub-population defined by a small set of exemplary cases. Especially interesting were the summarization and presentation of the characteristic factors by a prototypical pattern case, and the 'drill-down' options into exemplifying cases. Especially the drill-down techniques from factors to sets of cases and for navigating the neighborhood of the a set of retrieved cases proved very helpful during the application. This provided for an easier analysis of the important factor combinations, their contributions and the specific contexts they occurred in.

5 Conclusion

In this paper we have introduced case-based methods for subgroup analysis and characterization. Combining these, we presented an approach that first characterizes a subgroup in terms of its characteristic factors, ranks these, retrieves corresponding typical or extreme cases and finally combines both into a prototypical pattern case. Using this representation, the user can get a comprehensive overview of the problem setting of the subgroup pattern. Furthermore, using 'drill-down' operations on the set of cases, further interesting meta-information contained in the characteristic (real) cases can be identified. We can apply several types of background knowledge during the merge step, depending on the requirements of the user.

In the future, we plan to investigate further techniques for subgroup characterization and summarization, e.g., based on clustering techniques, and also regarding other condensed forms of sets of subgroups.

Acknowledgements

We want to thank Achim Hemsing and Prof. Ernst-Jürgen Richter from the department of prosthodontics at the Würzburg University Hospital, Prof. Hans-Peter Buscher from the department of internal medicine at the DRK-Klinik Berlin/Köpenick, and Hardi Lührs from the department of sonography at the Würzburg University Hospital for their medical expertise and analysis while performing the case studies of this research project.

References

[Aamodt and Plaza, 1994] Agnar Aamodt and Enric Plaza. Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches. *AI Communications*, 7(1):39 – 59, 1994.

[Aha, 1992] David W. Aha. Tolerating Noisy, Irrelevant and Novel Attributes in Instance-Based Learning Algorithms. *Intl. Journal of Man-Machine Studies*, 36(2):267–287, 1992.

[Atzmueller *et al.*, 2005a] Martin Atzmueller, Joachim Baumeister, Achim Hemsing, Ernst-Jürgen Richter, and Frank Puppe. Subgroup Mining for Interactive Knowledge Refinement. In *Proc. 10th Conference on Artificial Intelligence in Medicine (AIME 05)*, LNAI 3581, pages 453–462, Berlin, 2005. Springer.

[Atzmueller *et al.*, 2005b] Martin Atzmueller, Frank Puppe, and Hans-Peter Buscher. Exploiting Background Knowledge for Knowledge-Intensive Subgroup Discovery. In *Proc. 19th Intl. Joint Conference on Artificial Intelligence (IJCAI-05)*, pages 647–652, Edinburgh, Scotland, 2005.

[Atzmueller *et al.*, 2005c] Martin Atzmueller, Frank Puppe, and Hans-Peter Buscher. Profiling Examiners using Intelligent Subgroup Mining. In *Proc. 10th Intl. Workshop on Intelligent Data Analysis in Medicine and Pharmacology (IDAMAP-2005)*, pages 46–51, Aberdeen, Scotland, 2005.

[Atzmueller *et al.*, 2006a] Martin Atzmueller, Joachim Baumeister, and Frank Puppe. Introspective Subgroup Analysis for Interactive Knowledge Refinement. In Geoff Sutcliffe and Randy Goebel, editors, *Proc. 19th Intl. Florida Artificial Intelligence Research Society Conference 2006 (FLAIRS-2006)*, pages 402–407. AAAI Press, 2006.

[Atzmueller *et al.*, 2006b] Martin Atzmueller, Joachim Baumeister, and Frank Puppe. Semi-Automatic Learning of Simple Diagnostic Scores utilizing Complexity Measures. *Artificial Intelligence in Medicine. Special Issue on Intelligent Data Analysis in Medicine*, 37(1):19–30, 2006.

[Bareiss, 1989] Ray Bareiss. *Exemplar-Based Knowledge Acquisition: A Unified Approach to Concept Representation, Classification, and Learning*. Academic Press Professional, Inc., San Diego, CA, USA, 1989.

[Bartsch-Spörl *et al.*, 1999] Brigitte Bartsch-Spörl, Mario Lenz, and André Hübner. Case-Based Reasoning: Survey and Future Directions. In *XPS-99: Knowledge-Based Systems - Survey and Future Directions, Proc. 5th Biannual German Conference on Knowledge-Based Systems*, pages 67–89, 1999.

[Bergmann, 2002] Ralph Bergmann. *Experience Management: Foundations, Development Methodology, and Internet-Based Applications*. Springer, Berlin, 2002.

[Gamberger *et al.*, 2003] Dragan Gamberger, Nada Lavrac, and Goran Krstacic. Active Subgroup Mining: A Case Study in Coronary Heart Disease Risk Group Detection. *Artificial Intelligence in Medicine*, 28:27–57, 2003.

[Gamberger *et al.*, 2005] Dragan Gamberger, Antonija Krstacic, Goran Krstacic, Nada Lavrac, and Michele Sebag. Data Analysis Based on Subgroup Discovery: Experiments in Brain Ischaemia Domain. In *Proc. 10th Intl. Workshop on Intelligent Data Analysis in Medicine and Pharmacology (IDAMAP-2005)*, pages 52–56, Aberdeen, Scotland, 2005.

[Hall, 2000] Mark A. Hall. Correlation-based Feature Selection for Discrete and Numeric Class Machine Learning. In *Proc. 17th Intl. Conf. on Machine Learning*, pages 359–366. Morgan Kaufmann, San Francisco, CA, 2000.

[Huettig *et al.*, 2004] Matthias Huettig, Georg Buscher, Thomas Menzel, Wolfgang Scheppach, Frank Puppe, and Hans-Peter Buscher. A Diagnostic Expert System for Structured Reports, Quality Assessment, and Training of Residents in Sonography. *Medizinische Klinik*, 99(3):117–122, 2004.

[Klösgen, 1996] Willi Klösgen. Explora: A Multipattern and Multistrategy Discovery Assistant. In Usama M. Fayyad, Gregory Piatetsky-Shapiro, Padraic Smyth, and Ramasamy Uthurusamy, editors, *Advances in Knowledge Discovery and Data Mining*, pages 249–271. AAAI Press, 1996.

[Lavrac *et al.*, 2002] Nada Lavrac, Dragan Gamberger, and Peter Flach. Subgroup Discovery for Actionable Knowledge Generation: Shortcomings of Classification Rule Learning and the Lessons Learned. In Nada Lavrac, Hiroshi Motoda, and Tom Fawcett, editors, *Proc. ICML 2002 workshop on Data Mining: Lessons Learned*, July 2002.

[McSherry, 2002] David McSherry. Diversity-Conscious Retrieval. In *Proc. 6th European Conference on Advances in Case-Based Reasoning*, pages 219–233, Berlin, 2002. Springer.

[Puppe, 1998] Frank Puppe. Knowledge Reuse among Diagnostic Problem-Solving Methods in the Shell-Kit D3. *Intl. Journal of Human-Computer Studies*, 49:627–649, 1998.

[Schmidt and Gierl, 2001] Rainer Schmidt and Lothar Gierl. Case-based Reasoning for Antibiotics Therapy Advice: An Investigation of Retrieval Algorithms and Prototypes. *Artificial Intelligence in Medicine*, 23(2):171–186, 2001.

[Smyth and McClave, 2001] Barry Smyth and Paul McClave. Similarity vs. Diversity. In *Proc. 4th Intl. Conference on Case-Based Reasoning (ICCBR 01)*, pages 347–361, Berlin, 2001. Springer.

[Wrobel, 1997] Stefan Wrobel. An Algorithm for Multi-Relational Discovery of Subgroups. In Jan Komorowski and Jan Zytkow, editors, *Proc. 1st European Symposium on Principles of Data Mining and Knowledge Discovery (PKDD-97)*, pages 78–87, Berlin, 1997. Springer.

Visuelle Exploration multivariater Daten im Rahmen eines medizinischen Anwendungsszenarios

Stefan Audersch, Guntram Flach
ZGDV e.V., Rostock
Joachim-Jungius-Str. 11, 18059 Rostock
{stefan.audersch, guntram.flach}@rostock.zgdv.de

Abstract

In diesem Beitrag wird ein Ansatz vorgestellt, der basierend auf Techniken der visuellen Daten-Exploration und semantikbasierten Fusion eine Nutzung von Analysemethoden wie Data-Mining- und Visualisierungstechniken zur Wissensgenerierung in verteilten, kooperativen Umgebungen erlaubt. Unter Einsatz von Ontologien zur semantischen Beschreibung verteilter Quellen wird es ermöglicht, die Daten und Analysemethoden aus diesen Quellen zu fusionieren.
Kern der Architektur ist die Gatewaykomponente, die es dem Analysten erlaubt, Daten und Analysemethoden in einer verteilten Umgebung zu nutzen. Im Rahmen eines medizinischen Anwendungsszenarios wurden die vorgestellten Komponenten evaluiert.

1 Einleitung

Die visuelle Exploration von Daten und Modellen und damit die Wissensgenerierung spielt in verschiedenen Anwendungsbereichen der Medizin, Medizintechnik und der Biotechnologie eine zunehmend größere Rolle. Durch die graphische Veranschaulichung von Informationen und Sachverhalten wird das Potential der visuellen menschlichen Wahrnehmung und der Erkennungsleistung etwa von bestimmten Mustern in der Darstellung gewonnener Daten signifikant verbessert. Durch eine interaktive Erkundung der visualisierten Daten oder der graphischen Interaktion mit abstrakten Modellen können komplexe Sachverhalte veranschaulicht werden und fördern somit das Verständnis für Zusammenhänge und Strukturen.

Die automatische Erfassung von Daten durch kommerzielle Geräte und wissenschaftliche Instrumente führen zu immer größeren Mengen von immer komplexeren Daten, deren manuelle Analyse die kognitiven Fähigkeiten des Analysten bei weitem überschreiten. Zur Automatisierung dieser Analysen kommen Techniken aus dem Bereich des Knowledge Discovery in Databases (KDD) zum Einsatz, bei dem Data Mining und Visualisierung zentrale Schritte darstellen.

Die visuelle Datenexploration erlaubt es dem Benutzer, einen schnellen Einblick in die Struktur der Daten zu bekommen, Schlussfolgerungen aus den Daten zu ziehen sowie direkt mit den Daten zu interagieren (Overview, Zoom and filter, details-on-demand) [AS04, Ank04, Kei02]. Die Qualität der Ergebnisse einer solchen Wissensgewinnung ist stark von dem Expertenwissen abhängig, mit dessen Hilfe die eingesetzten Verfahren gesteuert werden. Neben dem benötigten Wissen ist eventuell die Datengewinnung, Vorverarbeitung bzw. Aufbereitung von Daten nur in einer speziellen Laborumgebung oder unter Einsatz besonderer Mittel, Werkzeuge oder Analysemethoden möglich.

Besonders auf den Gebieten der Medizin und Molekularbiologie führt die thematische und räumliche Trennung der weltweiten Forschung dazu, dass eine Vielzahl von Firmen, Gruppen und Konsortien existieren, von denen jede ihre eigene Basis an Forschungsdaten besitzt. Eine Unterstützung der Forschungsarbeit können Werkzeuge und Verfahren bieten, welche die Daten der durchgeführten Experimente mit Informationen aus komplementären Datenquellen anreichern und eine Einordnung und Bewertung der eigenen Daten im Vergleich mit Daten anderer Forschungen ermöglichen. Dabei ergibt sich die Notwendigkeit einer dynamischen Informationsfusion, die eine bedarfsgetriebene, skalierbare Kopplung und Integration von Datenbanken, Datenströmen und Datenanalysemodellen verwirklicht.

Notwendig ist demnach die Entwicklung von Konzepten und technischen Grundlagen für eine integrative Plattform mit visueller Explorations-Funktionalität, durch die die Möglichkeit zur intuitiven Analyse, Austausch und Präsentation multivarianter Datenbestände und geometrischer Modelle sowie zur Generierung neuen Wissens gegeben wird. Durch die flexible, semantisch gesteuerte Kopplung von Mining-, Data Fusion- und erweiterbaren 3D-Visualisierungs-Komponenten wird gleichzeitig das situations- bzw. projektspezifische Erschließen von Wissen aus großen Datenbeständen innerhalb domänenspezifischer, verteilter Anwendungsumgebungen unterstützt und ermöglicht.

2 Anforderungen

Ausgangspunkt dieses Vorhabens ist ein Anwendungsszenario, das medizinische Messwerte im Rahmen einer Klinischen Studie[1] betrachtet. In diesem Szenario geht es um die Analyse multivariater Patientendaten (z.B. Nerven-, Leber- und Blut-Daten) unter Einsatz von Data Mining und Visualisierungstechniken in verteilten Umgebungen. Obwohl die erhobenen Daten eine semantische Einheit bilden, werden sie aufgrund unterschiedlicher technischer und fachlicher Anforderungen in Teildatenbestände zerlegt und durch verschiedene Unternehmen und Forschungseinrichtungen getrennt voneinander analysiert (Siehe Abbildung 1).

[1] In Kooperation mit der Teraklin AG (http://www.teraklin.de)

Die getrennten Datenbestände werden einzeln ausgewertet, aufbereitet und analysiert. Hierbei kommen unterschiedliche Data-Mining- und Visualisierungstechniken zum Einsatz, die auf die Beschaffenheit der unterschiedlichen Daten zugeschnitten sind. Der Zusammenhang zwischen den einzelnen Daten kann in dieser Phase der Wissensgewinnung nicht erfasst werden. Erst eine zentrale Anwendung, die Zugriff auf die einzelnen Datenbestände (Explorationsquellen) hat, leistet dies.

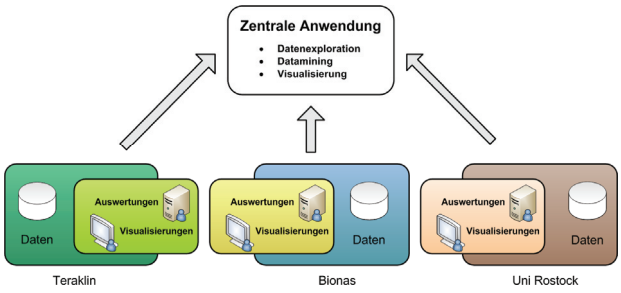


Abbildung 1: Anwendungsszenario

Eine Lösung in diesem Sinne würde es erlauben, auf der gesamten Datenbasis weiterzuarbeiten, wodurch sich beispielsweise die beiden folgenden Aufgabenstellungen lösen ließen.

Die Visualisierung von Zusammenhängen zwischen Komazustand, Blut- und Leberwerten der Patienten ist auf Grundlage der Teraklin-Datenbasis möglich. Im Rahmen der Studie ist es aber durchaus von Interesse, Messungen an Gewebeproben (Bionas[2] GmbH) oder Werte von Aminosäuren (Universität Rostock) in einen Zusammenhang mit Blut-, Leber- oder Komawerten zu bringen. Eine Kombination von Rohdaten und Ergebnissen einer Datenbasis mit Rohdaten und Ergebnissen anderer Datenbasen ist notwendig, um diese Analysen zu realisieren.

Zwischen unterschiedlichen Datenreihen der Teraklin-Datenbasis wird ein Zusammenhang vermutet, der sich anhand der vorliegenden Daten nicht eindeutig zeigen lässt. In den Auswertungen der Bionas- oder Universitäts-Daten wird festgestellt, dass ein bestimmtes Phänomen bei einem Teil der Patienten auftritt. Denkbar ist an dieser Stelle die Durchführung der ursprünglichen Analyse der Teraklin-Daten auf dem extern motivierten Teilbereich. Es muss möglich sein, selektive Anfragen an Daten einer Datenbasis unter Ausnutzung von Inhalten anderer Datenbasen zu formulieren. Das Formulieren beliebiger Anfragen an einen virtuellen Gesamtdatenbestand wäre in diesem Fall die optimale Lösung.

Zur Lösung dieser Aufgaben besteht die Notwendigkeit, die Daten sowie die Analyseprozesse aus den verschiedenen Explorationsquellen virtuell zu fusionieren. Voraussetzung für eine intelligente Zusammenführung ist eine maschinenverständliche Semantik der Explorationsquellen. Grundlage hierfür bietet eine gemeinsame Ontologie, die unter anderem eine gemeinsame Terminologie abbildet.

Die angedachten Anforderungen sollen nachstehend zusammengefasst und konkretisiert werden:

- **Datenintegration**: Es soll möglich sein, auf der gesamten Datenbasis zu arbeiten, ohne die Daten in einem initialen Schritt in eine einzige Datenbasis zu integrieren.
- **Datenexploration**: In der Datenexploration soll es möglich sein, den gesamten Datenbestand sowie auch die Ergebnisse der lokal durchgeführten Analysen, Data-Mining-Verfahren und Visualisierungen einzusehen.
- **Data Mining und Visualisierung**: Es soll möglich sein, vorhandene Data-Mining-Verfahren oder Visualisierungen auf neuen (aus der Datenfusion resultierenden) Datenbeständen durchzuführen. Im Sinne einer visuellen Datenexploration soll eine Interaktion mit bestimmten Visualisierungen möglich sein.
- **Semantik**: Die semantischen Beschreibungen sollen es erlauben, verschiedene Datenquellen einfach miteinander zu verbinden und deren Heterogenität aufzulösen. Durch Nutzung der Semantik sollte das Anwenderprogramm dem Benutzer Hilfestellung (z.B. in Form eines Wizards) bei der Exploration geben.

3 Systemarchitektur

Die entwickelten Konzepte wurden in der prototypischen Implementierung *KnowledgeDirect* [Kl04] umgesetzt. Zentraler Kern der Architektur ist das *Knowledge Explore Gateway*, bestehend aus *Control*, *Retrieval* und der *Integration Engine*.

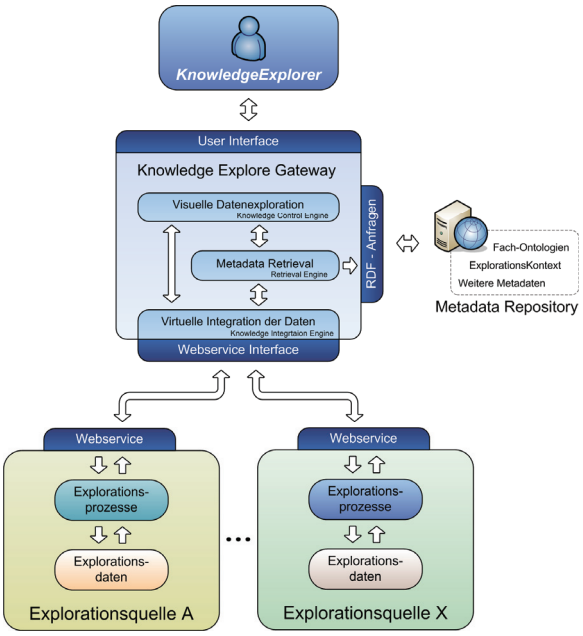


Abbildung 2: KnowledgeDirect-Architektur

Aufgabe der Integration Engine ist die Einbindung verschiedener Explorationsquellen, welche mit der semantischen Integration der in den Quellen bereitgestellten Datenstrukturen, Analyseergebnissen, Analyseprozessen, Data Mining und Visualisierungstechniken einhergeht.

[2] www.bionas.de

Grundlage für die Integration sind semantische Beschreibungen auf der Basis von RDF und OWL, die im Metadatenrepository verwaltet werden. Einen adäquaten Zugriff auf die Metadaten erhält das Knowledge Explore Gateway über die Retrieval Engine (RQL). Die Knowledge Control Engine ermöglicht die einfache Kombination der einzelnen Funktionen und Daten der Explorationsquellen auf Basis der semantischen Beschreibungen und erlaubt weiterhin die Erweiterung um komplexe Funktionalität auf dem Gebiet des Data Minings und der Visualisierung, wie z.B. 3D-Darstellungen. Der Zugriff auf verschiedene Quellen erfolgt auf der Basis von Web Services und derart transparent, dass sämtliche Aktionen explorationsübergreifend möglich sind. Eine Zuordnung von Daten zu einer bestimmten Explorationsquelle dient nur der Orientierung und birgt keine Einschränkungen in Bezug auf die Nutzung dieser Daten im Zusammenhang mit anderen Explorationsquellen.

4 Realisierungsaspekte

Die für den Lösungsansatz notwendigen Überlegungen werden im folgenden Abschnitt durch eine Auswahl verschiedener Realisierungsaspekte kurz vorgestellt.

Prozesse

Verfahren des Data Mining und der Visualisierung von Daten stellen zentrale Schritte im KDD-Prozess dar und bilden die Grundlage für die visuelle Datenexploration. Um diese Verfahren in das System zu integrieren, können diese als Prozesse aufgefasst und als Service von einer Explorationsquelle bereitgestellt werden. Ebenso lässt sich die Bereitstellung von Datentabellen als auch der Zugriff auf Analyseergebnisse als Prozess definieren. Eine Explorationsquelle (Siehe Abbildung 3) kann verschiedene Prozesse zur Verfügung stellen. Für die Integration von Prozessen ist es notwendig, die Explorationsquellen und deren Prozesse semantisch zu beschreiben. Die Beschreibungen umfassen dabei Informationen über die von der Explorationsquelle bereitgestellten Prozesse. Für den jeweiligen Prozess sind Informationen über dessen Vorbedingungen, Parameter und Ergebnisse definiert.

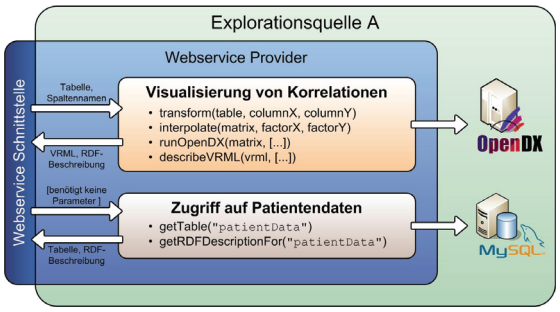


Abbildung 3: Explorationsquelle

Im Rahmen der Arbeiten wurden verschiedene Prozesse der Systeme Weka (Data Mining), OpenDX (Visualisierung) und JFreeChart (Visualisierung) entsprechend semantisch beschrieben und als Web Service zur Verfügung gestellt.

Semantische Daten- und Prozessintegration

Auf der Grundlage der semantischen Beschreibung kann die Integration der Daten und Prozesse erfolgen. Bei der Integration von Datentabellen kann hierdurch von Tabellen- und Attributnamen abstrahiert werden [AF04]. Existiert beispielsweise in einer Explorationsquelle ein Prozess P1, der die Korrelation eines Blutwertes zu einem Leberwerte (HE in T1) visualisiert, so kann dieser Prozess nun auch zur Darstellung der Korrelation zu einem anderen Leberwert (MELD in T2) aus einer anderen Explorationsquelle verwendet werden (Siehe Abbildung 4).

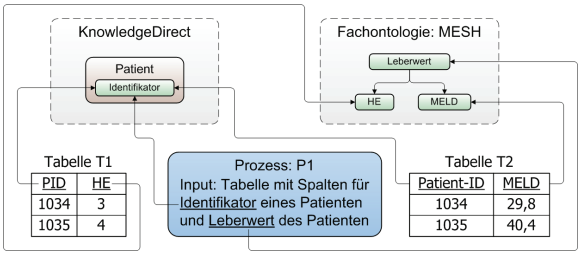


Abbildung 4: Nutzung semantischer Informationen

Auf der Grundlage der semantischen Beschreibungen ist es ebenfalls möglich, Datenbestände zusammenzuführen (Semantic Join) [LR03] und auf deren Basis neue Visualisierungen bzw. Data Mining-Verfahren anzuwenden. So kann beispielsweise mit einem geeigneten Prozess der Zusammenhang zwischen den beiden Leberwerten HE und MELD (Siehe Abbildung 4), die sich in unterschiedlichen Explorationsquellen befinden und über den Patientenidentifikator verbinden lassen, dargestellt werden.

Hilfestellungen durch semantische Informationen

Die semantischen Beschreibungen bieten durch Hilfestellungen oder semantische Kontrollen ebenfalls Potential für die Unterstützung des Benutzers bei der Exploration. Visualisierungs- und Data-Mining-Prozesse lassen sich hinsichtlich ihrer Eignung für bestimmte Datenstrukturen beschreiben. Auf Basis der im System vorhandenen Metadaten zu den verschiedenen Datentabellen können Vorschläge zur Eignung der bereitgestellten Prozesse gemachen werden [NS04]. So bieten sich beispielsweise Visualisierungen wie Shape Coding oder Parallele Koordinaten erst bei einer großeren Anzahl von Attributen an.

Durch die Nutzung von Fachontologien (u.a. MESH, Medical Subject Headings) lassen sich die in den Explorationsquellen bereitgestellten Analyseergebnisse semantisch einordnen und somit besser für weitere Recherchen nutzen. Zudem bieten die Fachontologien dem Benutzer Hilfestellung bei der Auswahl von Attributen. So kann beispielsweise für einen Prozess die Auswahl von Leberwerten (HE, MELD) über die in der Fachontologie enthaltenen Beziehungen leicht erfolgen.

5 Anwendung

Für die Evaluierung der entwickelten Konzepte und Methoden wurde das Experimentalsystem KnowledgeDirect entwickelt, das die durchgeführten Entwicklungen in einer praxisnahen Umgebung testet.

Den konkreten Datenbestand und Anwendungsfall stellt die Klinische FDA-Studie mit dem Titel *HE- Hepatic*

encephalopathy grade 3 and 4 dar. Im Rahmen dieser Studie wurden 70 Patienten über einen Zeitraum von fünf Tagen beobachtet. 35 dieser Patienten wurden dabei mit der MARS®-Therapie behandelt. Die im Rahmen der Studie erhobenen medizinischen Daten lassen sich in drei Gruppen einteilen, die durch die TERAKLIN AG (Patienten-Daten), Bionas GmbH (Gewebe-Proben) und die Universität Rostock (Aminosäuren) ausgewertet wurden.

Aufgabe des KnowledgeDirect-Frameworks war die Aufbereitung und Bereinigung dieser Daten sowie die Ableitung neuer Erkenntnisse durch die Visualisierung von Zusammenhängen zwischen den einzelnen Daten mit dem Ziel, ein besseres Verständnis über den Erfolg bzw. die Wirkungsweise der MARS®Therapie zu gewinnen.

Im Rahmen der Studie waren vor allem die medizinischen Werte HE (Hepatic Encephalopathy), GCS (Glascow Coma Scale) und Meld (Model End stage Liver Disease) von besonderem Interesse.

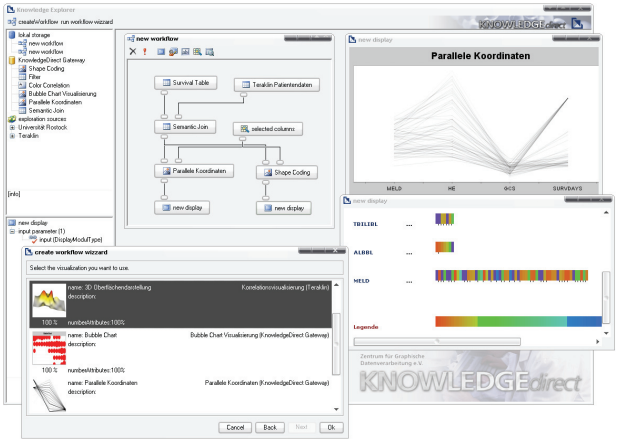


Abbildung 5: KnowledgeDirect Benutzerschnittstelle

Im Rahmen des Experimentalsystems wurden die entwickelten Konzepte und Methoden innerhalb des medizinischen Anwendungsszenarios evaluiert und durch die beteiligten Mediziner in den Bewertungsprozess der FDA-Studie mit einbezogen.

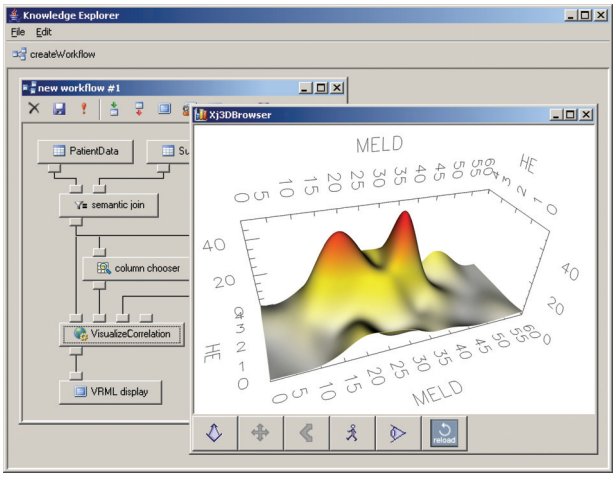


Abbildung 6: VRML-Visualisierung

Dazu gehörte auch die interaktive 3D-Visualisierung von Daten und Prozessen aus unterschiedlichen Explorationsquellen (Siehe Abbildung 6).

6 Zusammenfassung und Ausblick

Das entwickelte KnowledgeDirect-Framework dient der universellen Wissensgenerierung (Exploration) und semantikbasierten Fusion multimedialer sowie multivariater Datenbestände und ermöglicht es, die getrennt ermittelten Analyseergebnisse unter Nutzung von Ontologiewissen zusammenzuführen.

Die von den Explorationsquellen bereitgestellten parametrisierbaren Data Mining- und Visualisierungstechniken sowie die Analyseprozesse, Rohdaten und aggregierten Daten lassen sich integrieren und erlauben, globale Analysen über den gesamten Datenbestand durchzuführen. Mit semantisch gesteuerter Interaktions- und Navigationstechnik wird es auf einfache Weise ermöglicht, Daten aus verschiedenen Explorationsquellen zu selektieren, zu kombinieren, anzuzeigen und mit ihnen zu interagieren.

Gegenstand aktueller Entwicklungsarbeiten ist die Integration automatisierter Assistenz-Funktionalität durch die effiziente Nutzung semantischer Informationen während des visuellen Explorationsprozesses.

Literatur

[AF04] Audersch, S., Flach, G.: Universeller Gateway-Ansatz auf der Basis semantisch angereicherter Web Services im Rahmen heterogener eGovernment-Anwendungen, 16. Workshop über Grundlagen von Datenbanken, Monheim, 2004

[Ank04] Ankers, M.: Kooperatives Data Mining: Eine Integration von Data-Mining-Algorithmen und Visualisierungstechniken. In: Datenbank-Spektrum 4 (2004), Nr. 9, S. 6-10

[AS04] Ahlberg, C., Shneiderman, B.: Visual Information Seeking: Tight coupling of dynamic query filters with starfield displays. Proceedings of ACM CHI94, S.313-317, 1994

[Kl04] Klipps, T.: Exploration und semantikbasierte Fusion multivariater Datenbestände in domänenspezifischen Anwendungsumgebungen. Universität Rostock, Diplomarbeit, 2004.

[Kei02] Keim, D.A.: Datenvisualisierung und Data Mining. In: Datenbank-Spektrum 2 (2002), Nr. 2, S. 30-39

[LR03] Leser, U., Rieger, P.: Integration molekularbiologischer Daten. Datenbank-Spektrum 3 (2003) Nr. 6, S. 56-66.

[NS04] Nocke, T., Schumann, H.: Meta Data for Visual Data Mining. Proceedings Computer Graphics and Imaging, CGIM 2002, Kaua'i, Hawaii, USA, 2002.

User Centric Hierarchical Classification and Associated Evaluation Measures for Document Retrieval

Korinna Bade and Andreas Nürnberger
Fakultät für Informatik, Institut für Wissens- und Sprachverarbeitung,
Otto-von-Guericke-Universität Magdeburg, D-39106 Magdeburg, Germany
{kbade,nuernb}@iws.cs.uni-magdeburg.de

Abstract

The classification of documents/objects into hierarchical structures is a problem of increasing importance, e.g. considering the growing use of ontologies or keyword hierarchies in many web-based information systems. Therefore, it is not surprising that it is a field of ongoing research. Here, we propose an approach that takes hierarchy information in the classification process into account by utilizing the user behavior in the information seeking process. In contrast to other methods, the hierarchy information is used independently of the classifier rather than integrating it directly. This enables the use of arbitrary standard classification methods. Furthermore, we discuss how hierarchical classification can be evaluated appropriately, especially by considering the usefulness of the classification for a user. We present our algorithm and evaluate it on two datasets of web pages using Naïve Bayes and SVM as baseline classifiers. Significant improvements over this baseline could be found with all performance measures.

1 Introduction

Classifying objects into hierarchical structures is a task, which is needed in a growing number of applications, for example in maintaining ontologies or keyword hierarchies in web-based systems. This includes, e.g., classifying web pages into a web directory or a hierarchy that was defined by groups of users. As an example of such groups we mention social web communities such as, e.g., the open directory project [DMOZ, 2006]. Despite the possibility of creating such hierarchies manually and assigning documents by hand, automatic classification and hierarchy extension would be beneficial for many domains as the number of documents to classify has become huge.

Hierarchical structures are used to help locating information of value to a user. When searching for information in a hierarchy, the user has some idea about the topic this information belongs to. This knowledge is used to browse the hierarchy labels in a top-down manner until a subclass is found, which is expected to contain the information. Once the user reaches the most specific class describing his information need, he starts scanning the documents. If none contains the information, he might also browse more general classes, depending on how much time he is willing to spend for his search. However, it is very unlikely that he would browse other specific classes in branches of the hierarchy that deal with different topics.

What does this search behavior imply for a classification method? Each object that is classified in a wrong subclass will most likely not be retrieved by the user. In contrast to this, each object that is classified into a class that is a generalization of the correct class might still be retrieved. However, too much generalization also hinders the user's search for information as it contradicts the idea of hierarchically structuring the data. The strongest generalization would be storing everything in the root folder, which is the same as having no structure at all.

Another problem during document class assignment is that a specific class might not yet exist in the hierarchy for a certain document. Here, classification in one of the most specific classes would prevent the user from retrieving the document as he would not look so deep down the hierarchy. In this case, predicting a more general class is the only way of making retrieval possible.

In our opinion, it is important to capture this user behavior when designing and evaluating classification algorithms. Especially during evaluation this would lead to a more appropriate assessment on which algorithm better supports the user when looking for information. In the standard evaluation measures currently used, this is not considered. Therefore, we present a user centric adaptation of performance measures in Section 3. In Section 4, we then present an approach to hierarchical classification of documents. It aims on generalizing predictions that are uncertain to avoid misclassification and therefore allows more likely for retrieval. We then evaluate our algorithm in Section 5 by using our proposed user centric evaluation measures. In the following section, we start by reviewing related work.

2 Related Work

Most of the related work for hierarchical classification deals with integrating the hierarchy information into the classification process by adapting existing methods or building new classifiers. The authors of [Cai and Hofmann, 2004] try to integrate hierarchy information directly into a support vector machine (SVM) classifier by integrating a hierarchical loss function, which is motivated by a document filtering setting. Their motivation is similar to ours. Different SVM classifiers for each hierarchy node are learned by the authors of [Sun and Lim, 2001] and [Dumais and Chen, 2000] to find a suitable node in the tree. However, an inner node can only be predicted, when data is assigned to it. This is not appropriate as such inner nodes also carry important meaning defined by its child nodes.

The authors of [McCallum *et al.*, 1998] applied shrinkage to the estimates of a Bayes classifier to improve the

probability estimates. They reported large improvements. As estimating class probabilities is one step of our algorithm, this method could be applied in future work to improve the estimates. In [Cesa-Bianchi *et al.*, 2004], an incremental algorithm with performance close to SVM and also a new loss function for evaluation is presented. The authors propose to learn linear threshold classifiers for each node to decide whether a document should be classified into the node or further down the hierarchy.

In [Choi and Peng, 2004], a greedy probabilistic hierarchical classifier is used to determine the most suitable path in the hierarchy from the root to a leaf. In a second step, another classifier is used to determine the best class along this path. The authors also suggest some criteria to create a new category. In [Granitzer and Auer, 2005], the performance of two Boosting algorithms, BoosTexter and CentroidBooster, is compared to the performance of support vector machines.

The influence of different training sets (with and without using hierarchy information) is examined in [Ceci and Malerba, 2003] using Naïve Bayes and centroid learning for different numbers of extracted features. The author of [Frommholz, 2001] adapts the determined weights for each category for a certain document in a post-processing step by integrating the weights of all other categories according to their proximity (the distance in the category tree/graph) to this category. However, he makes no differences concerning the relation between two nodes.

In summary, three main approaches to hierarchical classification can be distinguished. Firstly, the original training data can be reinterpreted in a pre-processing step, which is mostly done by assigning it not only to one class but also to parent and/or child classes in the hierarchy. This approach can be critical in our context, as a class associated with a class is not merely the union of the classes of the successor classes. More specific, a document may belong to an inner class but not to any of the more specific topics associated with the successor classes.

Secondly, the hierarchy could be used directly in the classifier design, which means developing completely new classification methods. Some examples of this approach have been mentioned above.

Our approach belongs to a third category, in which the class hierarchy is exploited in a post-processing step. More specifically, this step consists of reinterpreting basic probability assignments, which typically come from a standard (non-hierarchical) classifier by integrating further knowledge. In other words, the hierarchical structure of the problem is exploited here. As an advantage of this class of methods let us mention that it allows for using well-established standard classifiers in the first step (this advantage is of course shared by the first class of methods).

3 Performance Measures for Hierarchical Classification

To compare results between different algorithms, it is necessary to define appropriate performance measures. For standard (flat) classification, evaluation is mostly done by precision and recall [Hotho *et al.*, 2005]. The precision of a class c is the fraction of all documents retrieved for this class that are correctly retrieved (see (1)), while the recall of c is the fraction of all documents belonging to this class that are actually retrieved (see (2)). The combination of the two, the F-Score, is usually used to evaluate overall performance (see (3)). Furthermore, the accuracy is often determined, which gives the percentage of correctly classified documents (see (4)).

$$prec_c = \frac{|relevant_c \cap retrieved_c|}{|retrieved_c|} \quad (1)$$

$$rec_c = \frac{|relevant_c \cap retrieved_c|}{|relevant_c|} \quad (2)$$

$$F_c = \frac{2}{1/rec_c + 1/prec_c} \quad (3)$$

$$acc = \frac{|\{d \in D | \text{predClass}(d) = \text{class}(d)\}|}{|D|} \quad (4)$$

These measures treat all classes equally. There is just one correct class and all others are wrong. However, as we already argued, in hierarchical classification, not all "wrong" classifications are equally "bad". Therefore, we propose a user oriented adaptation to the standard measures. Furthermore, collecting information about the types of misclassification can give more insight into the quality of the post-processing step.

3.1 Hierarchical Precision, Recall, and Accuracy

From our scenario at hand we derive the notion of the *retrieval path*, which starts at the correct class and goes up the hierarchy until the root class, i.e., the *retrieval path* rp_c associated with a class c contains all classes c_i from the hierarchy H which are either the class itself or a parent class thereof (denoted by $c_i \geq_H c$):

$$rp_c = \{c_i \in H | c_i \geq_H c\} \quad (5)$$

In Fig. 1, the retrieval path of class 4 is marked in bold as an example. Please note that the child classes of class 4 (here classes 6 and 7) do not belong to the retrieval path.

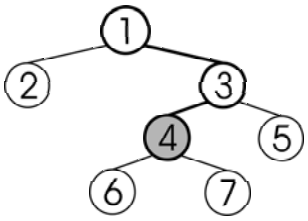


Figure 1: Example for a retrieval path

Furthermore, we denote by $\text{dist}_H(c_1, c_2)$ the number of edges on the path between c_1 and c_2 in the tree, e.g., $\text{dist}_H(4,4) = 0$ and $\text{dist}_H(1,4) = 2$.

As motivated by our retrieval scenario, classes, which belong to the retrieval path, are all beneficial to the user. However, the more the concept was generalized, the more unlikely it is that the user puts in the effort to retrieve this document. Therefore, the usefulness of a prediction along the retrieval path decreases on the way from the correct class to the root. To integrate this notion of prediction usefulness, we propose an adaptation to the standard measures, which uses gradual membership degrees of class matches.

As in the standard version, correctly classified documents are counted as a perfect match (with a value of 1) and documents that are not classified on the retrieval path are not counted at all (with a value of 0). However, documents that are classified into a more general concept along the retrieval path are now counted with a value between 0 and 1, depending on how far they are away in the class tree. These ideas can be expressed by a large number of

functions. Here, we propose the measure in (6) that shows the hierarchical similarity of a predicted class $\hat{c}$ to a class c.

$$\mathrm{sim_H}(\hat{c}, c) = \begin{cases} \exp(-\gamma \cdot \mathrm{dist_H}(\hat{c}, c)) & \text{if } \hat{c} \geq_H c \\ 0 & \text{else} \end{cases} \quad (6)$$

This function has a parameter γ that can be used to adapt the evaluation to certain types of users. More specifically, the "laziness" of the user can be modeled. The higher γ, the less would a user be willing to check documents in more general classes. Hence, it becomes more important to predict a class, which is close to the true class. In particular, it is worth considering the two extreme parameter configurations.

With $\gamma \to \infty$, every class except the correct one is assigned a value of 0. This configuration models an extremely lazy user who only explores a single class. In this configuration, the hierarchical similarity converges to the standard setting of evaluation measures. No distinction is made between incorrect predictions. Being on the retrieval path is no longer better than being aside.

The other extreme is obtained for $\gamma = 0$. Now, every class along the retrieval path has the same usefulness. Exploiting information about the hierarchical structure now becomes fairly easy. Predicting the root class by default is an optimal strategy and guarantees predicting the most useful class. Most users will most likely be found somewhere in between. For our evaluation, we therefore set γ to 0.5.

The $\mathrm{sim_H}$ function can be used to transform the crisp 0/1-assignment in the standard measures to gradual degrees. This is done in (7) - (9). Again, we want to mention that $\mathrm{sim_H}$ could be replaced by any other function computing values between 0 and 1 to express other class relationships.

$$prec_{h,c} = \frac{\sum_{d \in retrieved_c} \mathrm{sim_H}(c, \mathrm{class}(d))}{|retrieved_c|} \quad (7)$$

$$rec_{h,c} = \frac{\sum_{d \in relevant_c} \mathrm{sim_H}(\mathrm{predClass}(d), c)}{|relevant_c|} \quad (8)$$

$$acc_h = \frac{\sum_{d \in D} \mathrm{sim_H}(\mathrm{predClass}(d), \mathrm{class}(d))}{|D|} \quad (9)$$

Other researchers also proposed hierarchical performance measures, e.g. in [Sun and Lim, 2001]. However, their focus is more on evaluating the classifier performance itself, e.g. by taking category similarities or tree distances into account. Our focus is on the usefulness of the classification from a user's point of view, which is based on user behavior in the described application scenario.

3.2 Types of Misclassification

Besides the previously introduced performance measures, we are further interested in determining some statistics which allow for distinguishing different types of misclassification. These can help to get an idea about more specific behavior of the different classification methods. For example, the same accuracy values could be gained by different classifier behavior. One algorithm might have the tendency to predict for documents either the correct class or a class totally wrong, while another algorithm might prefer predicting many documents in some more general class along the retrieval path. We therefore also determined the following statistics:

(a) $\#n_c$ – number of predictions in the correct class

(b) $\overline{\#n_c}$ – predictions in a class that is a parent class of the correct class, i.e., predictions on the retrieval path but not the correct class itself

(c) $\overline{ml(n_c)}$ – average number of hierarchy levels between the correct class and the predicted class in case of (b)

(d) $\underline{\#n_c}$ – predictions of classes not on the retrieval path, i.e., not counted for $\#n_c$ and $\overline{\#n_c}$

(e) $\underline{ml(n_c)}$ – average number of hierarchy levels between the retrieval path and the predicted node in case of (d)

4 An Approach Based on Classification Uncertainty

In this section, we present an approach that uses the hierarchical class structure and further knowledge about the prediction process to avoid misclassification into classes that are not part of the retrieval path. This is done as a post-processing step, in which prediction probabilities are re-evaluated.

In our setting, we have given a class hierarchy H together with a set of training data D, whereby not every class must have training data assigned to it. Classes that have subclasses might be empty. However, the best prediction might be found in such a class. Furthermore, we have a classifier C that computes prediction probabilities $P_C(c|d)$ for each class c for a given document d. This can either be a flat (as in our experiments) or a hierarchical classifier. Keep in mind that a flat classifier would produce a probability of 0 for empty classes. In addition, the sum of the prediction probabilities for all classes does not need to sum up to 1 as the rest of the probability mass could be describing unknown classes. The goal of our approach is to find for each document a prediction that is either the correct class or another class on the retrieval path by being as specific as possible.

The basic idea of our approach is to traverse the class hierarchy top-down. At each class, the algorithm looks for the most suitable child class of the current class and decides whether it is still reasonable to descend to it or whether it is better to return the current class as the final prediction.

To be able to make this decision, we utilize two different kinds of information. The first one is based on the prediction probabilities $P_C(c|d)$ from the classifier. We integrate the hierarchical relationship between classes into these probabilities by propagating the maximum prediction probability up the hierarchy, i.e. for each class the following equation is computed:

$$P(c|d) = \max_{c' \leq_H c} P_C(c'|d) \quad (10)$$

Descending the class hierarchy by always choosing the subclass with the highest value $P(c|d)$ now produces an equal prediction to choosing the class with the highest classifier prediction $P_C(c|d)$. Please note that for this purpose for each original inner class of the hierarchy a virtual leaf class is created, which is a direct sub-class of the concerning class. The classifier predictions $P_C(c|d)$ of these classes are then associated to the virtual classes. This is needed as an internal representation but will not be visible to the user.

Second, we use the training data to derive further class information. In prior work [Bade and Nürnberger, 2005], we used class similarities. Here, we extract the probability of predicting a class correctly. We denote this as prediction accuracy of a class $P_A(c)$. By performing 10-fold cross-validation on the training data, we build a confusion matrix

M for the original classification. In a second step, we again utilize the hierarchy information to derive the prediction accuracy of each class by the following equation:

$$P_A(c) = \frac{\sum_{c_1,c_2 \leq_H c} M(c_1, c_2)}{\sum_{c_3 <_H c, c_4 \in H} M(c_3, c_4)} \quad (11)$$

In other words, we interpret each class as the combination of it with its subclasses. So for each class, the number of all documents is counted that belong to this class or a subclass and that are also predicted to belong to one of these classes. This number is set in relation to all predictions of this "group" of classes.

After having determined both probabilities for each class, we can formulate a criterion to decide whether further descend into the class hierarchy should be stopped or not. The hierarchy is traversed along the highest prediction probability. If the following criteria holds, the descend is stopped:

$$\sqrt{P(c_b|d) \cdot P(c_{sb}|d)} \cdot P_A(c_c)^t > P(c_b|d) \cdot P_A(c_b)^t \quad (12)$$

where c_c is the current class of interest and c_b and c_{sb} are the direct child classes with the highest and second highest prediction probability. t defines a threshold parameter, which can be used to tune the algorithm.

What does the above equation model? At both sides of the inequality, the class accuracy and the prediction probability of a class are combined, weighted by the parameter t. The main idea is to compare the combined value of the current class c_c (left side of inequality) with the combined value of the best child class c_b (right side of inequality). If the value decreases down the hierarchy, the algorithm stops.

However, the prediction probability in the current class is always equal to the prediction probability of the best child class due to our initialization procedure described in (10). Therefore, we decided to replace the prediction probability of the current class with the harmonic mean of the prediction probabilities of the two best child classes. The main idea is that this expresses the "purity" of the class assignment. If the two best class probabilities are almost the same, the classifier is rather undecided, which class is the best, producing an almost equal value for the prediction probability. And this is actually the kind of classification uncertainty, we want to detect and avoid by generalization. The parameter t can be used to tune how strong the influence of the class accuracy should be for this matter.

The complete algorithm of our approach to user centric hierarchical classification (UCHC) is summarized in Fig. 2.

UCHC(d, C, γ)
- For each $c_i \in H$:
  - Compute probability estimate $P_C(c_i|d)$ by C
- For each $c_i \in H$:
  - Compute $P(c_i|d) = \max_{c' \leq_H c_i} P_C(c'\,d)$
- Build M by running C on training data
- For each $c_i \in H$:
  - Compute $P_A(c_i) = \frac{\sum_{c_1,c_2 \leq_H c_i} M(c_1,c_2)}{\sum_{c_3 \leq_H c_i, c_4 \in H} M(c_3,c_4)}$
- $c_c = root(H)$
- While c_c has child classes:
  - Determine the two child classes c_b and c_{sb} of c_c that have the highest values for $P(c|d)$
  - If $\sqrt{P(c_b|d) \cdot P(c_{sb}|d)} \cdot P_A(c_c)^t > P(c_b|d) \cdot P_A(c_b)^t$
    - break
  - $c_c = c_b$
- Return c_c

Figure 2: Our User Centric Hierarchical Classification Algorithm (UCHC)

5 Evaluation

To evaluate our algorithm, we used two different, well-established flat classifiers over all classes containing training data as reference classifiers, a standard Naïve Bayes classifier and a SVM classifier (We used an implementation based on libSVM [Chang and Lin, 2001]). In a first setting, we applied the classifiers in their original form and recorded the results gained with our performance measures introduced earlier. After that, we used each classifier in combination with our user centric hierarchical approach and again recorded the results.

5.1 Data Sets

We evaluated our algorithm with two datasets. The first is the banksearch dataset [Sinka and Corne, 2002], consisting of 11000 web pages in a 2 level hierarchy (see Fig. 3). The second is a part of the Open Directory [DMOZ, 2006], consisting of 8132 web pages in a hierarchy of depth up to 4 (see Fig. 4). The number of child nodes varies from 2 to 17. We crawled the data in April 2006 and selected the categories presented in Fig. 4. Small subcategories were merged into their parent category.

- **Banking & Finance** (0 docs)
  - Commercial Banks (1000 docs)
  - Building Societies (1000 docs)
  - Insurance Agencies (1000 docs)
- **Programming Languages** (0 docs)
  - Java (1000 docs)
  - C/C++ (1000 docs)
  - Visual Basic (1000 docs)
- **Science** (0 docs)
  - Astronomy (1000 docs)
  - Biology (1000 docs)
- **Sport** (1000 docs)
  - Soccer (1000 docs)
  - Motor Sport (1000 docs)

Figure 3: The Banksearch Dataset

Our two datasets have quite different characteristics. The banksearch dataset has a rather shallow hierarchy and its classes can be separated quite well (in a classification sense). It was created by researchers for evaluation purposes only. On the other hand, the Open Directory data set is far more difficult. The data was structured by different people without any intention to provide a dataset, which is easy to classify (by machine learning methods). In other words, it is truly a "real world" dataset, which makes it clearly more interesting for evaluation from a practical point of view.

5.2 Preprocessing

After parsing the documents, we filtered out terms that occurred in less than five documents, in almost all documents ($> |D| - 5$), stop words, terms with less than four characters, and terms containing numbers. After that, we selected

- **Fitness** (124)
  - Certification (38)
  - Gyms (4)
    - Europe (88)
    - North_America (0)
      - Canada (35)
      - United_States (351)
    - Oceania (32)
  - Personal_Training (86)
  - Pilates_Method (55)
  - Services (31)
- **Society** (0)
  - Activism (131)
    - Anti-Corporation (75)
    - Internet (35)
    - In_Daily_Life (49)
    - Media (50)
      - Culture_Jamming (215)
      - Radio (149)
    - Nonviolence (55)
    - Regional (180)
    - Resources (41)
- **Society** (cont.)
  - Paranormal (144)
    - Bermuda_Triangle (32)
    - Crop_Circles (87)
    - Ghosts (47)
      - Investigators (214)
      - Personal_Pages (47)
      - Places_and_Hauntings (68)
      - Stories (39)
    - Personal_Pages (83)
    - Prophecies (81)
    - Psychic (75)
      - Animals (60)
      - Entertainers (30)
      - ESP (59)
      - Healers (28)
      - Ouija (86)
      - Personal_Pages (30)
      - Readings (444)
      - Teaching (56)
    - UFOs (282)
- **Travel** (26)
  - Guides_and_Directories (115)
  - Image_Galleries (149)
  - Lodging (13)
    - Bed_and_Breakfast (18)
    - Consolidators (65)
    - Directories (19)
    - Home_Exchanges (11)
    - Hospitality_Clubs (15)
    - Hostels (24)
      - Africa (31)
      - Asia (16)
      - Europe (95)
      - North_America (14)
      - Oceania (72)
    - Hotels_and_Motels (26)
    - Vacation_Rentals (73)
  - Preparation (37)
    - Currencies (22)
    - Health (84)
    - Passports_and_Visas (84)
  - Publications (225)
- **Travel** (cont.)
  - Specialty_Travel (246)
    - Adventure_and_Sports (215)
    - Archaeology (41)
    - Arts (126)
    - Backpacking (81)
    - Battlefields (38)
    - Boat_Charters (677)
    - Corporate (130)
    - Cruises (289)
    - Culinary (114)
    - Ecotourism (253)
    - Educational (40)
    - Family (45)
    - Pilgrimage (22)
    - Rail (41)
    - Spas (86)
    - Students (91)
    - Volunteering (137)
  - Transportation (14)
    - Air (259)
    - Car_Rentals (33)
    - Limousines_and_Shuttles (104)

Figure 4: The Open Directory Dataset; the number behind the category name indicates the number of documents directly assigned to this category.

out of these the 100 most distinctive features per class as described in [Borgelt and Nürnberger, 2004]. Each document that had an empty feature vector after this preprocessing was ignored. Each classifier was learned with the same training data. For the banksearch data, we have chosen 300 randomly selected documents from each class, and for the Open Directory data, we have chosen two third of the data in each class. The classifiers were tested with the remainder of the data.

5.3 Results

Tables 1 and 2 summarize our results using the presented evaluation measures. The results of the baseline classifiers are shown in direct comparison to our user centric approach. For each combination of dataset and baseline classifier the results are presented for the optimal setting of the parameter t. As can be seen, the influence on t of the classifier is stronger than the influence of the dataset. This is due to the different qualities of the probability estimates of the two classifiers. In future work, we want to further investigate the influence of the dataset by applying the algorithms to more datasets.

If combining results of the baseline to our user centric approach, it can be seen that our approach generally achieves significant improvements for all performance measures on both data sets. By comparing the performance on the Open Directory data in Table 2 to the results on the banksearch data in Table 1, one can find that the improvements gained by the UCHC method are much higher for the Open Directory data. This was also expected by us as this dataset is a lot more difficult. (This is also proven by the baseline performance results that are much lower for the Open Directory data. E.g., the accuracy of the SVM drops from 93% to 68%.) In specific, the statistics show, e.g., that, if we assume a search strategy as discussed throughout the paper, with UCHC(NB) 29.8% $((\underline{\#n_c}(NB) - \underline{\#n_c}UCHC(NB))/|D = (1391 - 592)/2681)$ more of the data can be retrieved for the Open Directory data, in comparison to 9.3% $((1341-633)/7626)$ for the banksearch data.

As an example of the effect of our user centric approach, the classification tree of documents from the Sport class of the Banksearch dataset is shown in Fig. 5. As can be seen, most documents that had been classified in too specific classes by the pure Naïve Bayes approach (87 documents in *Soccer* and *MS*) were correctly moved up the hierarchy by the UCHC(NB) approach and only 12 documents remained in the too specific classes. Furthermore, 2 out of 25 documents classified in nodes not on the retrieval path had been moved up to the root node. This effect occurs for all classes as can be seen by the $\underline{\#n_c}$ values in Tab. 1. Furthermore, accuracy as well as precision and f-measure could be improved.

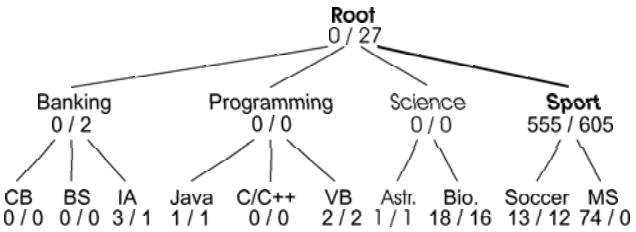


Figure 5: Classification of the Banksearch Sport Documents with Naïve Bayes / UCHC(NB).

Finally, we want to point out that our algorithm aims on generalizing predictions that are possibly wrong because the classifier is uncertain about what the correct class is. That could be, e.g., documents belonging to more than one class or documents, for which a specific class not yet exists. Setting a threshold to generalize wrongly classified documents will therefore almost always also lead to a generalization of correctly classified items, because they also might belong to more than one class. For these documents,

Table 1: Performance Results for the Banksearch Data

	acc_h	$prec_h$	rec_h	f_h	$\#n_c$	$\overline{\#n_c}\,(\overline{ml(n_c)})$	$\underline{\#n_c}\,(\underline{ml(n_c)})$
Naïve Bayes	0.8216	0.8358	0.8215	0.8286	6236	49 (1.0)	1341 (1.38)
UCHC(NB, 20.0)	0.8458	0.9003	0.8461	0.8724	5879	1114 (1.39)	633 (1.20)
SVM	0.9273	0.9278	0.9272	0.9275	7062	16 (1.0)	548 (1.40)
UCHC(SVM, 1.0)	0.9282	0.9319	0.9280	0.9300	7030	84 (1.14)	512 (1.41)

Table 2: Performance Results for the Open Directory Data

	acc_h	$prec_h$	rec_h	f_h	$\#n_c$	$\overline{\#n_c}\,(\overline{ml(n_c)})$	$\underline{\#n_c}\,(\underline{ml(n_c)})$
Naïve Bayes	0.4520	0.5436	0.3056	0.3912	1117	173 (1.26)	1391 (1.98)
UCHC(NB, 26.0)	0.5175	0.7597	0.4096	0.5323	886	1203 (1.98)	592 (1.74)
SVM	0.6847	0.6801	0.5486	0.6073	1687	86 (1.29)	759 (1.64)
UCHC(SVM, 1.5)	0.7154	0.7922	0.6086	0.6884	1518	563 (1.40)	451 (1.71)

it might be just a coincidence that the classification algorithm and the person who created the dataset did the same class assignment. Furthermore, this means (and is shown by the experiments) that our algorithms is especially useful, if the data has a more complex hierarchical structure.

6 Conclusion

In this paper, we presented a user centric approach to hierarchical classification that targets on avoiding misclassifications that would hinder the user to retrieve valuable information. Furthermore, we argued that standard evaluation measures are not suitable to evaluate this kind of setting and propose adaptations to them that take the user behavior into account. We then applied our algorithm to two benchmark datasets and evaluated the results using our proposed measures in comparison to standard classification approaches. The empirical evaluation as presented in the previous section has shown significant improvements of our method over the standard methods, especially on more complex hierarchies. For future work, we consider learning a different threshold parameter t for each class in the hierarchy. Such a class specific threshold could better adjust to class specific differences. However, the available training data for determining the best threshold value is much smaller, which could on the other hand decrease performance again.

References

[Bade and Nürnberger, 2005] Korinna Bade and Andreas Nürnberger. Supporting web search by user specific document categorization: Intelligent bookmarks. In *Proc. of LIT05*, 2005.

[Borgelt and Nürnberger, 2004] Christian Borgelt and Andreas Nürnberger. Fast fuzzy clustering of web page collections. In Mirco Nanni, Michelangelo Ceci, and Marco Gori, editors, *Proc. of PKDD Workshop on Statistical Approaches for Web Mining (SAWM)*, 2004.

[Cai and Hofmann, 2004] L. Cai and T. Hofmann. Hierarchical document categorization w. support vector machines. In *Proc. of 13th ACM Conf. on Inf. and Knowl. Management*, 2004.

[Ceci and Malerba, 2003] M. Ceci and D. Malerba. Hierarchical classification of html documents with webclassii. In *Proc. of 25th Europ. Conf. on Inform. Retrieval*, 2003.

[Cesa-Bianchi *et al.*, 2004] N. Cesa-Bianchi, C. Gentile, A. Tironi, and L.Zaniboni. Incremental algorithms for hierarchical classification. In *Neural Information Processing Systems*, 2004.

[Chang and Lin, 2001] C. Chang and C. Lin. *LIBSVM: a library for support vector machines*, 2001. Software available at http://www.csie.ntu.edu.tw/ cjlin/libsvm.

[Choi and Peng, 2004] B. Choi and X. Peng. Dynamic and hierarchical classification of web pages. *Online Information Review*, 28(2):139–147, 2004.

[DMOZ, 2006] Open directory project, www.dmoz.org, 2006.

[Dumais and Chen, 2000] S. T. Dumais and H. Chen. Hierarchical classification of web content. In *Proceedings of SIGIR-00, 23rd ACM International Conference on Research and Development in Information Retrieval*, pages 256–263, 2000.

[Frommholz, 2001] I. Frommholz. Categorizing web documents in hierarchical catalogues. In *Proc. of the European Colloquium on Information Retrieval Research*, 2001.

[Granitzer and Auer, 2005] M. Granitzer and P. Auer. Experiments with hierarchical text classification. In *Proc. of 9th IASTED International Conference on Artifical Intelligence*, 2005.

[Hotho *et al.*, 2005] Andreas Hotho, Andreas Nürnberger, and Gerhard Paaß. A brief survey of text mining. *GLDV-J. for Computational Linguistics and Language Technology*, 20(1):19–62, 2005.

[McCallum *et al.*, 1998] A. K. McCallum, R. Rosenfeld, T. M. Mitchell, and A. Y. Ng. Improving text classification by shrinkage in a hierarchy of classes. In *Proceedings of ICML-98, 15th International Conference on Machine Learning*, pages 359–367, 1998.

[Sinka and Corne, 2002] M. Sinka and D. Corne. A large benchmark dataset for web document clustering. In *Soft Computing Systems: Design, Management and Applications, Volume 87 of Frontiers in Artificial Intelligence and Applications*, pages 881–890, 2002.

[Sun and Lim, 2001] A. Sun and E. Lim. Hierarchical text classification and evaluation. In *Proc. of the 2001 IEEE International Conference on Data Mining*, pages 521–528, 2001.

Designing Semantic Kernels as Implicit Superconcept Expansions

Revised Version to appear in Proceedings of ICDM-2006

Stephan Bloehdorn*, Roberto Basili, Marco Cammisa** and Alessandro Moschitti****
*Institute AIFB, University of Karlsruhe, Germany
{bloehdorn}@aifb.uni-karlsruhe.de
**University of Rome 'Tor Vergata', Italy
{basili,cammisa,moschitti}@info.uniroma2.it

Abstract

Recently, there has been an increased interest in the exploitation of background knowledge in the context of text mining tasks, especially text classification. At the same time, kernel-based learning algorithms like Support Vector Machines have become a dominant paradigm in the text mining community. Amongst other reasons, this is also due to their capability to achieve more accurate learning results by replacing standard linear kernel (bag-of-words) with customized kernel functions which incorporate additional a-priori knowledge. In this paper we propose a new approach to the design of 'semantic smoothing kernels' by means of an implicit superconcept expansion using well-known measures of term similarity. The experimental evaluation on two different datasets indicates that our approach consistently improves performance in situations where (i) training data is scarce or (ii) the bag-of-words representation is too sparse to build stable models when using the linear kernel.

1 Introduction

Finding means for organizing, analyzing and searching the ever growing amounts of textual documents is a challenging task in knowledge management. Text classification systems [Sebastiani, 2002], which aim at automatically classifying text documents into predefined thematic classes are one approach to govern this growing complexity. Their design is mainly based on machine learning methods among which Support Vector Machines (SVMs) [Vapnik *et al.*, 1997] along with other kernel-based algorithms have become a dominant technique during the last years. The popularity of SVMs stems from two vital properties: one the one hand, being firmly grounded in statistical learning theory, they exhibit very high generalization capabilities. On the other hand, they easily incorporate prior knowledge about the target domain by means of a specific choice of the employed *kernel function*. Pioneered by [Joachims, 1998], SVMs have been heavily used for text classification, typically showing good results. The standard feature representation used in text classification settings is the so called *bag-of-words* model originating from Information Retrieval. Here, documents are encoded as vectors whose dimensions correspond to the terms in the overall corpus and the entries correspond to appropriately weighted counts of the terms in the document. Typically, the inner product (or the cosine, i.e. its normalized variant) between two vectors is used as kernel hence making the similarity of two documents dependant only on the amount of terms they share. While this approach has an appealing simplicity, it suffers from data sparseness problems in those cases where reliable distributions of terms are not available in the training documents.

To overcome the above drawback, recently, there has been an increased interest in using prior knowledge about semantic dependencies between terms of different surface form. In text-mining tasks, *semantic smoothing kernels* have emerged as one paradigm to approach this task [Siolas and d'Alché Buc, 2000; Cristianini *et al.*, 2002; Mavroeidis *et al.*, 2005; Basili *et al.*, 2005]. The knowledge encoded by such kernels is derived either from explicit background knowledge in the form of semantic networks or implicitely from statistics about the co-occurrence of terms. The rationale behind these approaches is the observation that the index terms that constitute the feature space cannot be regarded as mutually orthogonal dimensions but rather as dimensions with varying degrees of semantic similarity (with synonymous terms being the most extreme cases where distinct dimensions actually correspond to a single one). In this view, linear kernels within the *bag-of-words* paradigm appear as a rough approximation only. Despite this, literature studies indicate that the *bag-of-words* approach achieves very good results. This is typically explained by the implicit assumption that stable patterns can be detected even in a poor representation as long as sufficient training data is available. However, in those cases where training data is scarce or the representation of individual instances is hampered by extreme sparseness, an a-priori bias in form of a more adequate kernel is likely to boost the overall performance.

In this paper we investigate the use of a new type of semantic smoothing kernels for text classification. We exploit the similarity of two terms within a semantic smoothing matrix which generalizes the standard linear kernel by giving the vector components *across dimensions* a say when evaluating the kernel of two documents. To determine appropriate term similarities, we represent index terms as instances within a seperate *concept space* and determine their mutual similarities by means of their dot product in this space. The term space is indexed by the nodes of a semantic network and the corresponding feature weightings are derived from a number of conceptually well-motivated measures of semantic similarity.

We assess the performance of our approach by means of experiments on the well-known Reuters-21578 corpus using very small subsets of the typically employed 'ModApte' training set partitioning. Additionally, since the benefit of the above similarity metrics is emphasized

when data is highly affected by data sparseness [Basili *et al.*, 2005], we carried out a set of experiments in the domain of question classification (QC). Question classification aims at detecting the type of a question, e.g. whether it asks for a person or for an organization which is critical to locate and extract the right answers in question answering systems. A major challenge of question classification compared to standard text classification settings is that questions typically contain only extremely few words which makes this setting a typical victim of data sparseness.

Our evaluation studies indicate a consistent improvements of results in situations of little training data and data sparseness. The results on Reuters-21578 show that when only few training examples are available our kernels based on semantic similarity outperform one based on *bag-of-words*. The results of our second series of experiments indicate that improvements are even higher in the case of TREC question datasets independently of the size of the training examples.

The remainder of this paper is structured as follows. Section 2 provides preliminary notions on kernels and semantic networks. Section 3 presents a number of measures of lexical semantic relatedness and related notions that will be used for designing semantic kernels. Section 4 describes the design of the semantic kernels whereas Section 5 gives an account on the performance of these in a series of evaluation experiments. We review the related work in section 6 while concluding in section 7 with a summary of the contributions, final remarks and a discussion of envisioned future work.

2 Preliminaries

In this section, we briefly review the basic concepts of SVMs, Kernel Methods (section 2.1) and a few definitions and notions about semantic networks (section 2.2).

2.1 Support Vector Machines and Kernel Methods

Support Vector Machines are state-of-the-art learning methods based on the earlier idea of linear classification. The distinguishing feature of SVMs is the theoretically well motivated and efficient training strategy for determining the separating hyperplane based on the margin maximization principle. The other interesting property of SVMs is their capability of naturally incorporating data-specific notions of item similarity by means of a corresponding kernel function.

Definition 1 (Kernel Function)**.** *Any function κ that for all $x, z \in X$ satisfies $\kappa(x, z) = \langle \phi(x), \phi(z) \rangle$, is a valid kernel, whereby X is the input vector space under consideration and ϕ is a suitable mapping from X to a feature space F.*

Note that the choice of a particular kernel function implies an indirect mapping to a feature space different from the input space x. Kernels can be designed by either choosing an explicit mapping function ϕ and incorporating it into an inner product or by directly defining the kernel function κ while making sure that it complies with the requirement of being a positive semi-definite function. The reader is referred to the rich literature for further information on SVMs and kernel methods, e.g. [Müller *et al.*, 2001; Shawe-Taylor and Cristianini, 2004] for comprehensive introductions.

2.2 Semantic Networks

The target semantic dependencies are encoded in structures which we call, for simplicity, *semantic networks*. These can be seen as *directed graphs*.

Definition 2 (Semantic Network)**.** *A* semantic network *is a tuple $\mathcal{S} := (\mathcal{C}, R)$ consisting of a set $\mathcal{C}$ whose elements are called* concept identifiers*, and a relation $R \subseteq \mathcal{C} \times \mathcal{C}$ called* semantic link*. Often, we call concept identifiers just* concepts *and the semantic links just* links*, for sake of simplicity. For two concepts $c_1, c_2 \in \mathcal{C}$ and $(c_1, c_2) \in R$ we say that c_2 is* superconcept of c_1 *or vice versa that c_1 is* subconcept of c_2.

Our formalization is deliberately generic to capture a wide range of linguistic resources, taxonomies and ontologies. However, in this work, we restrict our attention to WordNet[1], a free lexical reference system and semantic network [Fellbaum, 1998]. WordNet organizes English terms into groups of synonyms (*synsets*) connected by a number of semantic relations. As most of the previous related work, we focus on the hypernym/hyponym relations for nouns that correspond to the superconcept/subconcept relations introduced above.

The measures nextly introduced require three further notions. By *distance* (*d*) of two concepts c_1 and c_2, we refer to the number of superconcept edges between c_1 and c_2. These can be easily computed from the network's adjacency matrix using the Floyd-Warshall algorithm [Floyd, 1962] for all pairs of concepts. The notion of the *depth* (*dep*) of a concept relates to the frequent assumption of a tree-like structure of the semantic network having a unique root element. For an acyclic graph (which we assume in the remainder), a root element can be introduced which becomes superconcept of all concept nodes that are not equipped with outgoing superconcept edges[2]. The depth of a concept is then defined as the distance of the concept to the root. Based on this, the *lowest super ordinate (lso)* of two concepts refers to the concept with maximal depth that subsumes them both.

3 Measuring Semantic Relatedness

The measurement of semantic similarity is a problem that pervades computational linguistics with respect to a large number of applications in natural language processing. A large amount of work has been devoted to defining measures of lexical semantic similarity or its opposite, lexical semantic distance[3] based on semantic networks – in most cases WordNet. In this section, we give a brief review of a number of measures of this type that have been used within this paper with emphasis on a compact description of the measures and the main rationales behind them, pointing the interested reader to [Budanitsky and Hirst, 2006] for a more detailed and most recent survey of the field.

[1] `http://www.cogsci.princeton.edu/~wn/`

[2] This is particularly true for the WordNet noun hierarchy, which up to version 2.0 defined 9 distinct *unique beginner concepts* up to which each concept can be traced.

[3] Note that [Budanitsky and Hirst, 2006] have made a good point in distinguishing the more general concept of *semantic relatedness* from *semantic similarity*. While this distinction is useful in terms of a fine-grained interpretation of the specific type of relation that ties two lexical entities together, it is not critical in the context of our work.

Path Based Measures The *inverted path length* can be seen as an example of particularly simple way to compute semantic similarity between two concepts in a semantic network:

$$sim_{IPL}(c_1, c_2) = \frac{1}{(1 + d(c_1, c_2))^{\alpha}},$$

whereby α specifies the rate of decay. Note that the [Siolas and d'Alché Buc, 2000] have used this measure to define semantic smoothing kernels for the first time. While the similarity of this measure is intriguing, it does not comply with the intuition that concepts closer to the root of the semantic network should have a higher distance compared to concepts far away. Among many others, the measure introduced by Wu&Palmer [Wu and Palmer, 1994] tries to scale the similarity with respect to the depth of the concepts and their lowest super ordinate in the semantic network:

$$sim_{WUP}(c_1, c_2) = \frac{2\, dep(lso(c_1, c_2))}{d(c_1, lso(c_1, c_2)) + d(c_2, lso(c_1, c_2)) + 2\, dep(lso(c_1, c_2))}.$$

Information Content Based Measures A different type of measures tries to incorporate additional knowledge about the information content of a concept besides the structural setup of the semantic network. Resnik [Resnik, 1999] has argued that neither the individual edges nor the absolute depth in a taxonomy can be considered as homogeneous indicators of the semantic content of a concept. To overcome this problem, he introduces the notion of the probability $P(c)$ of encountering a concept c. This probability is typically estimated by the relative frequencies of the lexicalizations of the concept in a corpus relevant for the target domain whereby the counts of subconcepts equally contribute to their respective superconcepts. Resnik follows the argumentation of information theory in quantifying the *information concept (IC)* of an observation as the negative log likelihood. Intuitively, a universal root concept having a probability of 1 carries an information content equal to zero while rare concepts carry high information content values. By means of the argument that *"one key to the similarity of two concepts is the extent to which they share information in common"* he proposes to measure the similarity of two concepts by means of the formula:

$$sim_{RES}(c_1, c_2) = -\log\, P(lso(c_1, c_2)).$$

Based on this proposal, Lin [Lin, 1998] derived a theoretically well motivated similarity measure given by:

$$sim_{LIN}(c_1, c_2) = \frac{2\, \log\, P(lso(c_1, c_2))}{\log\, P(c_1) + \log P\,(c_2)}.$$

As an extension to the original measure proposed by Resnik, the information content of the compared concepts is used as a means for normalization.

4 Designing Semantic Kernels

As motivated in section 1, the aim of our work is to embed the knowledge about the topological relations of the semantic networks in kernel functions. This allows the learning algorithm to relate distinct but similar features during kernel evaluation.

4.1 Semantic Kernels

The general concept of semantic smoothing kernels was for the first time introduced in [Siolas and d'Alché Buc, 2000] and subsequently revisited in [Cristianini *et al.*, 2002; Mavroeidis *et al.*, 2005; Basili *et al.*, 2005], each time based on different design principles.

Definition 3 (Semantic Smoothing Kernel)**.** *The semantic smoothing kernel for two data items (documents) $x, z \in X$ is given by $\kappa(x, z) = x'Qz$ where Q is a square symmetric matrix whose entries represent the semantic proximity between the dimensions of the input space X.*

Note that the definition of a kernel in section 2.1 implies that Q must be a positive semi-definite matrix. Conceptually this means that Q can be decomposed by $Q = PP'$ thus revealing the underlying feature mapping as $\phi(x) = P'x$. The matrix P is a $n \times m$ matrix whereby n corresponds to the dimensionality of the input space X and provides a linear transformation of the input document into a feature space of (possibly far higher) dimensionality m, similar to a query expansion. A first approach to designing semantic kernels would be to embed the pairwise measures of lexical semantic relatedness directly into the matrix Q. However, the requirement of Q being positive semi-definite can typically not be ensured for all measures in general if used directly.

4.2 Semantic Kernels based on Superconcept Expansions

As a way to avoid indefinite similarity matrices, authors like [Siolas and d'Alché Buc, 2000] have enforced the positive-definiteness of Q by exlicitely computing it from $Q = PP'$ whereby the information about the similarities is now encoded in the matrix P. While this approach ensures the validity of the Kernel, the interpretation of the resulting smoothing kernel is less clear. Conceptually, it maps each concept to a number of related concepts and the shared weight of these determines the overall similarity between two terms.

Following own prior work in a differnet setting [Bloehdorn and Hotho, 2004], we follow a different approach for the construction of Q which is, however, also based on an explicit construction of the type $Q = PP'$. We choose a setup of P such that it provides a mapping into the space of all possible *superconcepts* of the input instances, i.e. the terms or concepts in question. That is, the rows of P correspond to vector representations of the concepts of the input space by means of their respective superconcepts. The similarity of two concepts in the resulting smoothing matrix Q is thus the dot product of the vectors of their respective superconcepts. This approach is intuitive as we can typically regard two concepts as similar if they share a large number of superconcepts as opposed to sharing only few superconcepts.

Recently, [Mavroeidis *et al.*, 2005] have proposed this approach motivated by the observation that the dot product of two terms represented as vectors of their respective superconcepts can be shown to be equivalent to a number of popular similarity measures (among them the Resnik measure, but not the Lin and Wup measures) given a particular weighting scheme of the superconcept representation. However, this prior work has focused on the simple case of giving the superconcepts in the mapping P full and equal weight (i.e. restricting P to a 0/1 matrix) while varying the number of superconcepts that are considered. Consistent

with an argument made by the same authors, we argue that the variation of the number of superconcepts yields a high variance and its a-priori choice will always be an ad-hoc decision.

As an alternative approach, we have investigated the use of different weighting schemes for the representation of the superconcepts in P motivated by the following considerations:

1. The weight a superconcept c_j receives in the vectorial description of a concept c_i should be influenced by its distance from c_i.
2. The weight a superconcept c_j receives in the vectorial description of a concept c_i should be influenced by its overall depth in the semantic network.

Based on these rationales and the measures introduced in section 3, we have investigated the following weighting schemes:

full: No weighting, i.e. $P_{ij} = 1$ for all superconcepts c_j of c_i and $P_{ij} = 0$ otherwise.

full-ic: Weighting using information content of c_j, i.e. $P_{ij} = sim_{RES}(c_i, c_j)$.

path-1: Weighting based on inverted path length, i.e. $P_{ij} = sim_{IPL}(c_i, c_j)$ for all superconcepts c_j of c_i and $P_{ij} = 0$ otherwise using the parameter $\alpha = 1$.

path-2: The same but using the parameter $\alpha = 2$.

lin: Weighting using the Lin similarity measure, i.e. $P_{ij} = sim_{LIN}(c_i, c_j)$.

wup: Weighting using the Wu&Palmer similarity measure, i.e. $P_{ij} = sim_{WUP}(c_i, c_j)$.

The different weighting schemes behave differently wrt the above motivations. While full does not implement any of them, full-ic considers rationale 2 while path-1 and path-2 consider rationale 1. The schemes lin and wup reflect combinations of both rationales.

5 Experimental Evaluation

In a series of experiments we aimed at showing that our approach is effective for IR and data mining applications. For this purpose, we experimented with two different datasets related to two different mining tasks: Reuters-21578 for traditional Text Categorization and TREC question classification corpus for advanced retrieval based on the Question Answering paradigm.

5.1 Experimental Setup

We implemented the semantic kernel within a custom kernel module for the current version of SVMlight[4] which is freely available for download[5]. For both Reuters-21578 and TREC datasets, we used the noun hierarchy of WordNet as the underlying semantic network. We first describe the general setup of the smoothing matrices in the following section whereas the results are reported in sections 5.2 and 5.3.

[4] http://svmlight.joachims.org/

[5] http://www.aifb.uni-karlsruhe.de/WBS/sbl/software/semkernel/

Proximity Matrix Setup

The setup of the smoothing matrices used in the evaluation experiments was based on the particular choice of the proximity matrix design, discussed in section 4, as well as on two simplifying assumptions.

Firstly, the existing bag-of-word representation of the documents required the design of a *term proximity matrix* as opposed to the *synset proximity matrix* implicitly assumed so far. We used a simple strategy that maps each term to its most frequent noun sense (if it exists). Note that this approach implies an inherent word sense disambiguation side effect, both with respect to the respective part-of-speech as well as to the chosen noun sense. While this effect is likely to have a negative impact on the results, the error introduced by this approach is systematic. In the light of these considerations, the results can also be seen as a pessimistic estimate of the potential effectiveness given a perfectly disambiguated input.

Secondly in the case of the Reuters-21578 experiments, we restricted the entries in the term proximity matrix to those terms having document frequencies of at least five. This speeds up the computation during kernel evaluation while we used the full term similarity matrix in the case of the question classification experiments. Entries that were undefined in the term proximity matrix – be it because a missing mapping to a noun synset or because of low document frequency – were implicitly assumed to take the default values (i.e. zero and one for off-diagonal and diagonal entries respectively) during kernel evaluation[6]. Frequency counts needed for the calculation of the measures making use of information content were obtained from (i) the complete Reuters-21578 collection in the case of the Reuters-21578 experiments or (ii) from the Brown corpus in the case of the experiments on the TREC question dataset[7].

5.2 Experiments on Reuters-21578

As basis for our experiments on Reuters-21578 we used the 'ModApte' split which divides the Reuters-21578 collection into 9,603 training documents, 3,299 test documents and 8,676 unused documents. We prepared the bag-of-words representation of the documents based on the standard preprocessing steps, namely tokenization, removal of the standard stopwords for English defined in the SMART stopword list and lemmatization, resulting in a total number of 32,443[8] distinct features which were all weighted using the standard TFIDF scheme.

Based on results of previous work, we were well aware of the fact that the introduction of prior semantic knowledge typically has a small effect when sufficient training data is available and in some cases may even degrade the performance compared to the linear kernel. In our experiments, we thus primarily aimed at quantifying performance

[6] Technically, this can be seen as defining the overall kernel κ as the sum of two individual kernels: $\kappa(x, y) = \kappa_s(x_s, z_s) + \kappa_t(x_t, z_t)$ whereby κ_s is the semantic smoothing kernel as introduced above and κ_t is the conventional linear kernel, defined on the vectors formed by restriction to the dimensions indexed in the smoothing matrix and the remaining dimensions respectively.

[7] This decision was motivated by the fact that word frequency estimations on the dataset itself would be rather unreliable due to its far smaller overall size.

[8] The high number of features compared to other published work is an effect of the preprocessing scheme that we used which includes sequences of digits as features. This is, however, unlikely to have a significant effect on the results.

gains in those cases where very little training data was available. For this purpose, we prepared small subsets of the ModeApte training set by randomly choosing 2%, 3%, 4% and 5% of the available training data. To account for the high inherent sampling variance, this approach was repeated 10 times for each of the 4 subset sizes resulting in a total number of 40 subsets. Note that we checked that at least one positive training document for each of the aforementioned 10 categories was present in every created training subset. Binary classification experiments were then conducted for each category and each subset resulting in a total number of 400 experiments in each run. While in each of these experiments the SVM classifier was trained using the respective subset, the corresponding testing was conducted using the full ModeApte test set.

Evaluation Results Table 1 summarizes the absolute macro F_1 values obtained over the different subsets of Reuters-21578 as explained above. The 'soft margin' parameter c that controls the influence of misclassified examples was set to $c = 0.1$ in all experiments. The results indicate a consistent improvement of the F_1 values for all of the smoothing kernels based on superconcept representations.

	Subset Size			
kernel	**02p**	**03p**	**04p**	**05p**
linear	0.45	0.51	0.54	0.57
full	0.50	0.53	0.57	0.58
full-ic	0.53*	0.55	0.60	0.61
path-1	0.50	0.54	0.59	0.61
path-2	0.48	0.53	0.57	0.59
lin	0.53*	0.57*	0.61*	0.62*
wup	0.52	0.55	0.59	0.61

Table 1: Absolute macro F1 results for Reuters-21578 subsets and different semantic smoothing kernels. The best result per subset is highlighted.

The extent of the improvement for the smoothing kernels based on superconcept representations relative to the linear kernel can be seen more clearly in figure 1. According to prior findings in [Mavroeidis *et al.*, 2005] the improvement gradually diminishes as more training data becomes available. Among the different weighting schemes for superconcept representations, the lin weighting scheme consistently outperforms the other measures. This finding confirms our assumptions on the desired structure of the weighting scheme as the Lin measure respects both the overall depth of the respective superconcept by virtue of the information content as well as the distance from the base concept by means of the difference in information content. On the contrary, the default scheme (full) that does not employ any weighting schemes tends to be inferior to other models that use them.

While we were not primarily interested in the application of our approach in those cases where sufficient training data is available, we have nevertheless investigated the effect of a selection of superconcept smoothing kernels when the full ModeApte training set is used. Figure 2 summarizes the results in terms of per-category F_1 values. The results indicate only little shifts in performance, typically degrading performance compared to the linear kernel to a small extent. This finding supports our assumption that

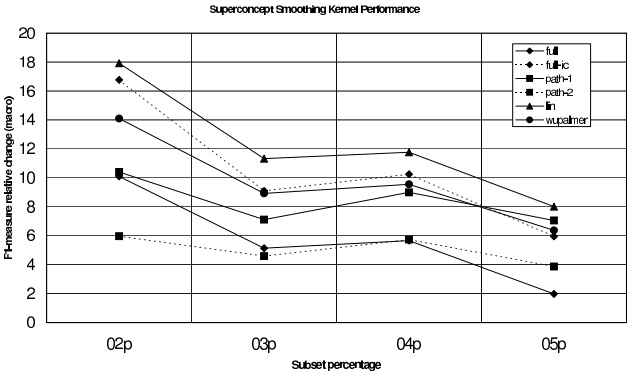


Figure 1: Relative improvements of the macro F1 results for Reuters-21578 subsets and different superconcept-based semantic smoothing kernels.

the semantic smoothing kernels are not particularly useful in scenarios where training data isn't scarce. Also note that the comparatively high complexity of semantic kernels limits the practical application for training based on large amounts of training datas.

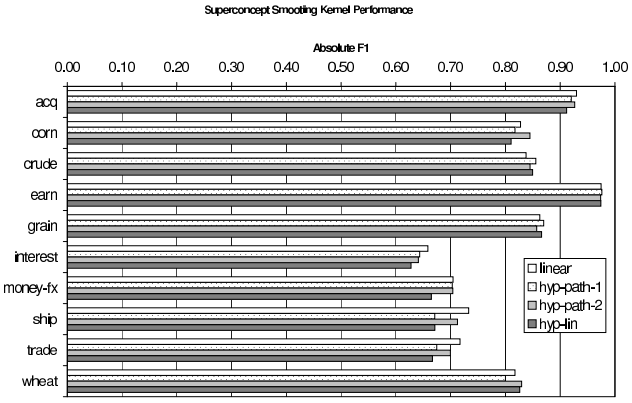


Figure 2: Absolute F1 results for 10 Reuters-21578 categories and selected superconcept-based semantic smoothing kernels using the full training sets.

5.3 Experiments on the TREC Question Classification Dataset

The long tradition of QA in TREC has produced a large question set used by several researchers which can be exploited for experiments on question classification (QC). Such questions are categorized according to different taxonomies of different *grains*. We consider the *coarse grained* classification scheme described in [Zhang and Lee, 2003; Li and Roth, 2002]: Abbreviations, Descriptions (e.g. *definition* and *manner*), Entity (e.g. *animal*, *body* and *color*), Human (e.g. *group* and *individual*), Location (e.g. *city* and *country*) and Numeric (e.g. *code* and *date*).

We used a set of questions labeled according to the above taxonomy. This dataset has also been employed in [Zhang and Lee, 2003; Li and Roth, 2002] and is freely available[9]. It is divided into 5,500 questions[10] for training and the 500 TREC 10 questions for testing. Similarly to the

[9] http://l2r.cs.uiuc.edu/~cogcomp/Data/QA/QC/

[10] These are selected from the 4500 English questions published by USC (Hovy et al., 2001), 500 questions annotated for rare classes and the 894 questions from TREC 8 and TREC 9.

first experiment, we preprocessed the questions using the usual steps leading to a total number of 8,075 distinct features weighted according to standard TFIDF. Again, we performed binary classification experiments on each of the 9 question types.

Evaluation Results In this experiment, we additionally applied several values of the 'soft margin' parameter c since our preliminary tests showed that its variation has an important influence on the overall results. Starting from $c = 0.1$ and $c = 1.0$ as typical default choices, we varied these in three steps to $c = 0.1 \ldots 0.3$ and $c = 1 \ldots 3$. Table 2 summarizes the absolute macro F_1 as well as the micro F_1 values obtained in the question classification setting. The best values per setting of c are highlighted.

macro-averaging						
	soft margin parameter *c*					
kernel	**0.1**	**0.2**	**0.3**	**1.0**	**2.0**	**3.0**
linear	0.21	0.38	0.47	0.62	0.63	0.64
full	0.38	0.49	0.55	0.61	0.61	0.68
full-ic	0.53*	0.53*	0.53	0.62	0.55	0.55
path-1	0.25	0.42	0.51	0.64*	0.64	0.64
path-2	0.22	0.39	0.47	0.63	0.65*	0.64
lin	0.36	0.49	0.56*	0.64*	0.62	0.70*
wup	0.34	0.49	0.54	0.62	0.61	0.69

macro-averaging						
	soft margin parameter *c*					
kernel	**0.1**	**0.2**	**0.3**	**1.0**	**2.0**	**3.0**
linear	0.09	0.25	0.34	0.55	0.57	0.58
full	0.27	0.38	0.45	0.55	0.56	0.68
full-ic	0.47*	0.46*	0.47*	0.60*	0.49	0.48
path-1	0.14	0.32	0.40	0.57	0.58	0.59
path-2	0.08	0.28	0.37	0.57	0.59*	0.58
lin	0.27	0.37	0.47*	0.57	0.57	0.69*
wup	0.23	0.37	0.45	0.56	0.56	0.68

Table 2: Absolute macro and micro F1 results for QC, for different values of *c* and different semantic smoothing kernels. The best results per setting of *c* are highlighted

The results indicate a consistent superior accuracy of the semantic smoothing kernels over the linear kernel baseline. With the exception of the full-ic setup, which shows good results for small values of c but detoriates later on, all semantic smoothing kernels improve performace in both the macro- as well as micro-averaged setting. According to the results on the Reuters-21578 experiments, the lin scheme achieves the best overall performance with a relative improvement of 9.32% for the macro F_1 value in the case of $c = 3$ (i.e. the setting for which the linear kernel achieves its maximum). We generally note that the improvements are more extreme for the case of small values of c while they appear more stable for larger values.

6 Related Work

To date, the work on integrating prior knowledge about feature similarities into text classification or other related tasks is quite scattered. Much of the early work in this direction was done in the context of *query expansion* techniques as e.g. reported in [Bodner and Song, 1996]. Early work in the direction of incorporation semantic background knowledge in combination with the Ripper classification algorithm was reported in [Scott and Matwin, 1999]. However, this early work showed negative results on two independent data sets. An alternative approach motivated by the idea of letting terms and higher level semantic features (including fixed depths of hypernyms) compete within the boosting algorithm paradigm was reported in [Bloehdorn and Hotho, 2004].

Semantic kernels were initially introduced in [Siolas and d'Alché Buc, 2000] using inverted path length as a similarity measure and subsequently explored in [Basili *et al.*, 2005] using conceptual density as a similarity measure among others. An alternative approach reported in [Cristianini *et al.*, 2002] aimed at incorporating the well-established technique of Latent Semantic Indexing (LSI) into the semantic kernel paradigm. As [Cristianini *et al.*, 2002] have pointed out, a similar framework has been used in [Jiang and Littman, 2000], although without the explicit notion of kernel functions. Recently [Mavroeidis *et al.*, 2005] reported on experiments with semantic smoothing kernels defined on superconcept representations such that it forms a natural basis for our work. In contrast to our approach, the authors used extensive word sense disambiguation (WSD) machinery which also formed a core contribution. Similar to [Bloehdorn and Hotho, 2004], the superconcept representations of terms were built upon fixed numbers of superconcepts without further weighting.

7 Conclusion

In this paper, we have investigated the design of semantic smoothing kernels. We similar framework to the one used in [Mavroeidis *et al.*, 2005] which expresses the similarity of term features by means of the shared superconcepts. In contrast to earlier work in this direction, we employed theoretically well motivated measures of semantic similarity between the base concepts under consideration and their corresponding superconcepts.

We conducted a series of experiments on the Reuters-21578 corpus using different sizes of training subsets and on the TREC question classification data. Our results indicate a consistent improvement in performance for superconcept semantic smoothing kernels in those cases where little training data is available or the feature representations are extremely sparse. Especially the lin scheme as proved to be a weighting scheme with stable improvements.

As both [Mavroeidis *et al.*, 2005] and [Bloehdorn and Hotho, 2004] have pointed out, the success of the introduction of semantic background knowledge in text-mining tasks critically depends on the employed word sense disambiguation strategy. Our experiments were deliberatly kept simple and thus did not use a decent word sense disambiguation step. We thus expect a further improvement in results when a powerful WSD technique (e.g. the one explored in [Mavroeidis *et al.*, 2005]) is applied. We aim at investigating this issue together with experiments on other corpora and the exploitation of semantic relations different from those based on superconcepts. We also aim at employing semantic kernels in scenarios different from text classification where the target background knowledge may take the form of arbitrary ontological structures. As a different trail we will investigate the combination of our semantic kernels with other types of kernels that also exploit more input structure.

Acknowledgements

This research was partially supported by the European Commission under contract IST-2003-506826 SEKT. The expressed con-

tent is the view of the author(s) but not necessarily the view of the SEKT consortium.

References

[Basili *et al.*, 2005] Roberto Basili, Marco Cammisa, and Alessandro Moschitti. A semantic kernel to classify texts with very few training examples. In *In Proceedings of the Workshop on Learning in Web Search, at the 22nd International Conference on Machine Learning (ICML 2005), Bonn, Germany*, 2005.

[Bloehdorn and Hotho, 2004] Stephan Bloehdorn and Andreas Hotho. Text classification by boosting weak learners based on terms and concepts. In *Proceedings of the 4th IEEE International Conference on Data Mining (ICDM 2004), 1-4 November 2004, Brighton, UK*, pages 331–334. IEEE Computer Society, NOV 2004.

[Bodner and Song, 1996] R. C. Bodner and F. Song. Knowledge-Based Approaches to Query Expansion in Information Retrieval. In *Advances in Artificial Intelligence*. Springer, New York, NY, USA, 1996.

[Budanitsky and Hirst, 2006] Alexander Budanitsky and Graeme Hirst. Evaluating wordnet-based measures of lexical semantic relatedness. *Computational Linguistics*, 32(1):13–47, March 2006.

[Cristianini *et al.*, 2002] Nello Cristianini, John Shawe-Taylor, and Huma Lodhi. Latent semantic kernels. *Journal of Intelligent Information Systems*, 18(2-3):127–152, 2002.

[Fellbaum, 1998] Christiane Fellbaum, editor. *WordNet - An Electronic Lexical Database*. MIT Press, 1998.

[Floyd, 1962] Robert W. Floyd. Algorithm 97: Shortest path. *Commun. ACM*, 5(6):345, 1962.

[Jiang and Littman, 2000] Fan Jiang and Michael L. Littman. Approximate dimension equalization in vector-based information retrieval. In *Proceedings of the 7th International Conference on Machine Learning. Stanford University June 29-July 2, 2000*, pages 423–430, 2000.

[Joachims, 1998] Thorsten Joachims. Text categorization with suport vector machines: Learning with many relevant features. In Claire Nedellec and Céline Rouveirol, editors, *Proceedings of ECML 1998*, volume 1398 of *Lecture Notes in Computer Science*, pages 137–142. Springer, 1998.

[Li and Roth, 2002] Xin Li and Dan Roth. Learning question classifiers. In *Proceedings of the 19th International Conference on Computational Linguistics (COLING'02)*, 2002.

[Lin, 1998] Dekang Lin. An information-theoretic definition of similarity. In Jude W. Shavlik, editor, *Proceedings of the Fifteenth International Conference on Machine Learning (ICML 1998), Madison, Wisconson, USA, July 24-27, 1998*, pages 296–304. Morgan Kaufmann, 1998.

[Mavroeidis *et al.*, 2005] Dimitrios Mavroeidis, George Tsatsaronis, Michalis Vazirgiannis, Martin Theobald, and Gerhard Weikum. Word sense disambiguation for exploiting hierarchical thesauri in text classification. In Alípio Jorge, Luís Torgo, Pavel Brazdil, Rui Camacho, and João Gama, editors, *Knowledge Discovery in Databases: Proceedings of the 9th European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD 2005), Porto, Portugal, October 3-7, 2005*, pages 181–192. Springer, 2005.

[Müller *et al.*, 2001] K.-R. Müller, S. Mika, G. Rätsch, S. Tsuda, and B Schölkopf. An introduction to kernel-based learning algorithms. *IEEE Transactions on Neural Networks*, 12(2):181–202, 2001.

[Resnik, 1999] Philip Resnik. Semantic similarity in a taxonomy: An information-based measure and its application to problems of ambiguity in natural language. *Journal of Artificial Intelligence Research*, 11:95–130, 1999.

[Scott and Matwin, 1999] Sam Scott and Stan Matwin. Feature engineering for text classification. In Ivan Bratko and Saso Dzeroski, editors, *Proceedings of the Sixteenth International Conference on Machine Learning (ICML 1999), Bled, Slovenia, June 27 - 30, 1999.*, pages 379–388. Morgan Kaufmann, 1999.

[Sebastiani, 2002] F. Sebastiani. Machine Learning in Automated Text Categorization. *ACM Computing Surveys*, 34(1):1–47, 2002.

[Shawe-Taylor and Cristianini, 2004] John Shawe-Taylor and Nello Cristianini. *Kernel Methods for Pattern Analysis*. Cambridge University Press, 2004.

[Siolas and d'Alché Buc, 2000] Georges Siolas and Florence d'Alché Buc. Support vector machines based on a semantic kernel for text categorization. In *IJCNN '00: Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks (IJCNN'00)-Volume 5*, page 5205, Washington, DC, USA, 2000. IEEE Computer Society.

[Vapnik *et al.*, 1997] V. Vapnik, S. Golowich, and A. Smola. Support vector method for function approximation, regression estimation, and signal processing. In M. C. Mozer, M. I. Jordan, and T. Petsche, editors, *Advances in Neural Information Processing Systems*, volume 9, pages 281–287, Cambridge, MA, 1997. MIT Press.

[Wu and Palmer, 1994] Zhibiao Wu and Martha Stone Palmer. Verb semantics and lexical selection. In *Proceedings of the 32nd. Annual Meeting of the Association for Computational Linguistics (ACL 1994)*, pages 133–138, 1994.

[Zhang and Lee, 2003] Dell Zhang and Wee Sun Lee. Question classification using support vector machines. In *SIGIR '03: Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pages 26–32, New York, NY, USA, 2003. ACM Press.

Mining Data Streams under Dynamicly Changing Resource Constraints

Conny Franke
Department of Computer Science,
University of California at Davis, USA

Marcel Karnstedt **Kai-Uwe Sattler**
Department of Computer Science and Automation,
TU Ilmenau, Germany

Abstract

Due to the inherent characteristics of data streams, appropriate mining techniques heavily rely on window-based processing and/or (approximating) data summaries. Because resources such as memory and CPU time for maintaining such summaries are usually limited, the quality of the mining results is affected in different ways. Based on *Frequent Itemset Mining* and an according *Change Detection* as selected mining techniques, we discuss in this paper extensions of stream mining algorithms allowing to determine the output quality for changes in the available resources (mainly memory space). Furthermore, we give directions how to estimate resource consumptions based on user-specified quality requirements.

1 Introduction

Stream mining has recently attracted attention by the database as well as the data mining community. The goal of stream mining is a fast and adaptive analysis of data streams, i.e., the discovery of patterns and rules in the data. An important task of traditional mining as well as stream mining is frequent itemset mining, that aims to identify combinations of items which occur frequent in a given sequence of transactions. Typical applications of mining data streams are click stream analysis, analysis of records in networking and telephone services and analysis of sensor data among others.

Another popular problem, especially for continuous data streams, is the detection of changes in the data. This includes aspects such as changes in the distribution of the data (possibly expressed in statistical terms like median or quantiles) and burst detection, and also task specific change detection, e.g., recognizing changes in the set of frequent itemsets, in the frequency of particular itemsets or changing correlations between itemsets. Concerning the problem of change detection, the challenge of in-time processing and signalizing gains additional importance beside resource restrictions.

The main challenge in applying mining techniques to data streams is that a stream is theoretically infinite and therefore in most cases cannot be materialized. That means that the data have to be processed in a single pass using little memory. Based on this restriction and the goals of data mining one can identify two divergent objectives: On the one hand the analysis should produce comprehensive and exact results and detect changes in the data as soon as possible. On the other hand the single pass demand and the problem of resource limitations allow only to perform the analysis on an approximation of the stream (e.g., samples or sketches) or a window (i.e., a finite subset of the stream).

However, using an approximation or a subset of the stream affects the quality of the analysis result: The mining model differs from the mining model we would get if the "whole" stream or a larger subset is considered. This is particularly important because some of the proposed stream mining approaches support time sensitiveness (reducing the influence of outdated stream elements) by using weaker approximations for outdated elements. Thus, the mining quality for these elements is worse than for newer data. The problem of the quality of the mining model can also be considered in the opposite direction: Based on user-specified quality requirements one could derive resource requirements, i.e., the memory needed for managing stream approximations in order to guarantee the requested quality.

We propose resource awareness in conjunction with quality awareness as one of the main requirements in stream mining – and challenges in parallel. Fig. 1 illustrates how the dependencies between both are integrated into the operational flow of stream processing. Two ways of putting resource and quality awareness into practice get evident:

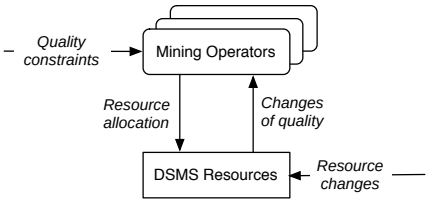


Figure 1: Mapping of quality

1. Claim for specific quality requirements and deduce the needed resources to achieve this quality.
2. Limit the resources provided for processing and deduce the achievable quality.

Based on this observation, we propose a resource-adaptive and quality-aware mining approach for frequent itemset mining. We argue that quality awareness is basically orthogonal to the specific mining problem, even if the individual mining approach requires dedicated techniques for considering mining quality. For this purpose, we discuss the general applicability of our approach and outline specific characteristics that potential mining techniques have to meet.

The remainder of this paper is structured as follows. After a brief survey of related work in Section 2 we introduce relevant quality measures in Section 3. In Section 4 we present our extended frequent itemset mining approach by adding quality measuring as well as a resource adaption based on user-specified quality requirements. Based on

this, we discuss approaches for resource-adaptive change detection in Section 4.3. After discussing the applicability of our approach to general mining problems in Section 4.4, we report results of an experimental evaluation in Section 5. Finally, we conclude the paper and discuss open issues for future work in Section 6.

2 Related Work

In this paper, we discuss strategies for adaptively mining data streams in general. We will develop our considerations on the basis of a corresponding approach for Frequent Itemset Mining proposed in [Franke *et al.*, 2005].

Frequent itemset mining deals with the problem of identifying sets of items occurring together in so-called transactions frequently. Any itemset occurring in more than σ (the *support*) percentage of all input transactions is regarded as frequent. Usually, a deviation of ε in the support of an itemset is accepted. σ and ε are (optionally dynamicly) user-defined parameters. Basically, two classes of algorithms can be distinguished: approaches with candidate generation (e.g., the famous apriori algorithm) as well as without candidate generation. Here, only the latter ones are suitable for stream mining. Usually, these approaches are based on a prefix-tree-like structure. In this tree – the frequent pattern (FP) tree – each path represents an itemset in which multiple transactions are merged into one path or at least share a common prefix if they share an identical frequent itemset [Han *et al.*, 2000]. For this purpose, items are sorted as part of their transaction in their frequency descending order and are inserted into the tree accordingly. Again, the FP tree is used as a compact data summary structure for the actual (offline) frequent pattern mining (the FP-growth algorithm).

In order to mine streaming data in a time-sensitive way an extension of this approach was proposed [Giannella *et al.*, 2003a]. Here, so-called tilted time window tables are added to the nodes representing window-based counts of the itemsets. The tilted windows allow to maintain summaries of frequency information of recent transactions in the data at a finer granularity than older transactions. The extended FP tree, called pattern tree, is updated in a batch-like manner: incoming transactions are accumulated until enough transactions of the stream have arrived. Then, the transactions of the batch are inserted into the tree. For mining the tree a modification of the FP-growth algorithm is used taking the tilted window table into account. The original approach assumes that there is enough memory available to deliver results in any required quality (expressed in terms of σ and ε, see Section 3) and no way is described how to proceed if the algorithm runs out of memory. In Section 4.2 we will discuss how the amount of required memory can be adjusted and how this affects the result quality.

In recent time, the idea of processing streams while adapting to resource and quality constraints gained boosted attention. There are several recent works dealing with quality-aware online query processing applications in centralized and distributed systems, e.g., [Berti-Équille, 2006]. But, only a couple discusses concrete approaches and algorithms in stream mining scenarios, least of all the conjunction of resource adaptiveness and quality awareness.

In [Gaber *et al.*, 2003] the authors propose a resource-adaptive mining approach based on similar goals as our work. They suggest the adaptation to memory requirements, time constraints and data stream rate by solely adapting the produced output granularity. The paper focuses on clustering, but the authors state the applicability to classification and frequent item mining (but not itemset mining). In succeeding works, e.g., [Gaber and Yu, 2006], they extend their approach by adding resource adaptiveness to the input rate and the actual data mining algorithm as well. However, the approach lacks providing quality guarantees for the mining results.

The algorithm RAM-DS (Resource-Aware Mining for Data Streams) proposed in [Teng *et al.*, 2004] uses a wavelet-based approach to control the resource requirements. It is mainly concerned with mining temporal patterns and the method can only be used in combination with a certain regression-based stream mining algorithm proposed by the same authors. Although the proposed algorithm for mining temporal patterns is resource-aware, it is not resource-adaptive and does not provide guarantees for the quality of the mining results.

Q	**Output**	**Factors**
Q_{Ma}	ε/σ	queried time interval
Q_{Mi}	σ	-
Q_{Tr}	maximal "look back time"	-
Q_{Tg}	minimal granularity	queried "look back time"
Q_{Tc}	minimal time till detection of changes	update interval u

Table 1: Quality measures and influencing factors

3 Quality Measures for Stream Mining

As stated in section 1, any resource adaptiveness comes along with effects on the achievable result quality. In [Franke *et al.*, 2005] we introduced different quality measures we examine in the context of stream mining. As a result, we distinguish several different classes of quality measures, which are summarized together with exemplarily chosen representatives in Table 1. For a more detailed discussion we refer to [Franke *et al.*, 2005].

All Q_{T*} are identical for different mining problems and symbolize concrete measures, while Q_{M*} represent classes of quality measures that are always specific to the investigated problem and the applied algorithm(s). A special measure for the problem of frequent itemset mining is the ratio between a value ε and the support σ, which reflects the maximal deviation from the defined support each of the itemsets finally identified as frequent could possess. In the context of frequent itemsets the support is a so called interesting measures Q_{Mi} ([Tan and Kumar, 2000]).

From our point of view, any mining techniques applied on continuous and evolving data streams should take time sensitiveness into account, thus, we define time as another important quality measure. Q_{Tr} describes how far we can look back into the history of the processed stream and Q_{Tg} how exact we can do this, which means which time granularity we can provide. Q_{Tc} corresponds to one of the main challenges of stream mining: the actual time we need in order to register changes in the stream. These temporal quality measures must not be confused with temporal aspects that influence the methodical quality (see Table 1).

For the remainder of this paper, if we refer to all quality measures as a whole, we will use the symbol of the superclass Q and the general term 'quality'.

This work aims for determining two (theoretical) kinds of functions:

1. $r : Q \rightarrow R$ - maps claimed quality to the resources needed, and
2. $\overline{r} : R \rightarrow Q$ - maps provided resources to the best achievable quality, as an inverse function to r.

More detailed, r is one function $r(args_x, Q_x(args_x), args_y, Q_y(args_y), \ldots)$ taking all claimed quality measures $Q_x, Q_y, \ldots$ and their factors as input, but we write $r(Q)$ for short. In contrast, $\overline{r}$ represents a bundle of inverse functions $\overline{r_x}, \overline{r_y}, \ldots$, each corresponding to one quality measure $Q_x, Q_y, \ldots$. Moreover, as we do not state the distribution of the stream elements as input factor for any function, r and $\overline{r}$ differ with different stream characteristics.

4 Resource-aware Mining Operators

4.1 Operators for Data Streams

The techniques proposed in this work are implemented on top of a Data Stream Management System (DSMS) called PIPES [Krämer and Seeger, 2004]. Rather than a monolithic DSMS, PIPES is an infrastructure that, in conjunction with the comprehending Java library XXL [d. Bercken *et al.*, 2001], allows for building a DSMS specific to concrete applications with full functionality. Usually, DSMS manage multiple continuous queries specified by operator graphs, which allows for reusing shared subqueries. PIPES adapts this concept and introduces three types of graph nodes: sources, sinks and operators, where operators combine the functionality of a source and a sink with query processing functionalities. The resulting query graphs can be build and manipulated dynamically using an inherent publish and subscribe mechanism. This offers, among others, the possibility to adaptively optimize the processing according to resource awareness. PIPES provides the operational run-time environment to process and optimize queries as well as a programming interface to implement new operators, sources and sinks.

The aimed resource awareness mainly arises from implementing the mining techniques as separate PIPES operators. But why do we implement them as operators, rather than, for instance, building specialized DSMS for clustering and frequent itemset mining? The answer is, we want to be able to freely choose among any combination of these mining techniques between themselves, with other mining algorithms and, generally, with all operators implemented in PIPES. Fig. 2 pictures a small example to illustrate this approach (each operator is pictured by its corresponding algebra symbol). The stream data produced by two sources O_1 and O_2 is processed in three ways: sink S_1 receives all clusters determined (ζ) for O_1 after a preprocessing filter step. S_2 and S_3 receive association rules determined (Φ) on joined data from O_1 and O_2 – this implies finding the frequent itemsets (ϕ). S_3 works on the determined rules directly, while S_2 receives them after applying another clustering step (ζ) in order to identify interesting rules by grouping the related ones (similar to the approach in [Toivonen *et al.*, 1995]).

The frequent itemset mining operator takes three dynamically adjustable parameters:

1. The size b of a batch.
2. The queried time interval $[t_s, t_e]$.
3. An output interval o.

b represents the finest granularity of observed time and is equal to the internal update interval u. Thus, o should be a multiple of b. The resulting output is a data stream consisting of one stream element per passed output interval, containing the frequent itemsets found in $[t_s, t_e]$. In correspondence to the aimed resource awareness a user can decide between two possibilities to initialize the operator by providing:

1. Claimed qualities.
2. A memory limit.

In the first case, the amount of memory is calculated by adapting parameters inside the operator to achieve the claimed quality. For the second case, the adapting technique tries to find the ideal parameter settings to achieve optimal quality results, if possible, while adhering to the given memory limit. In order to achieve this, the user must define which of the supported qualities is prioritized and/or weight them accordingly.

4.2 Frequent Itemset Mining

We record two main requirements of frequent itemset mining techniques in order to lend themselves for our needs: they are able to provide error guarantees (frequent itemset mining on data streams usually produces approximate results, e.g., there may be some false positives in the resulting output), and the approach has to be time-sensitive. The FP-Stream approach in [Giannella *et al.*, 2003a] is capable to satisfy these requirements. Asked for the frequent itemsets of a time period $[t_s, t_e]$, FP-Stream guarantees that it delivers all frequent itemsets in $[t_{s'}, t_{e'}]$ with frequency $\geq \sigma \cdot W$, where W is the number of transactions the time period $[t_{s'}, t_{e'}]$ contains. $t_{s'}$ and $t_{e'}$ are the time stamps of the used tilted time window table (TTWT) that correspond to t_s and t_e, depending on Q_{Tg}. The result may also contain some itemsets whose frequency is between $(\sigma - \varepsilon) \cdot W$ and $\sigma \cdot W$.

Our first goal was to find out how much memory the algorithm needs in order to deliver results in a certain quality. We conducted an extensive series of tests for the algorithm's memory requirements in different parameter settings. Secondly, we extended the approach from [Giannella *et al.*, 2003a] so we can cope with limited memory, resulting in the algorithm's resource and quality awareness.

Estimating the amount of memory

Firstly, we estimate the amount of memory a pattern tree needs within a given parameter setting. For estimating the overall memory requirements of a pattern tree, we need to know the number of nodes in a pattern tree, and the amount of memory each individual node needs. In [Giannella *et al.*, 2003a] an upper bound is given for the size of a TTWT by $2 \lceil log_2(N) \rceil + 2$, where N is the number of batches seen so far. This is because the algorithm uses a logarithmic TTWT to the basis 2 and is designed with one buffer value between each two values except the two most recent. In our experiments the actual number of entries averaged over all TTWTs in a tree was always less than 70% of this upper bound. Besides the size of the TTWT, each node in a pattern tree needs some fixed sized memory c for storing the itemset it represents and information about its parent node and child nodes.

The number of nodes in a pattern tree depends on several facts. On the one hand there are the values of the algorithm's input parameters ε and b. The value of σ does not affect the number of nodes in a pattern tree, because the adding and dropping conditions in a pattern tree depend only on ε, though σ is a quality measure! On the other hand there are the characteristics of the underlying data stream like the average number of items per transaction and the number of distinct items in the stream. We have not yet found a concrete formula describing the maximum number of nodes in a pattern tree for specific parameter settings. But, we conducted a series of tests showing that the maximum number of nodes remains constant over time while

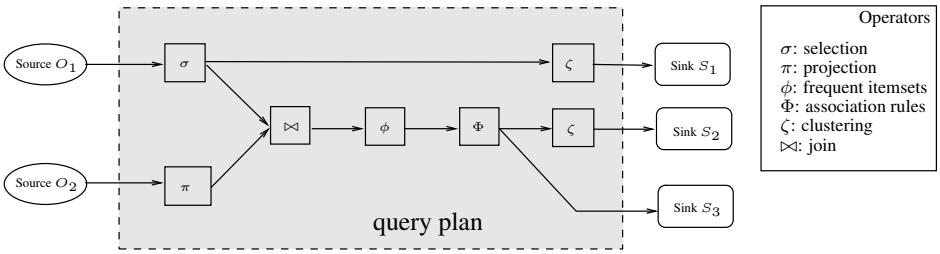


Figure 2: Example query plan

the algorithm processes an infinite series of transactions. Thus, for certain values of ε and b and specific characteristics of the underlying data stream we know the maximum number of nodes in a pattern tree. In general, we can state that a large pattern tree leads to mining results with better quality than a small pattern tree.

The fact that the overall space requirements stabilize or grow very slowly as the stream progresses was already shown in [Giannella *et al.*, 2003a]. The authors investigated different values for σ and the average number of items per transaction. [Giannella *et al.*, 2003b] additionally showed the same effect for varying values of ε. We extended these tests to different values of the number of distinct items in the stream and the size of b. With a constant input rate the size of b affects the number of transactions a batch contains. As we will show in Sect. 5 the value of b is the only one that has very little impact, as long as the number of transactions in a time interval of size b exceeds a certain threshold (depending on ε and some data stream characteristics). This is in order to fade out the effect of temporarily frequent itemsets that have no significance for the overall mining result.

According to Sect. 3 we can determine r as: *number-of-nodes*$(\varepsilon,\, b) \,*\, [2 * \lceil log_2\,(N)\rceil + 2 + c]$, representing a heuristic approximation of the resources actually needed.

Dynamic tree size adjustment

In the following we introduce an extended approach of [Giannella *et al.*, 2003a] that can cope with limited memory and uses the available memory as effective as possible. If there is not enough memory available for finding frequent itemsets in the claimed quality, we need to dynamically adjust the size of the pattern tree according to the actual memory conditions. At first we implemented the approach in [Giannella *et al.*, 2003a] and examined its memory requirements for different parameter settings. After that, we considered a couple of possibilities to control the memory requirements of the pattern tree. Thus, we gained an extended approach that has several alternatives for controlling the tree size depending on the user's quality weighting. We introduce a memory filling factor f defined as follows: $f = \frac{\textit{actual memory usage}}{\textit{provided memory}}$. Depending on f, our approach takes action to reduce or increase the size of the pattern tree. The possibilities for manipulating the size of the tree are:

1. Adjust the value of ε while keeping σ constant, i.e., change the approximation quality Q_{Ma} of our mining results according to available memory.
2. Adjust the value of σ according to available memory while keeping ε/σ constant, i.e., keep the quality Q_{Ma} of the mining result constant but change the quality of interestingness Q_{Mi}.
3. Limit the size of each TTWT according to available memory, i.e., change the time range quality Q_{Tr}.
4. Adjust the value of b according to available memory, i.e., change the time granularity quality Q_{Tg}.

Since we can estimate the memory requirements of a pattern tree for a given set of parameters, we can also estimate the maximum number of nodes our pattern tree may not exceed in order to adhere to a certain memory limit. In each step we assume the size of a TTWT to be at its upper bound. The changes in this upper bound are estimated only at constant intervals (in our tests every 100 batches), because we want to avoid registering negligible changes of the maximum number of nodes after every batch.

Option 1: Adapting ε. A first option for manipulating the size of the pattern tree is to change the value of ε, i.e., change Q_{Ma}. Therefore, we estimate an ideal ε that results in a pattern tree with approximate as many nodes as possible, rather than taking the value of ε as an input parameter. However, the user may specify a lower bound for the value of ε, i.e., an upper bound for the quality of the mining result.

Since we are not yet able to calculate the maximum number of nodes for a given amount of memory exactly, we also cannot estimate an ideal value of ε. Our algorithm adjusts the value of ε depending on the filling factor f after every processed batch as follows:

- $f < 0.85$: Decrease ε by ten percent. Use this ε when processing the following batches.
- $0.85 \leq f \leq 1.0$: The value of ε remains fixed.
- $f > 1.0$: Increase ε by ten percent. Conduct tail pruning at the TTWTs of each node in the pattern tree and drop all nodes with empty TTWTs. Repeat these steps as long as $f > 1.0$.

Note that f can become greater than 1. This is because we assume that we have more memory available in the system than the amount we provide to the pattern tree. $f > 1$ indicates that the pattern tree consumes more memory than we granted to it and the tree will thus reduce its size.

In this approach we have to store the value of ε we used in each specific time interval, in addition to monitoring the number of transactions in each interval. If two TTWT frequency entries n_i and n_j are merged, we also have to merge the according ε values ε_i and ε_j in order to determine the achievable quality for this time period. We can average the two distinct values of ε as described by equation 1. w_i and w_j denote the sizes of time intervals t_i and t_j.

$$\hat{\varepsilon} = (\varepsilon_i w_i + \varepsilon_j w_j) \,/\, (w_i + w_j) \qquad (1)$$

It was shown in [Giannella *et al.*, 2003a] that for a fixed ε, if all itemsets whose approximate frequency is larger than $(\sigma - \varepsilon) \cdot W$ are requested, then the result will contain all frequent itemsets in the period $[t_{s'}, t_{e'}]$. Thus, in our extended approach all itemsets with approximate frequency $(\sigma - \hat{\varepsilon}) \cdot W$ are returned. The value of $\hat{\varepsilon}$ depends on the time intervals contained in $[t_{s'}, t_{e'}]$. If $[t_{s'}, t_{e'}]$ covers only time intervals where the same value of ε was used for all

batches, then $\hat{\varepsilon}$ is equal to this ε. If $[t_{s'}, t_{e'}]$ contains time intervals where the value of ε differs, the value of $\hat{\varepsilon}$ becomes:

$$\hat{\varepsilon} = \left(\sum_{i=s'}^{e'} \varepsilon_i w_i \right) / W \quad (2)$$

In summary, in our first option we control the amount of memory used by varying the value of ε and so ε/σ, which reflects the approximational quality Q_{Ma} of our mining result.

Option 2: Adapting σ. The second option for adjusting the size of the pattern tree is to alter the value of σ. As σ does not influence the size of the tree directly, ε/σ remains the same. That is why the user does not provide a fixed value of σ, but rather claims for a certain ε/σ that should be guaranteed. Again, the user may also specify a lower bound for the value of σ. The handling of σ is analog to the handling of ε in the first option with the only difference, that ε must be adjusted in parallel in order to keep ε/σ constant. The analogy also applies for determining the value of $\hat{\sigma}$:

$$\hat{\sigma} = \left(\sum_{i=s'}^{e'} \sigma_i w_i \right) / W \quad (3)$$

The value of $\hat{\varepsilon}$ again results from equation 2. Returned are all itemsets whose approximate frequency is larger than $(\hat{\sigma} - \hat{\varepsilon}) \cdot W$. We guarantee to deliver all itemsets whose actual frequency is $\geq \hat{\sigma}$. Because $\hat{\varepsilon}/\hat{\sigma}$ is kept constant as requested by the user, the quality Q_{Ma} meets the user's requirements. By adjusting the value of σ we alter the quality Q_{Mi} by varying the frequency an itemset must occur in order to belong to the delivered set of results.

But, what is the difference between the first and the second option? In the first option we keep σ constant and change ε. Thus, we can request all itemsets with a minimum support of $(\sigma - \hat{\varepsilon}) \cdot W$ and accept a poorer quality Q_{Ma}. In the second option, we modify σ but keep ε/σ constant. Thus, we also keep the user defined quality Q_{Ma} constant and return all itemsets with a minimum support of $(\hat{\sigma} - \hat{\varepsilon}) \cdot W$. In this case, only the more interesting itemsets are found, in terms of σ as an interestingness measure from Q_{Mi}. An effect on memory usage is achieved in both options. Moreover, in both options the methodology of pruning TTWTs remains unchanged.

Option 3: Limiting the size of the TTWTs. The third option is to limit the size of each TTWT to a fixed value. This results in a restriction of how far we can look back into the history of the processed stream, because we limit the number of time intervals for which we store frequency information. Thus, we are not able to deliver results from a time period that includes batches lying farther back in history than the information we recorded. According to Sect. 3 this leads to an impairment of the time range quality Q_{Tr}.

As the number of time intervals stored in one entry of a TTWT increases logarithmically, saving a large amount of memory demands for limiting the size of a TTWT drastically. In this way, the information of a considerable portion of the observed batches would get lost. Lowering the maximal size by small values, only a small amount of memory can be saved.

Option 4: Adapting b. The last option in order to limit the used memory is to adjust the value of b. Assuming a constant input rate, the size of b affects the number of transactions a batch contains. Increasing b leads to an impairment of the time granularity quality Q_{Tg}, as we reduce how exact we can look back. Our experiments reveal that the number of transactions per batch does not affect the number of nodes in the pattern tree significantly. By increasing the value of b we can only reduce the total number of entries in a TTWT while processing a finite part of the stream. Considering infinite data streams, since the size of a TTWT grows logarithmically, the number of entries in a TTWT will converge to the same value for all choices of b. Therefore, this option will only have a significant impact on the required memory if it is combined with a limitation of the TTWT size. Combining both, we can reduce the granularity of a time interval and look back farther in the history of the data stream using the same number of TTWT entries.

4.3 Change Detection

In Sect. 1, we briefly discussed the importance of quickly detecting changes in data streams. In the following we will exemplarily discuss change detection on the basis of the introduced frequent itemset mining algorithm. We will outline how change detection may be implemented efficiently and, more important, resource-aware. In the next section we will investigate how our approach can be mapped to general stream mining tasks, including change detection as a post-processing step on the basis of concrete mining tasks as well as an independent mining problem. On the basis of the extended FP-Stream approach there arise several possibilities in order to implement an efficient detection of changes in the frequent itemsets themselves, their temporal occurrences and/or other relevant aspects. With several of these alternatives we even get a resource-adaptive approach for detecting changes for free. We will sketch five different approaches and briefly discuss pros and cons.

From our point of view any technique for change detection should meet the following criteria:

- it is based on data structures which can be limited in size, preferably in a dynamic manner
- it produces realtime results, which requires an online and incremental processing
- it is capable of detecting any kind of changes that may be of interest (i.e., frequency changes, correlation changes, temporal changes, and so on) and is not restricted to the detection of specific kinds
- changes can be represented to the user in an intuitive and understandable manner

With the problem of frequent itemset mining, a lot of *sophisticated* changes in the stream (e.g., if subsets of a formerly frequent itemset are still frequent) can be detected by analyzing *basic* changes (i.e., the occurrence/omission of single members of the set of all frequent itemsets). Therefore, for now we primarily focus on detecting these basic changes.

Approaches on Separate Data Structures

A naive but simple solution is to take the output of the frequent itemset mining operator as an independent input stream for a separate change detection operator. Like this, information about the itemsets found in the past must be stored in some separate data structure. We investigated three different approaches for that:

1. store all frequent itemsets produced so far together with additional information (e.g., temporal) in a table

2. store snapshots of the pattern tree in regular intervals
3. store only differences between the set of all frequent itemsets in regular intervals (similar to the incremental backup technique for database systems)

A main problem of all three approaches is the need for a separate data structure. This data structure allocates extra resources, and thus, resource limits must be shared between the actual mining operator and the change detection operator. However, the resource-adaptive techniques introduced in this work could be applied to this data structure in order to meet given resource limits. The first two methods allow for detecting a wide variety of changes, even temporal ones, but are consuming by far too much memory. With the first variant, the task of detecting changes in one specific itemset is complicated, because there is no information about the location of itemsets in the pattern tree if they are registered after being output from the frequent itemset operator. The last of the three methods is suitable for quickly detecting new or omitted itemsets between two successive time steps – but complicates the handling of arbitrary time intervals and the detection of changes in the frequency of itemsets that have already been frequent before.

Approaches on the Pattern Tree

In order to implement a resource-efficient change detection, a more intuitive approach is to try to detect changes directly from the pattern tree used in the frequent itemset mining operator. This would need no extra memory, which is one of the main resources in our considerations. Moreover, because change detection and itemset mining is combined into one operator, realtime signalizing is easy to achieve. Again, we distinguish two specific approaches:

1. detect changes as soon as the pattern tree is modified
2. detect changes after executing the FP-growth algorithm on the pattern tree (when looking for the actual frequent itemsets after each batch)

The second approach is capable of detecting more changes, because with the first one we can only watch itemsets that are modified in the current batch run. If a change in an itemset is only recognized during the run of FP-growth, the first method will not detect this change. The disadvantage of the second method is that it can only be run after processing a whole batch and has to completely traverse the tree – which leads to worse reaction time and less information about new or deleted nodes. However, in the first method we have to deal with *change candidates*, because not each modification will lead to a actual change in the itemsets. Advantages of both approaches, in contrast to those on separate data structures, are low runtime, easy detection of temporal changes (the TTWTs containing temporal information can be analyzed directly) and no extra memory consumption.

Usually, the reaction time Q_{T_c} is the most important quality measure for change detection in data streams. The approaches working on the pattern tree, particularly the first one, can provide better values for Q_{T_c} than those on separate data structures, because it does not depend on the output interval of the mining operator or a predefined interval. Rather, both techniques can be integrated right into the frequent itemset mining operator.

In this section, we briefly presented preliminary results we achieved when investigating approaches for resource-aware change detection. For now, we state that the choice of algorithm depends on the goals actually desired by the user. Under special circumstances several of the introduced methods should be combined – when aiming for a resource-aware, and moreover, resource-efficient, method, those based on the pattern tree should be preferred. In future works we will investigate approaches for change detection more detailed, including the combination with resource-adaptive techniques, considerations about achievable qualities (mainly for Q_{T_c}, but also other measures like Q_{T_g} and Q_{M_a} have to be considered) as well as change detection methods for general mining tasks.

4.4 General Applicability of the Approach

We are currently generalizing our approach of resource-adaptive frequent itemset mining. Our work aims at making the technique applicable to data stream mining algorithms with certain characteristics in general.

To start with, there exist other algorithms for mining frequent itemsets in data streams that use the same approach of approximated frequencies as the FP-Stream algorithm, e.g., [Manku and Motwani, 2002; Chang and Lee, 2004]. We can thus extend these algorithms in a way analog to the modification of FP-Stream. In general, each algorithm whose resource requirements depend on parameters like σ and ϵ can be modified like that – with different impact on the grade of adaptiveness.

Some of the presented methods to manipulate the size of the pattern tree can be applied to these algorithms without major changes to both method and algorithm. Despite the usage of different data structures in these algorithms, it is possible to adapt the size of their synopsis by varying the values of σ and ϵ. The quality of the mining results will remain a computable value that can be guaranteed like in the extended FP-Stream algorithm.

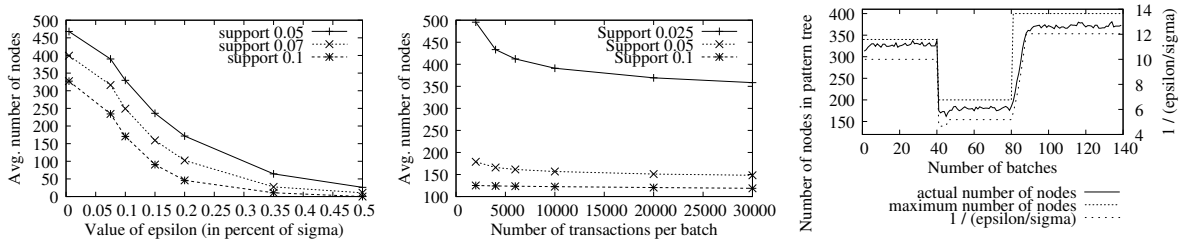
A majority of data stream mining algorithms uses synopsis data structures to capture the content of the data stream. Since these synopses only represent an approximation of the stream's actual content, there is always the notion of quality associated with it. Our method can thus be applied in principle to such stream mining algorithms as well. In [Franke *et al.*, 2005] we already successfully applied the approach in order to implement a resource-aware clustering operator for PIPES.

5 Evaluation

The purpose of the following evaluation is twofold: at first we will show how we achieve the aimed quality awareness. To show this, we will evaluate how good the algorithms achieve a claimed quality and how exact the corresponding calculations are. In the context of the proposed stream mining approach quality awareness comes along with the adaption to provided resources and the determination of needed resources concerning aimed quality measures. This is the second objective of this section: we will show that memory requirements are approximated satisfyingly and that they are met finally, and which conclusions we can draw to the achieved quality.

Before evaluating our quality-aware frequent itemset mining approach we conducted a series of tests with the original FP-Stream algorithm. We figured out how the algorithm behaves for different parameter settings of ε, σ and b. We used synthetic data generated by the IBM market-basket data generator. In the first set of experiments 1M transactions were generated using 1K distinct items and an average transaction length of 3. All other parameters were set to their default values from the generator.

Since in the original approach a batch does not cover a constant period of time but a constant number of transac-



(a) Varying support and epsilon (b) Varying support and batch size (c) Resource awareness

Figure 3: Average number of nodes in a pattern tree

tions, we set the size of a batch to 5000 transactions. Figure 3(a) shows some of our experimental results with the original algorithm. We measured the average number of nodes in a pattern tree for three different values of σ. For each σ we run the algorithm with various ratios between ε and σ to simulate different quality demands. As expected the number of nodes decreases with rising ε and σ.

Figure 3(b) shows results of another series of tests we conducted with the original algorithm. We processed the test data with different values of σ (always setting $\varepsilon = 0.1 \cdot \sigma$) and with various batch sizes. The number of nodes in the pattern tree does not decrease significantly when the batch size is highly raised.

We also processed test data having more average items per transactions (5, 10) and/or more distinct items (5K, 10K) and we additionally tried different batch sizes (1000 to 30000). The main conclusion is always as stated above. One remarkable thing we noticed is that for small batch sizes the number of nodes in a pattern tree is far from being constant. For example, when processing testing data with 1M transactions, 1K distinct items and average transaction length of 3 we set $\sigma = 0.025$, $\varepsilon = 0.1 \cdot \sigma$ and the batch size to 1000 transactions. The number of nodes in the resulting pattern tree oscillated between under 1000 to nearly 5000. When processing the same data set with higher values of σ, ε or batch size this range got significantly smaller. For a support of 0.07 the difference between the minimum and the maximum value of the number of nodes was 45. Remainders of this effect can be seen in Figure 3(b) for $\sigma = 0.025$ by the sharp decrease of the number of nodes for low batch sizes.

For evaluating the quality-aware frequent itemset mining approach, we again generated data sets with 1M transactions using 1K distinct items and an average transaction length of 3. Our algorithm consumed the stream of transactions from a source with a constant output rate of 3 seconds. The value b of the finest granularity of time was set to 15000 seconds, so each processed batch contained 5000 transactions. The value of σ was set to 0.05.

Firstly, we wanted to demonstrate that our approach can cope with changing memory conditions. Instead of limiting the actual amount of available memory we limited the number of nodes that the pattern tree may have. One could calculate the actual amount of memory needed, since the maximum size of a TTWT can be estimated as described in Section 4.2.

Initially we limited the maximum number of nodes the generated pattern tree may have to 340. As we do not have a formula yielding the maximum number of nodes in a pattern tree for a given set of parameters, we had to choose an adequate value of ε for the algorithm. Our previous experiments showed, that $\varepsilon = 0.1 \cdot \sigma = 0.005$ would be a good value to start.

We started the algorithm and decreased the maximum number of nodes after 40 batches to 200 nodes. We then raised it again after 40 batches to a value of 400 nodes. Figure 3(c) shows the algorithm's behavior. After 40 batches, it had to raise the value of ε several times to get its filling factor below 1. Then, after the next 40 batches, the algorithm lowered the value of ε after every processed batch until a tree size was reached that was close enough to the maximum according to the filling factor. Figure 3(c) also displays the quality of the stream mining for every batch, i.e., the value of $1/(\varepsilon/\sigma)$.

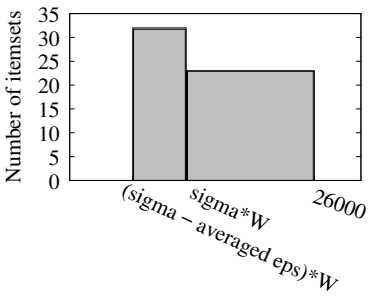


Figure 4: Quality awareness

With our second series of tests of our quality-aware approach we demonstrate the quality of our mining results. We started the algorithm using option 1 of our proposed techniques for adjusting the size of the pattern tree, i.e., we change the value of ε when necessary. The experimental settings are equal to these of the above test. Only this time we set $\sigma = 0.025$ and $\varepsilon = 0.2 \cdot \sigma$ to get any frequent itemsets at all. The experiments show that we estimated the overall quality of our mining results ranging over several batches (and thus several values of ε) correctly and that it is in fact suitable to average distinct values of ε as described.

First we run our approach once to get the value of $\hat{\varepsilon}$, which was approximately 0.00765. Then we used the original FP tree algorithm to get the actual set of frequent itemsets from our test data. We also got the actual frequencies of all itemsets having frequencies between $(\sigma - \hat{\varepsilon})$ and a little less and the required support. Then, we ran our quality-aware approach and asked for the set of frequent itemsets after the 120th batch. We set the time period $[t_s, t_e]$ to the whole period of time the stream was processed so far. So the window we requested contained $W = 600000$ transactions. After we received the result, we compared the output to what we knew from the FP-growth method. For each itemset our approach delivered, we looked up its real frequency and printed these information in the histogram shown by Figure 4. All itemsets our approach output were inserted in frequency buckets, according to their real frequency which we gained through using the FP tree algorithm.

Figure 4 shows that the output contained no itemsets having frequency less than $(\sigma - \hat{\varepsilon}) \cdot W$. The histogram also shows that exactly 23 itemsets having frequency of at least σ were delivered. These itemsets are the same as FP-growth delivered.

All in all the experiments met our expectations. In [Franke *et al.*, 2005] we also conducted similar experi-

ments for a resource-aware clustering strategy, which emphasize the results showed here. The approximations of memory usage hold, the quality deduced from the available resources is close to the actually achieved quality. This is true for all examined quality measures, as far as our tests can show that. Of course, we have to do a couple of extended test series, including different parameter settings, varying stream characteristics and a deeper analysis of several (classes of) quality measures. With the results of these subsequent tests we could finally demonstrate the quality and resource awareness already achieved in this work.

6 Conclusion

In this paper, we argued that quality of analysis results is an important issue of mining in data streams. The reason is that stream mining can be usually performed only on a resource-limited subset or approximation of the entire stream, which affects different measures of data quality. Based on specific quality measures for stream mining, we investigated and enhanced a frequent itemset mining technique in order to estimate the quality depending on the current resource situation (mainly the available memory) as well as to allocate resources needed for guaranteeing user-specified quality requirements. Furthermore, we gave directions for making a whole class of stream mining algorithms resource- and quality-aware, including complementary tasks such as application specific change detection.

Beside these goals and the mentioned considerations about resource-aware approaches for change detection in data streams we will also apply the introduced techniques to other mining primitives. Based on the earned experiences about the method's practical applicability and the reflections on different quality measures we plan to build a formal framework for resource-adaptive stream mining under quality guarantees. This includes the definition of the concrete quality/resource functions r and $\overline{r}$.

From our point of view, other resource requirements than memory consumption of stream synopses have to be regarded as well. For instance, beside the memory required by the pattern tree itself, the algorithm needs to have additional memory available during runtime. This memory is used for the actual computations and for storing two batches, the one that is currently processed as well as the one that is currently build from the newly arriving transactions. When considering the computations done by the FP-Stream algorithm, we note that the FP-growth method used to determine the batch's frequent itemsets is very expensive in terms of memory requirements. Using a more memory efficient FP-growth method as proposed and implemented by Özkural and Aykanat [Özkural and Aykanat, 2004] would lead to decreased overall memory requirements of the algorithm.

Moreover, in addition to memory awareness we will take the algorithms' runtime into account. In this context several additional aspects have to be considered, like the streaming rate of the incoming data and the runtime overhead imposed by our extension. In addition, we will have to deal with the conflict that adding resource adaptiveness to an algorithm imposes a runtime and memory overhead itself.

Finally, we have addressed only operator-locally parameters like the amount of memory available for this specific operator. In future work, we plan to take also global properties of the whole analysis pipeline into account, e.g. load shedding and windowing operators, which have an impact on the output quality, too.

References

[Berti-Équille, 2006] L. Berti-Équille. Contributions to Quality-Aware Online Query Processing. *IEEE Data Eng. Bull.*, 29(2):32–42, 2006.

[Chang and Lee, 2004] J. H. Chang and W. S. Lee. A Sliding Window Method for Finding Recently Frequent Itemsets over Online Data Streams. *Journal of Information Science and Engineering*, 20(4):753–762, 2004.

[d. Bercken *et al.*, 2001] J. V. d. Bercken, B. Blohsfeld, J.-P. Dittrich, J. Krämer, T. Schäfer, M. Schneider, and B. Seeger. XXL - A Library Approach to Supporting Efficient Implementations of Advanced Database Queries. In *VLDB 2001*, pages 39–48, 2001.

[Franke *et al.*, 2005] C. Franke, M. Hartung, M. Karnstedt, and K. Sattler. Quality-Aware Mining of Data Streams. In *IQ*, 2005.

[Gaber and Yu, 2006] M. M. Gaber and Ph. S. Yu. A framework for resource-aware knowledge discovery in data streams: a holistic approach with its application to clustering. In *SAC'06*, pages 649–656, 2006.

[Gaber *et al.*, 2003] M. M. Gaber, Sh. Krishnaswamy, and A. Zaslavsky. Adaptive mining techniques for data streams using algorithm output granularity. In *The Australasian Data Mining Workshop*, 2003.

[Giannella *et al.*, 2003a] C. Giannella, J. Han, J. Pei, X. Yan, and P. S. Yu. Mining Frequent Patterns in Data Streams at Multiple Time Granularities. In *Workshop on Next Generation Data Mining*, 2003.

[Giannella *et al.*, 2003b] C. Giannella, J. Han, E. Robertson, and C. Liu. Mining Frequent Itemsets over Arbitrary Time Intervals in Data Streams. Technical report, Indiana University, 2003.

[Han *et al.*, 2000] J. Han, J. Pei, and Y. Yin. Mining Frequent Patterns without Candidate Generation. In *SIGMOD 2000, Dallas, USA*, pages 1–12, 2000.

[Krämer and Seeger, 2004] J. Krämer and B. Seeger. PIPES - A Public Infrastructure for Processing and Exploring Streams. In *SIGMOD 2004*, pages 925–926, 2004.

[Manku and Motwani, 2002] G. S. Manku and R. Motwani. Approximate Frequency Counts over Data Streams. In *VLDB 2002, Hong Kong, China*, pages 346–357, 2002.

[Özkural and Aykanat, 2004] E. Özkural and C. Aykanat. A Space Optimization for FP-Growth. In *ICDM Workshop on Frequent Itemset Mining Implementations*, 2004.

[Tan and Kumar, 2000] P. Tan and V. Kumar. Interestingness Measures for Association Patterns: A Perspective. Technical report, University of Minnesota, 2000.

[Teng *et al.*, 2004] W.-G. Teng, M.-S. Chen, and Ph. S. Yu. Resource-aware mining with variable granularities in data streams. In *SDM 2004*, 2004.

[Toivonen *et al.*, 1995] H. Toivonen, M. Klemettinen, P. Ronkainen, K. Haton, and H. Mannila. Pruning and grouping discovered association rules. In *ECML Workshop on Statistics, Machine Learning and Knowledge Discovery in Databases*, pages 47–52, 1995.

Automated Model Selection with AMSF in a production process of the automotive industry

Florian Grewe and **Peter Owotoki**
Hamburg University of Technology (TUHH)
21077 Hamburg, Germany
florian.grewe|owotoki@tu-harburg.de

Abstract

Machine learning, statistics and knowledge engineering provide a broad variety of supervised learning algorithms for classification. In this paper we introduce the *Automated Model Selection Framework* (AMSF) which presents automatic and semi-automatic methods to select classifiers. To achieve this we split up the selection process into three distinct phases. Two of those select algorithms by static rules which are derived from a manually created knowledgebase. At this stage of AMSF the user can choose between different rankers in the third phase. Currently, we use instance based learning and a scoring scheme for ranking the classifiers. After evaluation of different rankers we will recommend the most successful to the user by default. Besides describing the architecture and design issues, we additionally point out the versatile ways AMSF is applied in a production process of the automotive industry.

1 Introduction

Following CRISP, the process of Data-Mining can be broken down into the phases of *business understanding*, *data understanding*, *data preparation*, *modeling*, *evaluation* and *deployment* [CRI, 2000]. In the modeling phase algorithms are selected and applied to generate models. The issue of selecting appropriate methods in this phase we refer to as *Model Selection*. CRISP describes algorithm selection as an exploratory process, highly dependant on the analyst's knowledge and on the problem domain.

In the field of *Knowledge Discovery in Databases* different projects with different approaches exist dealing with the issue of Model Selection. The most recent project is MetaL, combining the results of previous research projects in an online service [Met, 2004]. Users can upload their datasets and receive a ranking of algorithms.

The *Waikato Environment for Knowledge Analysis* (WEKA) is a tool for data analysis and includes implementations of different classification algorithms. A book describing the software was published in 2005 by Ian H. Witten and Eibe Frank [Witten and Frank, 2005]. WEKA's binaries and sources are freely available. Implemented methods include instance-based learning algorithms, statistical learning like Bayes methods and tree-like algorithms like ID3 and J4.8 (slightly modified C4.5). Including combinations of classifiers, e.g. bagging and boosting schemes, there are over sixty methods available in WEKA. Table 1 shows a list of all non-combined methods with a short description. All algorithms can be utilized for supervised learning with their standard parameters.

This paper introduces AMSF which's intention is to provide help to human users by semiautomatically selecting appropriate methods. Furthermore its selection schemes can be utilized to automatically analyze problem domains by analyzing datasets and ranking methods. For the semi-automatic use, support is provided by a user interface in form of a wizard[1].

2 AMSF Components

This Section introduces the different components AMSF consists of. From an architectural point of view the components and classes are separated according to a three layer architecture. In the data-layer we put two database files

- the *Knowledgebase of Classifiers* and
- the *Performance Database*.

In the logic layer there are the classes to handle the selection and ranking of the algorithms. Important components here are

- the *Preselection Component* and
- the *Ranking Component*.

In the presentation layer we find the wizard and other GUI-components.

2.1 Knowledgebase of Classifiers

The Knowledgebase of Classifiers is a database and contains an entry for every algorithm WEKA 4.5 provides.

The entries contain information about the applicability of an algorithm on a dataset. Here, we mean wether an algorithm can handle missing values or unlabeled instances. Furthermore the algorithms vary in their ability of dealing with different types of attributes and classes. Following the conventions of the Attribute-Relation File Format (ARFF) each attribute may be either numeric or nominal.

In addition to applicability characteristics the algorithms in the Knowledgebase of Classifiers are categorized according to the type of knowledge representation they generate. J4.8 for example generates a decision tree. Figure 1 exemplarily shows the Knowledgebase of Classifiers' entry for J4.8.

J. Gama and P. Brazdil provide additional criteria in [Gama and Brazdil, 1995] to characterize classifiers. In contrast to the Performance Database described in Section 2.2 the Knowledgebase of Classifiers was created manually by considering relevant literature and the sources of WEKA.

[1] A wizard is a user friendly stepwise dialog

```
<Method name="J48">
    <Characterization>
        <Group>
            <GroupEntry>trees</GroupEntry>
        </Group>
        <ModelInterpretability>interpretable</ModelInterpretability>
        <KnowledgeRepresentation>
            <KnowledgeRepresentationEntry>Decision Tree</KnowledgeRepresentationEntry>
        </KnowledgeRepresentation>
        <Input>
            <NumericalValues value="true"/>
            <NominalValues value="true"/>
            <MissingValues value="true"/>
        </Input>
        <Output>
            <NumericalValues value="false"/>
            <NominalValues value="true"/>
            <BinaryClass value="true"/>
            <MissingValues value="true"/>
        </Output>
    </Characterization>
    <Information>
        <Source>WekaAPI</Source>
        <Text>Class for generating an unpruned or a pruned C4.5 decision tree. For more
            information, see Ross Quinlan (1993). C4.5: Programs for Machine
            Learning, Morgan Kaufmann Publishers, San Mateo, CA. </Text>
    </Information>
</Method>
```

Figure 1: J4.8 entry in the Knowledgebase of Classifiers

2.2 Performance Database

In addition to the described Knowledgebase of Classifiers, we generated a Performance Database in an automated creation process. Details about the creation process are outlined in Section 2.3.

The Performance Database comprises information about the method's error rate, training and testing time when applied to a particular dataset. It currently contains about two-hundred entries, one for each dataset. To summarize, each entry of the Performance Database contains

- dataset characterization measures described in Section 2.4 and a
- list of performance related data consisting of one entry per applied learning algorithm.

The Performance Database and the Knowledgebase of Classifiers described in Section 2.1 are stored in XML-format and can be validated against schema documents after new entries have been added.

2.3 Performance Database Creator

The Performance Database contains empirical data. It was generated by applying the algorithms provided by WEKA in their standard setup. Here, for every dataset we utilized the Knowledgebase of Classifiers to select a subset of methods applicable to the dataset. The methods were applied using stratified ten-fold cross-validation. The resulting Error rates, training and testing times were averaged according to the cross-validation settings.

We expect the performance of AMSF to increase when additional entries are added to the Performance Database. AMSF therefore includes a graphical component to add new entries comfortably.

2.4 Dataset Characteristics

Besides error rate and time dependent information, we store information about the datasets themselves in the Performance Database. The coverage of characterization information stored follows the proposals of R. Engels and C. Theusinger in [Engels and Theusinger, 1998]. Referring to their terminology, the characteristics include

- *simple characteristics* like number of cases, number of defective cases and number of binary attributes,
- *statistical measures* like median and median deviation of classes and
- *information theoretical measures* like class entropy.

2.5 User Interface

The main GUI component of the AMSF so far is the wizard already mentioned in Section 1. Figure 2 shows a screenshot of one step in the model selection process. Besides the wizard AMSF also provides an user interface for controlling the creation and extension of the performance database. We furthermore developed a GUI for the preselection process described in Section 3.1 to enable fast and direct access to the information of the Knowledgebase of Classifiers.

3 Selection Process

In this Section we explain how the components described in Section 2 are utilized to perform the selection and ranking of the classifiers.

3.1 Preselection Component

Fed by the Knowledgebase of Classifiers the Preselection Component selects the methods applicable to a given domain description. Here, the domain description includes the user preferences regarding the method's knowledge representation and furthermore the characterization of the input attributes and the class attribute as described in Section 2.1. The preselection is carried out by simply suppressing the algorithms which cannot handle the properties

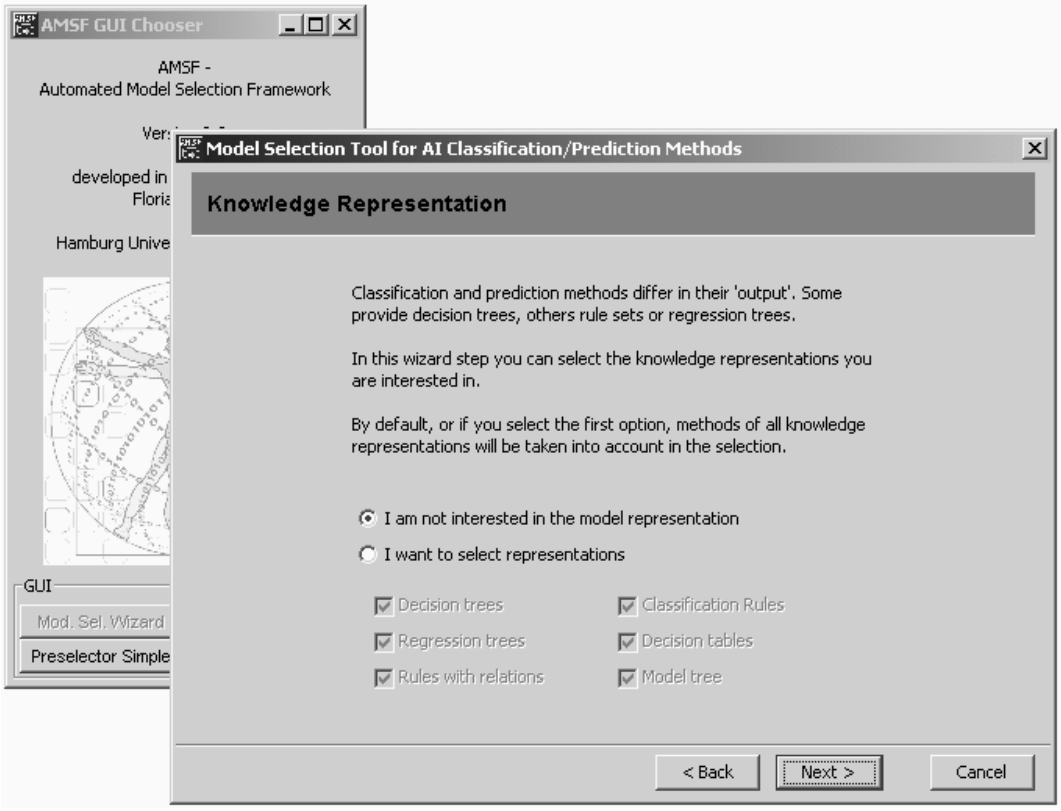


Figure 2: The third wizard step of the algorithm selection

of the dataset. Their properties can be comfortably calculated directly whereas those have to be provided in ARFF-format. Some of the input and class attribute information is included in the ARFF header and does not have to be derived from the data itself. It is therefore important that the ARFF-file was created according to the specification.

3.2 Ranking Component

Currently, AMSF includes two rankers. The first ranker utilizes an instance-based learner to find the k nearest neighbors in the dataset characterization space introduced in Section 2.4.

To find the nearest neighbors, the user has to provide a dataset that the characteristics can be calculated of. In the current version we have not yet implemented any special weighting of the dimensions. After the k nearest neighbors have been identified, only the error rate information is considered to perform the ranking by now.

For this purpose we created a scoring scheme: Within every of the k nearest neighbors the classifiers are sorted according to error rate. The best algorithm obtains three points, the second best two and the third best one point. The points are then accumulated over the k datasets for each method and then ranked according to their score. This is particulary useful in this stage of the software, as we presume the methods to become more obvious winners when we adjust the weighting of the dataset characteristics dimensions.

The second ranker is applicable without providing a concrete dataset and only ranks the methods by the standardized error rate sum

$$err_s = \frac{1}{N} \sum_{i=1}^{N} err_i \qquad (1)$$

where N denotes the number of datasets the algorithm was applied to and err_i is the error rate of the method applied to the i-th dataset. The only information the user has to provide besides the preselection criteria is the nature of the class attribute. The methods are then either ranked according to Equation 1 or according to the corresponding equation of the sum of squares.

The whole selection procedure is illustrated in Figure 3 and summarized in the caption.

4 AMSF in the real world

Data Mining methods can be utilized to uncover new valuable insights. Still, one major problem when those methods are to be applied is insufficient usability. Although tools like WEKA offer a comprehensive spectrum of analysis alternatives, the appropriate selection of applicable methods is still an issue of the user.

In modern production lines hundreds to thousands of parameters can be set and adjusted not to mention the high number of measurement values that are recorded during production. Analyzing and interpreting these data can be of high benefit with respect to quality and cost optimization.

In a production process of the DaimlerChrysler plant in Hamburg we therefore set up an *Integrated Database* which collects data of the complete production process including quality data about input factors and former auxiliary conditions like data about hall temperature and humidity. In doing so, it is possible to identify the produced parts throughout the production process and mine valuable knowledge.

In order to increase quality characteristics or to determine the process robustness the parameters are often varied systematically. The methodology of *Design of Experiments*

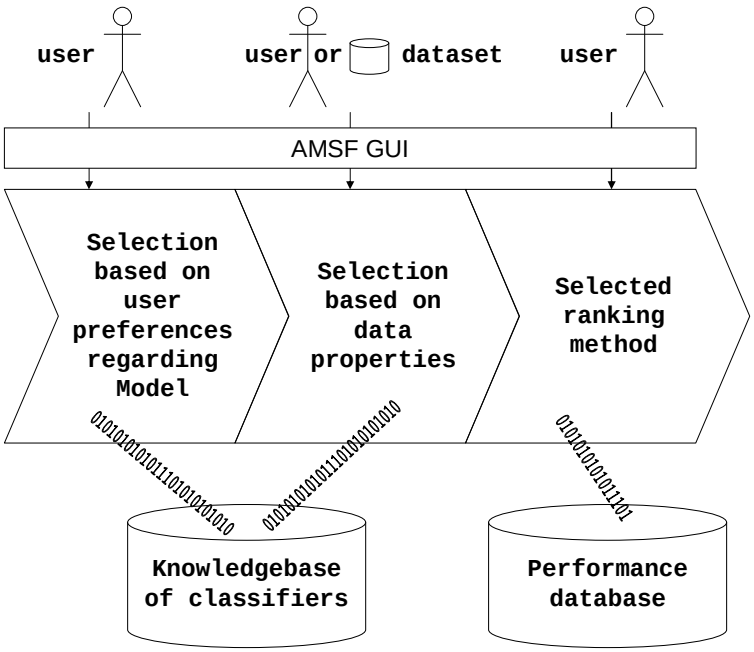


Figure 3: The selection process split up in the three distinct phases *user preferences*, *data properties* and *ranking*. The user employs the wizard to control the process and analyze the optionally provided dataset. The third phase utilizes the performance database. The previous phases generate rules from the Knowledgebase of Classifiers.

(DOE) provides procedures to maximize the knowledge that can be gained with a fixed preferably small number of experiments. Classically, the input as well as the output parameters of such experiments are assumed to be numerical. After the experiments have been carried out linear models are adjusted by the method of linear regression. In doing so only a small number of parameters, namely the varied input parameters and the output parameter, are utilized to build the models. So, almost all the data collected during the experiments is not considered. This is especially unsatisfactory if the created linear models do not sufficiently fulfill the necessary requirements and are discarded after the procedure for that reason.

By mining the data and carefully looking at the created interpretable models (in this case the models have to be 'interpretable', as described above), we gained valuable new insights in the production process. The dependencies revealed are used for example to classify produced parts avoiding expensive experiments.

5 Evaluation of preselection and rankers

As mentioned in Section 2.2 we created performance data for about two-hundred datasets. In that process, the preselection component described in Section 3.1 was utilized and tested. It attracted attention that a relatively large ration of classifiers could be applied to the datasets (approximately 70% in average). We used UCI and our own datasets to calculate the performance data and are currently in the process of producing more data.

A comparison of rankers provided by different ranking methods of AMSF and ideal ranking (calculated later on) is not yet available. Anyhow, the described ranker already produces reasonable results although the possibilities are not yet exploited.

6 Conclusion and Future Work

AMSF provides user support in selecting appropriate methods. To reach this goal it compares the domain-description provided by the user and the dataset on one hand with certain databases on the other hand.

AMSF provides assistance to the user in different ways:

- It automatically analyzes datasets.
- In a preselection step it identifies applicable methods.
- It provides a ranking of the applicable by different ranking methods.

Considering the ranking component described in Section 3.2 other approaches, e.g. the *Zoomed Ranking* method of P. Brazdil and C. Soares described in [Soares and Brazdil, 2000] also take time issues into account. This is particularly valuable if the user has fixed notions about the runtime, for example if a method for real-time systems has to be chosen.

Acknowledgments

We want to thank Dr. Oliver Kropla, Eckhard Reese, Michael Picker, Matthias Plagmann, Jan Zimmermann, Klaus Vukelic, Jürgen Westphal, Henning Bendfeldt and Heiko Inert of the DaimlerChrysler AG for the valuable discussions and for providing the data.

References

[CRI, 2000] Cross industry standard process for data mining (crisp-dm 1.0), www.crisp-dm.org, 2000.

[Engels and Theusinger, 1998] Robert Engels and C. Theusinger. Using a data metric for preprocessing advice for data mining applications. In *European Conference on Artificial Intelligence*, pages 430–434, 1998.

Name	Description
AODE	Averaged, one-dependence estimators
BayesNet	Learn Bayesian nets
Complement NaiveBayes	Build a Complement Nave Bayes classifier
NaiveBayes	Standard probabilistic Nave Bayes classifier
NaiveBayes Multinomial	Multinomial version of Nave Bayes
NaiveBayes Simple	Simple implementation of Nave Bayes
NaiveBayes Updateable	Incremental Nave Bayes classifier that learns one instance at a time
ADTree	Build alternating decision trees
DecisionStump	Build one-level decision trees Id3 Basic divide-and-conquer decision tree algorithm
J48	C4.5 decision tree learner (implements C4.5 revision 8)
LMT	Build logistic model trees M5P M5 model tree learner
NBTree	Build a decision tree with Nave Bayes classifiers at the leaves
RandomForest	Construct random forests
RandomTree	Construct a tree that considers a given number of random features at each node
REPTree	Fast tree learner that uses reduced-error pruning
UserClassifier	Allow users to build their own decision tree
ConjunctiveRule	Simple conjunctive rule learner
DecisionTable	Build a simple decision table majority classifier
JRip	RIPPER algorithm for fast, effective rule induction
M5Rules	Obtain rules from model trees built using M5
Nnge	Nearest-neighbor method of generating rules using nonnested generalized exemplars
OneR	1R classifier Part Obtain rules from partial decision trees built using J4.8
Prism	Simple covering algorithm for rules
Ridor	Ripple-down rule learner
ZeroR	Predict the majority class (if nominal) or the average value (if numeric)
LeastMedSq	Robust regression using the median rather than the mean
LinearRegress.	Standard linear regression
Logistic	Build linear logistic regression models
Multilayer Perceptron	Backpropagation neural network
PaceRegression	Build linear regression models using Pace regression
RBFNetwork	Implements a radial basis function network
SimpleLinear Regression	Learn a linear regression model based on a single attribute
SimpleLogistic	Build linear logistic regression models with built-in attribute selection
SMO	Sequential minimal optimization algorithm for support vector classification

Table 1: Non-combined methods in the knowledgebase [Witten and Frank, 2005]. The first block shows the group of Bayes methods, the second tree algorithms, the third instance-based learners and the last the functions group.

[Gama and Brazdil, 1995] J. Gama and P. Brazdil. Characterization of classification algorithms. *Progress in Artificial Intelligence, 7th Portuguese Conference on Artificial Intelligence, EPIA-95*, pages 189–200, 1995.

[Met, 2004] Metal, www.metal-kdd.org, 2004.

[Soares and Brazdil, 2000] Carlos Soares and Pavel Brazdil. Zoomed ranking: Selection of classification algorithms based on relevant performance information. pages 126–135, 2000.

[Witten and Frank, 2005] Ian H. Witten and Eibe Frank. *Data Mining: practical machine learning tools and techniques*. Morgan Kaufmann, 2005.

Two-Phase Clustering Strategy for Gene Expression Data Sets

Dirk Habich, Thomas Wächter, and Wolfgang Lehner
Dresden University of Technology
Database Technology Group
dbgroup@mail.inf.tu-dresden.de

Christian Pilarsky
Dresden University of Technology
Visceral Thorax Surgery Group
christian.pilarsky@mailbox.tu-dresden.de

Abstract

In the context of genome research, the method of gene expression analysis has been used for several years. Related microarray experiments are conducted all over the world, and consequently, a vast amount of microarray data sets are produced. Having access to this variety of repositories, researchers would like to incorporate this data in their analyses to increase the statistical significance of their results. In this paper, we present a new two-phase clustering strategy which is based on the combination of local clustering results to obtain a global clustering. The advantage of such a technique is that each microarray data set can be normalized and clustered separately. The set of different relevant local clustering results is then used to calculate the global clustering result. Furthermore, we present an approach based on technical as well as biological quality measures to determine weighting factors for quantifying the local results proportion within the global result. The better the attested quality of the local results, the stronger their impact on the global result.

1 Introduction

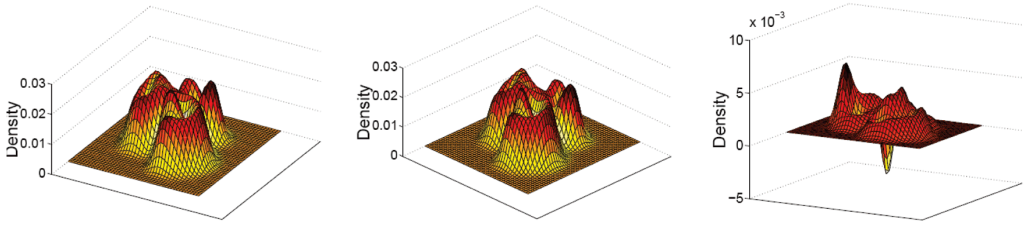
Deoxyribonucleic acid (DNA) microarrays are an important part of a new and promising field of biotechnology. They allow the simultaneous measurement of expression values in cells for thousands of genes. Microarray experiments are increasingly popular in biological as well as medical research to address a wide range of problems. One prominent example is cancer research, where microarrays are used to study the molecular variations among tumors with the aim of developing better diagnostics and treatment strategies. Within the last few years, a vast amount of microarray data sets has been produced for various studies worldwide and made commonly available in public data repositories. Having access to this variety of repositories, researchers would like to incorporate this data sets in their analyses to increase the statistical significance of their results.

The classic gene expression analysis process consists of four steps: *data integration*, *data normalization*, *data analysis* and *interpretation*. The order of the four steps is typically fixed; however, the algorithms within every step are very flexible and not standardized. In this workflow, many different tools are currently involved, which are working in an independent and incompatible way. A first challenge in the *data integration* step is to locate relevant data sets, to download them, and finally, to build up an integrated data base. This step usually requires a lot of time. Due to variations in the experimental conditions and the quality of the biological material, the measurements are not directly comparable and appropriate normalization has to be applied. As the chosen normalization has a strong influence on the analysis results [7], it is desirable to adjust the normalization method according to a data set's characteristics and separate for all the respective data sets.

By following the classic approach for meta-analysis, a combined global data set is normalized to achieve comparability of independently measured expression levels leading to a heterogeneous view of the data, that is not necessarily corresponding to the biological truth. Furthermore, it can be said that data it analyzed in a conjoint manner, which underlies a huge experimental variance. The last step of the classic analysis approach is the analysis of a global normalized data set. As an inherently data-driven technique, clustering can determine the statistical characterization of unknown data distribution, whereas clustering results depend on the underlying data characteristics as well as the initial parameters of the algorithm. Therefore, different clustering results can be produced from one single data set. It is also desirable to adjust the clustering method as well as the normalization according to each single microarray data set and to compute a global result afterwards.

We suggest that each microarray data set should be normalized and clustered separately (first phase) and that the combination of the local clustering results to a global result (second phase) yields better results than the classical meta-analysis. Summarizing the advantages of our two-phase clustering strategy in comparison to the classic approach, it can be said:

- The new approach possibly leads to better results by evaluating single homogeneous microarray data sets instead of just one fully integrated data base, because normalization and clustering can be adjusted for each data set separately.
- The global result is calculated using a set of different local clustering results. In this case, we integrate results instead of data and then analyze the integrated data.
- For every local result, a statistical weighting factor can be determined for quantifying the local results proportion within a global result based on technical and biological quality measures. The better the attested quality of the local results, the stronger their impact on the global result.



(a) $\hat{f}^D$, $n = 200$ (b) $\hat{f}^P$, $k = 50$ (c) $\hat{f}^P - \hat{f}^D \leq 10\%$

Figure 1: Comparison of the kernel density function with an approximation, which is based on k centroids ($k = 50$) for a $2D$ Example.

The remainder of the paper is organized as follows: In section 2, we give an overview of analysis methods for microarray data sets as well as of other related approaches. In section 3, some preliminary work is presented. Our two-phase clustering strategy is then described in section 4. The evaluation is done in section 5. Finally, in section 6, we conclude the paper.

2 Related Work

An overview of state-of-the-art approaches to cluster microarray data sets is given by Chipman et al. in [3]. Using clustering methods, it is possible to identify groups of similar samples (genes). The basis for this is often a similarity measure between genes or samples as a function of the rows or columns in the gene expression matrix. A simultaneous clustering, also called "biclustering," of both genes and samples is proposed in [2].

A similar approach to our two-phase clustering strategy is the Distributed Data Mining (DDM). An overview of some state-of-the-art research results is given in [10]. Januzaj et al. [8] propose a density-based distributed clustering approach. Their recursive technique consists of four different steps, whereas they assume that the data is horizontally distributed. In the first step, the data is clustered locally using the DBSCAN algorithm [5] followed by the determination of local models. In the models, each local cluster is represented by a set of specific core points. These local models are used to compute the global clustering model with the help of DBSCAN, too. In the fourth step, the global result is sent back to the local sites to update the local clustering. This step is necessary to consider data dependencies between local sites. For vertical distributed data sets, a collective hierarchical clustering algorithm is proposed by Johnson et al. [9].

Zeng et al. [15] have developed an adaptive meta-clustering approach combining the information from different clustering results. In their proposal, different clustering results are computed from one single input data set using different algorithms. The objective of their research is to provide a better understanding of the data, because all available clustering approaches are heuristic and can only derive an approximation of the optimal result.

3 Preliminaries

A powerful and effective method to estimate an unknown density function f in a non-parametric way from a set of data points is kernel density estimation [13, 14] and is defined as follows:

Definition 1 (*Kernel Density Estimation*) Let $D \subset \mathbb{R}^d$ be a data set, h be the smoothness level. Then, the kernel density estimate $\hat{f^D}(x)$ based on the kernel K is defined as:

$$\hat{f^D}(x) = \frac{1}{nh^d} \sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right)$$

with $0 \leq \hat{f^D}$ and $\int K(x)dx = 1$.

Various kernels K such as the gaussian kernel have been proposed in related literature. The computation of this standard kernel density function requires a number of operations in $O(n)$ (where n is the number of data points) to determine the density at a single point $x \in D$. In recent research activities, much attention has been paid to the development of more efficient density estimation methods.

The WARPing (Weighted Averaging of Rounded Points) framework describes an efficient way to develop density estimation methods based on pre-binning. Zhang et al. [17] propose a density function based on k centroids approximating the standard density function using this framework. Let $C = \{\mu_i \in D : 1 \leq i \leq k\}$ be the set of centroids, $I(x) = min\{i : dist(x, \mu_i) \leq dist(x, \mu_j) \forall j \in \{1, \ldots, k\}\}$ will be the index function delivering the minimal index of the nearest centroid $\mu_i(D) = \{x \in D : I(x) = i\}$ for the set of data points, which have μ_i as the nearest centroid, $n_i = \#\mu_i(D)$ the number and σ_i the standard deviation of the data points in $\mu_i(D)$. The determination of the centroids can be conducted using several vector quantization methods, e.g. k-means, centroid linkage or its popular variant BIRCH [16].

This density estimation based on k centroids uses the sufficient statistics (average, variance and the number of data points) of the Voronoi cells, which are given by the positions of the k centroids. The density function based on the Voronoi prebinning and the gaussian kernel for a given smoothing level h is:

$$\hat{f^C}(x) = \frac{1}{n} \sum_{i=1}^{k} \frac{n_i}{\sqrt{2\pi}\sqrt{\sigma_i^2 + h^2}^d} \cdot exp\left(-\frac{(x - \mu_i)^2}{2(\sigma_i^2 + h^2)}\right) \quad (1)$$

The computation of this density estimation function requires a number of operations in $O(k)$ (where k is the number of cluster centroids) to determine the density at a single point $x \in D$. This method reduces the runtime complexity significantly if $k \ll n$, where n is the number of data points in D. Figure 1 illustrates the impact of the data reduction and the loss of accuracy for a two-dimensional data sets. The density estimation was done with 50 centroids (Figure 1(b)). The loss of accuracy is depicted in Figure 1(c).

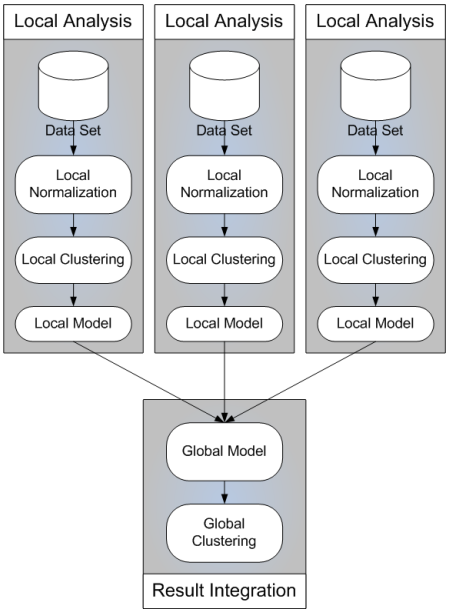


Figure 2: Two-phase Clustering Strategy

4 Two-Phase Clustering Strategy

With our new approach of retrieving cluster information from a set of gene expression data, we propose a novel way to incorporate data from several independent studies. In comparison to the classic approach the structure of the analysis process has changed. Normalization and a first statistical analysis, in our case clustering, are performed for each data set separately (Figure 2). This allows the adjustment of normalization and clustering according to the specific characteristic of the underlying data. Furthermore, data from different studies are grouped based on meaningful statistical values instead of measured raw intensity or ratio values.

To obtain a global interpretable view of l different gene expression data sets $S_1, \ldots, S_l$, an overall clustering result has to be computed. Therefore a set of local clustering results needs to be combined, which

1. consists of a varying number of genes

   $$\overrightarrow{X_i} = \{x_i^1, \ldots, x_i^{n_i}\}, 1 \leq i \leq l,$$

   where n_i is the number of investigated genes in the i^{th} microarray data set,
2. is divided in a varying number of clusters

   $$\overrightarrow{C_i} = \{c_i^1, \ldots, c_i^{k_i}\}, 1 \leq i \leq l,$$

   where k_i is the number of clusters in the i^{th} local clustering result and
3. has been derived from data of different dimensionality according to the numbers of samples within a single study.

A potential algorithm to combine l different local clustering results in gene expression analysis requires to overcome these problems.

4.1 Local Clustering and Local Model

The microarray data sets S_i are usually n_i x m_i matrices, where n_i is the number of genes and m_i is the number of samples. The entries represent the intensities of genes in samples. The first phase of our two-phase clustering strategy is that each gene expression data set S_i is separately normalized and clustered (called local normalization and local clustering). This allows the separate adjustment of normalization and clustering for each data set instead of having to deal with just one fully integrated data base of m different microarray data sets. The result of each local clustering is vector $\overrightarrow{C} = \{c_1, \ldots, c_k\}$, where c_i is the set of genes belonging to the i^{th} cluster. The number of clusters for each microarray data set can be different.

In order to compute a global clustering from a set of l different local clustering results, we require for each local clustering result a local model which satisfies the following aspects:

- All genes of the underlying microarray data sets must be included in the local model. This is necessary to know which genes are investigated in the experiment.
- The local model allows a good approximation of the similarity between genes according to the underlying intensity values and the clustering result.

A local model could consists of an n x k matrix LM, where n is the number of genes and k is the number of clusters. The entries of the matrix are either 0 or 1; while the value 0 indicates that the gene does not belong to the corresponding cluster, the value 1 indicates that the gene does belong to the cluster. The drawback of this local model is the lack of information on the data distribution of the underlying data set and the resulting inability to approximate the similarity of genes in a satisfying manner. This local model allows only to determine which genes belong to the same cluster.

A more accurate local model is also an n x k matrix LM, where the entries represent the density probability of a gene belongs to a cluster. We know that the local clustering is represented by a vector $\overrightarrow{C} = \{c_1, \ldots, c_k\}$, where c_i is the set of genes belonging to the i^{th} cluster. For each cluster, we can compute the centroid μ_i and the standard deviation σ_i. Moreover, we know that each gene belongs only to one cluster c_i. Using the formula (1) we can compute for each gene x the membership to a specific cluster μ_i based on the density estimation.

$$\hat{f^C}(x|\mu_i) = \frac{n_i}{n\sqrt{2\pi}\sqrt{\sigma_i^2 + h^2}^d} \cdot exp\left(- \frac{(x-\mu_i)^2}{2(\sigma_i^2 + h^2)} \right) \tag{2}$$

Compared to [15], we have reduced the computational complexity from $O(n^2)$ to $O(n \cdot k)$. Subsequently, we can determine a normalized density probability $dp(\mu_i|x)$ that a cluster μ_i includes the gene x.

$$dp(\mu_i|x) = \frac{\hat{f^C}(x|\mu_i)}{\hat{f^C}(x)} \tag{3}$$

For each gene $x_i \in \overrightarrow{X}, 0 \leq i \leq n$ in a gene expression data set the density probability for all centroids forms a vector $\overrightarrow{V_{x_i}} = \{dp_1, \ldots, dp_k\}$, where k denotes the number of local clusters. The vectors $\overrightarrow{V_{x_i}}$ for all genes x_i in the microarray data set S_i represent the local model. The complete procedure to determine the local model for a gene expression matrix is illustrated in algorithm 1.

To summarize, in the first phase of our two-phase clustering strategy, we transform each gene expression matrix

into a local model matrix with regard to the local clustering result. The local model matrix has a size of n x k, where n is the number of genes and k is the number of local clusters. The entries represent a normalized density probability that a gene belongs to a cluster. The advantage of this local model is that it includes information on the underlying data distribution and the local clustering result. The resulting local models can be made accessible over the network instead of the raw microarray data sets. The combination of l different local models to obtain a global clustering is presented in the following subsection.

4.2 Global Model and Global Clustering

The local models replace the raw microarray data sets in our approach as starting point for the global clustering. To obtain a global interpretable view of l user-specified different microarray data sets $S_1, \ldots, S_l$ now, an overall clustering result has to be computed from the local models. The first task in the second phase of our two-phase clustering strategy is to determine a global model of the l different local models. The resulting global model should represent the integrated information of the considered l local models, so that a global clustering could be computed. The last task in the second phase is to determine the global clusters from the global model.

The determination of the global model on the basis of the pure local models is difficult because of the different local clustering results. An important observation with regard to the local models is that the distance between two genes x_i and x_j in a local model is a measure of both genes belonging to the same cluster. Furthermore, the distance is an indicator for the similarity of the two genes in the underlying microarray data set.

$$dist(x_i, x_j) = \sqrt{\sum_{l=1}^{k} (dp_{i,l} - dp_{j,l})^2} : 1 \leq i, j \leq n, i \neq j$$

The entries in the local model are normalized density probability values for the event that the gene is included in the cluster. A high distance between two genes x_i and x_j indicates that the genes belong to different clusters and are therefore dissimilar. A small distance indicates that the genes belong to same cluster and are similar in the microarray data set. Using this distance function, a density-based similarity matrix $M : m_{ij} = dist(x_i, x_j), 1 \leq i, j \leq n, i \neq j$ can be derived for each local model. The resulting similarity matrices for the l local models satisfy all requirements to compute a global model in a subsequent step.

DETERMINING LOCAL MODEL
Required: Microarray Data Set S // gene expression matrix

Double[][] LM // local model matrix
S_{norm} = NORMALIZATION(S)
C = CLUSTERING(S_{norm})
for *all* $x_i \in S_{norm}$ **do**
  for *all* $\mu_j \in C$ **do**
    compute $LM[i][j] = dp(\mu_j|x_i) = \frac{\hat{f^C}(x_i|\mu_j)}{\hat{f^C}(x_i)}$
  end
end
return LM

Algorithm 1: Determining Local Model for a microarray data set S

For each of the m local models in our approach, a distance matrix $M^i, 0 \leq i \leq m$ is calculated. As part of the result integration, these matrices need to be combined resulting in one single distance matrix M^{global}:

$$M_{global} = w_1 \bullet M^1 + w_2 \bullet M^2 + ... + w_m \bullet M^m \quad (4)$$

The computation of the global matrix is similar to [15]. The global similarity matrix M^{global} contains the similarity between every two genes of the microarray data set. The global matrix is obtained by weighted addition of local results. The reason and the determination of the weighting factors will be considered in the next section.

From this relationship, a hierarchical and/or a density-based clustering result can be extracted dividing the full data set into groups of similar objects. In the case of a hierarchical clustering, each cluster $C_1, \ldots, C_n$ contains exactly one data point after initialization. In an iterative process, the entry showing the highest similarity between two clusters is identified and the corresponding clusters are merged while the affected entries in the distance matrix are updated. This iteration proceeds till a threshold defining the maximal cost for merging two clusters is reached. The data set gets separated into clusters. The similarity is defined by a distance measure based on a cost function, e.g. single, average or complete linkage. To increase the robustness towards outliers, we have chosen average-linkage hierarchical clustering.

4.3 Weighting of microarray data sets

Not all microarray experiments have the same quality. This fact should be considered in the computation of the global clustering result. The better the attested quality of local results, the stronger their impact on the global result. Aside from statistical properties of a clustering, such as the within-cluster standard deviation [15] or the silhouette index [11], technical and biological criteria could be used to estimate the weighting factors w_i of the local results within a distributed meta-analysis.

The term *technical criteria* refers to properties of the microarray or the experimental procedure itself, which could be used to utilize a quality measure for microarray data sets:

Existence of technical or biological replicates: A meaningful experimental result can be hardly achieved without replicates. While technical replicates are used to validate the labeling and hybridization process, biological replicates are based on separate RNA extraction from different individual samples. It is clear, that at least 3 replicates are necessary for statistical relevance.

Within-replicate variation: If replicates are available, the variation between replicates should be low to indicate that the experiments have been performed with high standards and that results can be reproduced.

Missing data points: For high-quality analysis, we favor complete data sets where the percentage of data points with no measurement available is low. Nevertheless, it frequently happens that some measurements can not be performed due to experimental problems or self-defined constraints, e.g. the exclusion of negative intensity values from the analysis of Affymetrix data sets.

Quality measures for the ranking of the scanning result after hybridization could contain the following features with a_i being a data point (spot) in an microarray scanning result and k being the total number of data points:

Saturation: Good scan results should contain a low percentage of saturated pixels within a data point. Fully saturated data points have to be excluded from the analysis. An averaged saturation factor can be defined as:
$SAT = \frac{1}{k} \sum_{i=1}^{k} \frac{\text{number of good pixels in data point } a_i}{\text{number of all pixels in data point } a_i}$

Shape Additionally, data points should show a compact round shape, signaling that the experiment has been performed successfully. A shape factor per data point can be described as:
$SHP = \frac{1}{k} \sum_{i=1}^{k} \frac{\text{data point area of } a_i}{\text{perimeter of } a_i}$

Homogeneity: The variation of within-data point pixel intensities should be small indicating that hybridization has been performed with high standards using a chip of high quality. The homogeneity of a data point a_i can be defined by the within-data point variance:
$HOM = \frac{1}{k} \sum_{i=1}^{k} var(a_i)$

Brightness: The ratio between foreground and background indicates the amount of bound biological material. For meaningful results a sufficient amount of biological material should be involved resulting in a high signal to noise ratio (SNR) which is commonly defined by $SNR = P_{Signal}/P_{Noise}$ and especially for microarray scan results as:
$SNR = \frac{\text{average foreground} - \text{average background}}{\text{standard deviation of background}}$

A combined technical criterion c_i^T for a data set corresponding to the distance matrix M_i given that all components are considered equally, can be calculated as:

$$c_i^T = SAT_i + SHP_i + HOM_i + SNR_i \quad (5)$$

Weights w_i^T based on technical quality measures for microarray experiments can then be derived accordingly:

$$w_i^T = \frac{c_i^T}{\sum_{i=1}^{m} c_i^T} \quad (6)$$

On the other hand, *biological criteria* could be used to estimate the quality of microarray data participating in a distributed analysis:

Description of sample attributes: Within gene expression experiments, the sample description is of great importance. However, in many data sets the description is inadequate. Therefore, an exact description could be used to weight the experiments, i.e. in the case of a comparison between tumor samples, we might expect the histological grading of the tissue and the survival time of the patients for performing DC to compare gene expression to the grade of the tumor or to survival time.

RNA quality: Criteria for sample RNA quality should be included since the quality of the input material (RNA) defines the quality of the output (gene expression data). Such criteria might be the $28S/18S$ ratio, which should be above 1.75, or quality values from PCR based approaches. With the use of more sophisticated instruments like the Agilent Bioanalyzer. it is possible to define other criteria for RNA quality.

Quality of the probe sequence: Different chip platforms use different types of probe designs. Spotted arrays usually consist of probes generated from PCR products of EST or other sources for a gene. Oligo arrays contain a different number of oligonucleotide probes for a single gene. In general, the annotation of the probes should contain stable identifiers of the common genomic databases used. Moreover, the exact location of the probe on the genome should be provided since splice variants of the RNA of genes might change the measurements. Also cross matching sequences should provided. If PCR products are used, consideration has to be given to the level of evidence on which the PCR product is annotated, i.e. based on sequencing of the product or based on the available annotation of the source sequence, since roughly 20 percent of the available EST clones are wrongly annotated.

In addition, a ranking of studies based on microarray experiments could be defined analogous to a standard scale similar to the existing evidence levels of medical studies [4]. Together with the technical and biological criteria mentioned above, this could lead to a model for the estimation of the weights of single studies within a distributed parallel meta-analysis.

5 Evaluation

For our evaluation, we generated synthetic three-dimensional data sets consisting of normally distributed natural clusters. Therefore, a priori knowledge of the data points' affiliation to clusters was available. In our initial configuration, it contained 1200 data points in four clusters with 600, 400, 100 and 100 data points. From this initial set, four data sets have been derived by changing the clusters' position in space and the within-cluster variation. To obtain test data similar to real microarray data sets, 10% of the data points in each cluster were arbitrarily moved to different clusters. One of the five sample data sets A, B, C, D, E can be seen in Figure 3 showing the generated clusters.

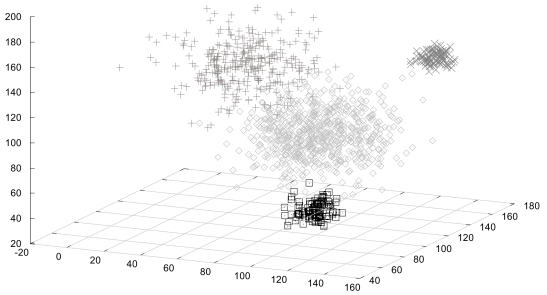


Figure 3: Three-dimensional test data set, labeled as generated.

Following the classic approach the five data sets were integrated, resulting in one multidimensional data set as basis for further data manipulations. Clustering was performed using a hierarchical average linkage algorithm with a Euclidean distance measure. The result is illustrated in Figure 4.

The local clusterings in our two-phase clustering strategy were performed using the k-means or the hierarchical clustering algorithm. The data sets were clustered in 10 clusters, which are more than the expected number of four.

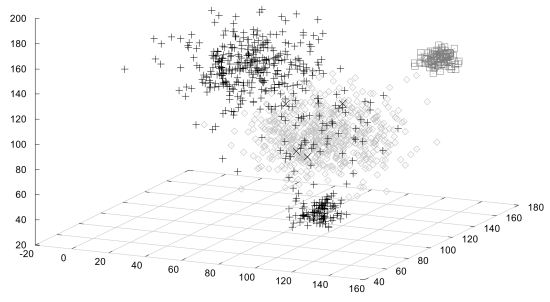


Figure 4: Clustering result for a three-dimensional test data set using the classic approach for meta-analysis.

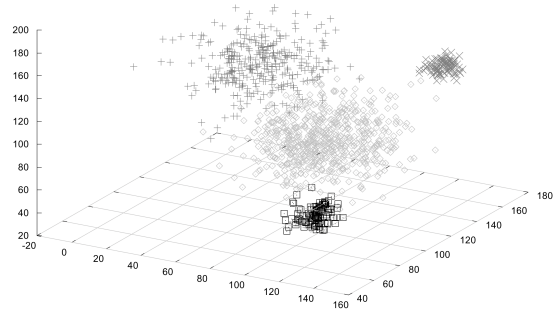


Figure 5: Clustering result for a three-dimensional test data set using our two-phase clustering strategy for meta-analysis.

After result integration, a hierarchical clustering algorithm has separated the data set into four clusters. The result can be seen in Figure 5. The example shows that our two-phase clustering strategy determines more accurate clusters than the classic approach.

In Figure 6, the jaccard indices [1] for the clustering results are shown. We see that for both approaches, the invariant set normalization has separated the data better than the simple scaling has. We also can see that our approach achieved better clustering than the classic approach. The example in Figure 4 illustrates that the classic method tend to assign points from clearly distinguishable clusters to the same cluster, whereas the parallel approach leads to results close to the original cluster structure. As there is no a priory knowledge of the correct clustering for real data, a technical evaluation of the clustering results cannot be performed.

Currently, we are analyzing real data for pancreatic cancer from Friess et al. [6] and Logsdon et al. [12] to perform a high-level comparison based on differentially expressed genes.

6 Conclusion

In this paper, we presented a novel approach for clustering of microarray data sets. Each data set is analyzed individually, which allows adjusted normalization and clustering. The local clustering is followed by the calculation of a local model based on density estimation. A set of local models is combined to a global model using a linear conjunction of derived density-based similarity matrices. Afterwards, the overall clustering can be computed from the global model. Furthermore, we presented an integrated approach to determine the weights of the microarray data sets within our

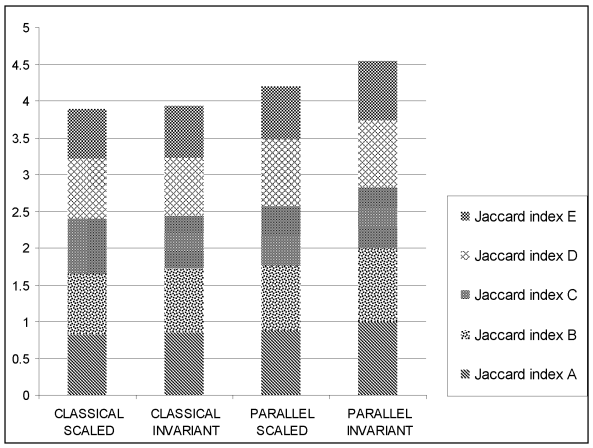


Figure 6: Correct classified pairwise cluster-based occurrence by means of the Jaccard index for test data

method. The weighting factors are determined based on technical as well as biological quality measures. The better the attested quality of local results, the stronger their impact on the global result. The evaluation is done with synthetic three-dimensionally generated data sets consisting of normally distributed natural clusters. Currently, we are analyzing real data for pancreatic cancer from Friess et al. [6] and Logsdon et al. [12] to perform a high-level comparison based on differentially expressed genes.

References

[1] J. Bryan. Problems in gene clustering based on gene expression data. *Journal of Multivariate Analysis*, 90:44–66, 2004.

[2] Yizong Cheng and George M. Church. Biclustering of expression data. In *Proceedings of the Eighth International Conference on Intelligent Systems for Molecular Biology (ISMB'00, August 19-23, 2000, La Jolla / San Diego, CA, USA)*, pages 93–103, 2000.

[3] Hugh Chipman, Trevor J. Hastie, and Robert Tibshirani. Clustering microarray data. In Terry Speed, editor, *Statistical Analysis of Gene Expression Microarray Datas*, pages 159–200. Chapman and Hall/CRC, 2003.

[4] Deborah J. Cook, Gordon H. Guyatt, Andreas Laupacis, David L. Sackett, and Robert J. Goldberg. Clinical recommendations using levels of evidence for antithrombotic agents. *Chest*, 108(4 Suppl):227–230, 1995.

[5] Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD'96)*, pages 226–231, 1996.

[6] H. Friess, J. Ding, J. Kleeff, L. Fenkell, J. A. Rosinski, A. Guweidhi, J. F. Reidhaar-Olson, M. Korc, J. Hammer, and M. W. Büchler. Microarray-based identification of differentially expressed growth and metastasis-associated genes in pancreatic cancer. *CMLS Cellular and Molecular Life Science*, 60:1180–1199, 2003.

[7] Reinhard Hoffmann, Thomas Seidl, and Martin Dugas. Profound effect of normalization on detection of differentially expressed genes in oligonucleotide microarray data analysis. *Genome Biology*, 3:0033.1–0033.11, 2002.

[8] Eshref Januzaj, Hans-Peter Kriegel, and Martin Pfeifle. Dbdc: Density based distributed clustering. In *Proceeding of the 9th International Conference on Extending Database Technology (EDBT'04, Heraklion,Crete, Greece, March 14-18, 2004)*, pages 88–105, 2004.

[9] Erik L. Johnson and Hillol Kargupta. Collective, hierarchical clustering from distributed, heterogeneous data. In *Proceedings of the Large-Scale Parallel Data Mining, Workshop on Large-Scale Parallel KDD Systems, SIGKDD, August 15, 1999, San Diego, CA, USA*, pages 221–244, 1999.

[10] Hillol Kargupta and Philip Chan. *Advances in Distributed and Parallel Knowledge Discovery*. AAAI/MIT Press, 2000.

[11] L. Kaufmann and P. Rousseeuw. *Finding Groups in Data: An Introduction to Cluster Analysis*. Wiley New York, 1990.

[12] Craig D. Logsdon, Diane M. Simeone, Charles Binkley, Thiruvengadam Arumugam, Joel K. Greenson, Thomas J. Giordano, David E. Misek, and Samir Hanash. Molecular profiling of pancreatic adenocarcinoma and chronic pancreatitis identifies multiple genes differentially regulated in pancreatic cancer. *Cancer Research*, 63:2649–2657, 2003.

[13] D.W. Scott. *Multivariate Density Estimation*. Wiley and Sons, 1992.

[14] B. W. Silverman. *Density Estimation for Statistics and Data Analysis*. Chapman & Hall, 1986.

[15] Yujing Zeng, Jianshan Tang, Javier Garcia-Frias, and Guang R. Gao. An adaptive meta-clustering approach: Combining the information from different clustering results. In *Proceedings of the 1st IEEE Computer Society Bioinformatics Conference (CSB'02, 14-16 August 2002, Stanford, CA, USA)*, pages 276–287, 2002.

[16] Tian Zhang, Raghu Ramakrishnan, and Miron Livny. Birch: An efficient data clustering method for very large databases. In *Proceedings of the 1996 ACM SIGMOD International Conference on Management of Data, Montreal, Quebec, Canada, June 4-6, 1996*, pages 103–114. ACM Press, 1996.

[17] Tian Zhang, Raghu Ramakrishnan, and Miron Livny. Fast density estimation using cf-kernel for very large databases. In *Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining(San Diego, August 15-18, CA, USA. ACM, 1999*, pages 312–316, 1999.

An Evaluation of Text Retrieval Methods for Similarity Search of multi-dimensional NMR-Spectra

Alexander Hinneburg[1], **Andrea Porzel**[2], **Karina Wolfram**[2]
[1] Institute of Computer Science
Martin-Luther-University of Halle-Wittenberg, Germany
hinneburg@informatik.uni-halle.de
[2] Leibniz Institute of Plant Biochemistry (IPB), Germany
{aporzel,kwolfram}@ipb-halle.de

Abstract

Searching and mining nuclear magnetic resonance (NMR)-spectra of naturally occurring substances is an important task to investigate new potentially useful chemical compounds. Multi-dimensional NMR-spectra are relational objects like documents, but consists of continuous multi-dimensional points called peaks instead of words. We develop several mappings from continuous NMR-spectra to discrete text-like data. With the help of those mappings any text retrieval method can be applied. We evaluate the performance of two retrieval methods, namely the standard vector space model and probabilistic latent semantic indexing (PLSI). PLSI learns hidden topics in the data, which is in case of 2D-NMR data interesting in its owns rights. Additionally, we develop and evaluate a simple direct similarity function, which can detect duplicates of NMR-spectra. Our experiments show that the vector space model as well as PLSI, which are both designed for text data created by humans, can effectively handle the mapped NMR-data originating from natural products. Additionally, PLSI is able to find meaningful "topics" in the NMR-data.

1 Introduction

Nuclear magnetic resonance (NMR)-spectra are an important finger printing method to investigate the chemical structure of organic compounds from plants or other tissues. Two-dimensional-NMR spectroscopy is able to capture the influences of two different atom types at the same time (e.g. ^{1}H, hydrogen and ^{13}C carbon). The result of an 2D-NMR experiment can be seen as an intensity function measured over two variables[1]. Regions of high intensity are called peaks, which contain the real information about the underlying molecular structure. The usual visualizations of 2D-NMR spectra are contour plots as shown in figure 1. An ideal peak would register as a small dot, however, due to the limited resolution available (dependent on the strength of the magnetic field) multiple peaks may appear as a single merged object with non-convex shape. In the literature peaks are noted by their two-dimensional positions without any information about the shapes of the peaks. Content-based similarity search of 2D-NMR spectra would be a valuable tool for structure investigation by comparing spectra of unknown compounds with a set of spectra, for which the structures are known. While the principle is already in use for 1D-NMR spectra [7; 1; 11; 6; 2], to the best of our knowledge, no effective similarity search method is known for 2D-NMR-spectra.

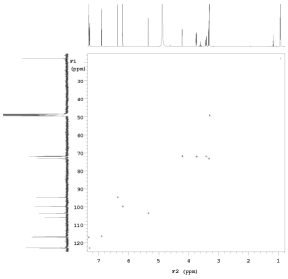


Figure 1: 2D-NMR spectrum of quercetrin. The plots at the axes are the corresponding 1D-NMR spectra.

Simplified, a 2D-NMR spectrum is a set of two-dimensional points. There is an analogy to text retrieval, where documents are usually represented as sets of words. Latent space models [5; 9; 3] were successfully used to model documents and thus improved the quality of text retrieval. Recently, a diversity of text mining approaches for different problems [4; 12; 8] have been proposed, which make use of probabilistic latent space models. The goal of this work is to show by example how to apply text retrieval and mining methods to biological data originating from experiments.

The contribution of this paper are methods to map 2D-NMR spectra to discrete text-like data, which can be analyzed and searched by any text retrieval method. We evaluate on real data the performance of two text retrieval methods, namely the standard vector space model [10] and PLSI [5] in combination our mapping methods for 2D-NMR spectra. Additionally, we propose a simple similarity function, which operates directly on the peaks of the spectra and serves as bottom line benchmark in the experimental evaluation. Our results indicate at a larger scope that text retrieval and mining methods, designed for text data created by humans, in combination with appropriate mapping functions may yield the potential to be also successful for experimental data from naturally occurring objects. In this paper we consider exemplarily ^{1}H, ^{13}C one-bond heteronuclear shift correlation 2D-NMR spectra.

The paper is structured as follows: first, in section 2, we introduce briefly the used text modeling methods while in section 3, we propose the mapping functions for 2D-NMR spectra. In section 4, we propose a simple similarity function as bottom line benchmark and define fuzzy duplicates.

[1] The measurements are in parts per million (ppm).

In section 5, we describe our experimental evaluation and section 6 concludes the paper.

2 Models for Text Retrieval

Like a 2D-NMR spectrum consists of a set of peaks, a document consists of many words, which typically are modeled as a set. So assuming a 2D-NMR spectrum can be transformed into a text-like object by mapping the continuous 2D peaks to discrete variables, a variety of text retrieval models can be applied. However, it is an open question, whether models designed for quite different data, namely texts created by humans, are effective on data which comes for naturally occurring compounds and thus do not include human design patterns.

We briefly introduce the essentials of the vector space model and PLSI to make the paper self contained. In the vector space model, documents are represented by vectors which have as many dimensions as there are words in the used vocabulary of the document collection. Each component of a documents vector reflects the importance of the corresponding word for the document. The typical quantity used is the raw term frequency (tf) of that word for the document, say the number of occurrences of that word in a document d. In order to improve the retrieval quality, those vectors are reweighed by multiply tf with the inverse document frequency (ifd) of a word. The inverse document frequency measures is large, if a word is included in only a small percentage of the documents in the collection. Formally, we denote the set of documents by $D = \{d_1, ..., d_J\}$ and the vocabulary by $W = \{w_1, ..., w_I\}$. The term frequency of a word $w \in W$ in a document $d \in D$ is denoted as $n(d, w)$ and the reweighed quantity is $\hat{n}(d, w) = n(d, w) \cdot idf(w)$. The similarity between a query document q and a document d from the collection is

$$S(d, q) = \frac{\sum_{w \in W} \hat{n}(d, w) \cdot \hat{n}(q, w)}{\sqrt{\sum_{w \in W} \hat{n}(d, w)^2} \cdot \sqrt{\sum_{w \in W} \hat{n}(q, w)^2}}$$

This can be interpreted as the cosine of the angles between the two vectors.

Probabilistic latent semantic indexing (PLSI) introduced in [5] extends the vector space model by learning topics hidden in the data. The training data consists of a set of document-word pairs $(d^{(i)}, w^{(i)})_{i=1,...,N}$ with $w^{(i)} \in W$ and $d^{(i)} \in D$. The joint probability of such a pair is modeled according to the employed aspect model as $P(d, w) = \sum_{z \in Z} P(z) \cdot P(w|z) \cdot P(d|z)$. The z are hidden variables, which can take K different discrete values $z \in Z = \{z_1, ..., z_K\}$. In the context of text retrieval z is interpreted as an indicator for a topic. Two assumptions are made by the aspect model. First, it assumes pairs (d, w) to be statistically independent. Second, conditional independence between w and d is assumed for a given value for z.

The probabilities necessary for the joint probability $P(d, w)$, namely $P(z)$, $P(w|z)$ and $P(d|z)$, are derived by an expectation maximization (EM) learning procedure. The idea is to find values for unknown probabilities, which maximize the complete data likelihood

$$\begin{aligned} P(S, z) &= \prod_{(d^{(i)}, w^{(i)}) \in S} [P(z) \cdot P(w^{(i)}|z) \cdot P(d^{(i)}|z)] \\ &= \prod_{d \in D} \prod_{w \in W} [P(z) \cdot P(w|z) \cdot P(d|z)]^{n(d,w)} \end{aligned}$$

with $S = \{(d^{i)}, w^{i)})_{i=1,...,N}\}$ is the set of all document-word pairs in the training set. In the E-step the posteriors for z are computed.

$$P(z|d, w) = \frac{P(z) \cdot P(w|z) \cdot P(d|z)}{\sum_{z' \in Z} P(z') \cdot P(w|z') \cdot P(d|z')}$$

The subsequent M-step maximizes the expectation of the complete data likelihood respectively to the model parameters, namely $P(z)$, $P(w|z)$ and $P(d|z)$.

$$\begin{aligned} P(d|z) &= \frac{\sum_{w \in W} P(z|d, w) \cdot n(d, w)}{\sum_{w \in W} \sum_{d' \in D} P(z|d', w) \cdot n(d', w)} \\ P(w|z) &= \frac{\sum_{d \in D} P(z|d, w) \cdot n(d, w)}{\sum_{w' \in W} \sum_{d \in D} P(z|d, w') \cdot n(d, w')} \\ P(z) &= \frac{\sum_{w \in W} \sum_{d \in D} P(z|d, w) \cdot n(d, w)}{\sum_{w \in W} \sum_{d \in D} n(d, w)} \end{aligned}$$

The EM algorithm starts with random values for the model parameters and converges by alternating E- and M-step to a local maximum of the likelihood.

There are serval ways possible to answer similarity queries using the trained aspect model. Because of its simplicity, we adopt the PLSI-U variant from [5]. The idea is to extend the cosine similarity measure from the tf-idf vector space model. The extension by Hofmann treats the learned multinomials $P(w|d)$ as term frequencies (tf). Note that $P(w|d) = P(d, w)/P(d)$ with $P(d) = \sum_{w' in W} n(d, w')/N$. The multinomials $P(w|d)$ are smoothen variants of the original term frequencies $\tilde{P}(w|d) = n(d, w)/(\sum_{w' \in W} n(d, w'))$. The proposed tf-weights are linear combinations of the multinomials $P(w|d)$ and $\tilde{P}(w|d)$. Thus, the new tf-idf weights used for the documents within the similarity calculation are

$$\hat{n}(d, w) = (\lambda \cdot P(w|d) + (1 - \lambda) \cdot \tilde{P}(w|d)) \cdot idf(w)$$

with $\lambda \in [0, 1]$. Hofmann suggests in [5] to set $\lambda = 0.5$. The tf-idf weights for the query are determined as in the standard vector space model. The smoothen tf-weight for a word which actually does not appear in the document may be still non-zero if the word belongs to a topic which is covered by the particular document. In that way a more abstract similarity search becomes possible.

For 2D-NMR spectra similarity search it is not clear, what is the best way to map the peaks of a spectrum to discrete words. We develop methods for this task in the next section. That will enable us to tackle the question, whether methods like the vector space model or PLSI, which is designed for text data, remains effective for experimental data from natural products.

3 Mapping of NMR Spectra

In this section we propose different methods to map the peaks of an NMR-spectrum from the continuous space of measurements to a discrete space of words. With the help of such a mapping, methods for text retrieval like PLSI can be directly applied. However, the quality of the similarity search depend on how the peaks are mapped to discrete words. A preliminary study of the proposed mappings appeared as poster in [13].

First we give a formal definition of 2D-NMR spectra. A two-dimensional NMR-spectrum of an organic compound captures many structural characteristics like rings and chains. Most important are the positions of the peaks.

As the shape of a peak and its height (intensity) strongly varies over different experiments with the same compound, the representation of a spectrum includes the peak positions only.

Definition 1 *A **2D NMR-spectrum** A is defined as a set of points $\{x_1, \ldots, x_n\} \subset \mathbb{R}^2$. The $|\cdot|$ function denotes the size of the spectrum $|A| = n$.*

The size of a spectrum is typically between 4 and 30 for small molecules found in naturally occurring products.

3.1 Grid-based Mapping

We introduce a simple grid-based method, on which we will build more sophisticated methods. A simple grid-based method is to partition each of the both axis of the two-dimensional peak space into intervals of same size. Thus, an equidistant grid is induced in the two-dimensional peak space and a peak is mapped to exactly one grid cell it belongs to. When a grid cell is identified by a discrete integer vector consisting of the cells coordinates the mapping of a peak $x \in \mathbb{R}^2$ is formalized as

$$g(x) = (g_c(x.c), g_h(x.h)) \text{ with } g_c(x.c) = \left\lfloor \frac{x.c}{w_c} \right\rfloor,\ g_h(x.h) = \left\lfloor \frac{x.h}{w_h} \right\rfloor$$

The quantities w_c and w_h are the extensions of a cell in the respective dimensions, which are parameters of the mapping. The grid is centered at the origin of the peak space. The cells of the grid act as words. The vocabulary generated by the mapped peaks consists of those grid cells which contain at least one peak. Empty grid cells are not included in the vocabulary. A word consists of a two-dimensional discrete integer vector.

Unfortunately the grid-based mapping has two disadvantages. First, close peaks may be mapped to different grid cells. This may lead to poor matching of related peaks in the discrete word space. Second, peaks of new query spectra are ignored when they are mapped to grid cells not included in the vocabulary. So some information from the query is not used for the similarity search which may weaken the performance.

3.2 Redundant Mappings

We propose three mappings which introduce certain redundancies by mapping a single peak to a set of grid cells. The redundancy in the new mappings shall compensate for the drawbacks of the simple grid-based mapping.

Shifted Grids

The first disadvantage of the simple grid-based method is that peaks which are very close in the peak space may be mapped to different grid cells, because a cell border is between them. So proximity of peaks does not guaranty that they are mapped to the same discrete cell.

Instead of mapping a peak to a single grid cell, we propose to map it to a set of overlapping grid cells. This is achieved by several shifted grids of the same granularity. In addition to the base grid some grids are shifted into the three directions $(1,0)(0,1)(1,1)$. An illustration of the idea is sketched in figure 2. In figure 2, one grid is shifted in each of the directions by half of the extent of a cell. In general, there may be $k-1$ grids shifted by fractions of $1/k, 2/k, \ldots, k-1/k$ of the extent of a cell in each direction respectively. For the mapping of the peaks to words which consist of cells from the different grids, two additional dimensions are needed to distinguish (a) the $k-1$ grids in each direction and (b) the directions themselves.

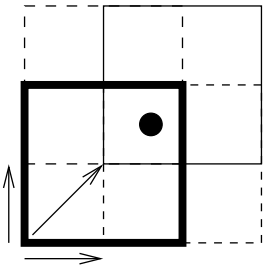


Figure 2: The four grids are marked as follows: base grid is bold, $(1,0)$, $(0,1)$ are dashed and $(1,1)$ is normal.

The third coordinate represents the fraction by which a cell is shifted and the fourth one represents the directions by the following coding: value 0 is (0,0), 1 is (1,0), 2 is (0,1) and 3 is (1,1). So each peak is mapped to a finite set of four-dimensional integer vectors. The mapping of a peak $x \in \mathbb{R}^2$ is

$$\begin{aligned} s(x) \quad = \quad & \{(g_c(x.c), g_h(x.h), 0, 0)\} \cup \\ & \bigcup_{i=1}^{k-1} \Big\{ (g_c(x.c + {}^{i}/_{k} \cdot w_c), g_h(x.h), i, 1), \\ & \quad (g_c(x.c), g_h(x.h + {}^{i}/_{k} \cdot w_h), i, 2), \\ & \quad (g_c(x.c + {}^{i}/_{k} \cdot w_c), g_h(x.h + {}^{i}/_{k} \cdot w_h), i, 3) \Big\} \end{aligned}$$

Thus, a single peak is mapped to $3(k-1)+1$ words. A nice property of the mapping is that there exists at least one grid cell for every pair of matching peaks both peaks are mapped to.

Different Resolutions

The second disadvantage of the simple grid-based mapping comes from the fact that empty grid cells (not occupied by at least one peak from the set of training spectra) do not contribute to the representation to be learned for similarity search. So peaks of new query spectra mapped to those empty cells are ignored. That effect can be diminished by making the grid cells larger. However, this is counterproductive for the precision of the similarity search due to the coarser resolution. Thus, there are two contradicting goals, namely (a) to have a fine resolution to handle subtle aspects in the data and (b) to cover at the same time the whole peak space by a coarse resolution grid so that no peaks of a new query spectrum have to be ignored.

Instead of finding a tradeoff for a single grid, both goals can be served by combining simple grids with different resolutions. Given l different resolutions $\{(w_c^{(1)}, w_h^{(1)}), \ldots, (w_c^{(l)}, w_h^{(l)})\}$ a peak is mapped to l grid cells of different sizes. In order to distinguish between the different grids an additional discrete dimension is needed. So the mapping function is

$$r(x) = \bigcup_{i=1}^{l} \{(g_c^{(i)}(x), g_h^{(i)}(x), i)\}$$

with $g_c^{(i)}$ and $g_h^{(i)}$ use $w_c^{(i)}$ and $w_h^{(i)}$ respectively. Note that a hierarchical, quad-tree like partitioning is a special case of the proposed mapping function with $w_c^{(i)} = 2^{i-1} w_c$ and $w_h^{(i)} = 2^{i-1} w_h$.

Combining shifted Grids with different Resolutions

Both methods are designed to compensate for different drawbacks of the simple grid mapping. So it is nat-

ural to combine both mappings. The parameters of such a mapping are the number of shifts k, the number of different grid cell sizes l and the actual sizes $\{(w_c^{(1)}, w_h^{(1)}), \ldots, (w_c^{(l)}, w_h^{(l)})\}$. Beside the two coordinates for the grid cells, additional discrete dimensions are needed for the shift, the direction and the grid resolution. Using the the definitions from above the mapping function of the combined mapping of a peak is

$$
\begin{aligned}
c(x) = \quad & \bigcup_{i=1}^{l} \left\{ \left(g_c^{(i)}(x.c), g_h^{(i)}(x.h), 0, 0, i\right) \right\} \cup \\
& \bigcup_{j=1}^{k-1} \Big\{ \left(g_c^{(i)}(x.c + {}^{j}/_{k} \cdot w_c^{(i)}), g_h^{(i)}(x.h), j, 1, i\right), \\
& \quad \left(g_c^{(i)}(x.c), g_h^{(i)}(x.h + {}^{j}/_{k} \cdot w_h^{(i)}), j, 2, i\right), \\
& \quad \left(g_c^{(i)}(x.c + {}^{j}/_{k} \cdot w_c^{(i)}\right), \\
& \quad g_h^{(i)}(x.h + {}^{j}/_{k} \cdot w_h^{(i)}), j, 3, i) \Big\}
\end{aligned}
$$

Thus a single peak is mapped to $l(3(k-1)+1)$ words. In the next section all mappings are compared with respect to the effectiveness for similarity search.

4 Directly Computing Similarity

In this section, we introduce a method to directly compute similarity between pairs of spectra. This method will be used in the experiments as a bottom line benchmark. We also propose on the basis of direct similarity a definition of fuzzy duplicates.

As a peak in a spectrum has two numeric attributes, which can vary continuously, we formalize the notion of matching peaks. A simple but effective approach is to require that a peak matches other peaks only within a certain spatial neighborhood. The neighborhood is defined by the ranges α and β.

Definition 2 *A peak x from spectrum A **matches** a peak y from spectrum B, iff $|x.c - y.c| < \alpha$ and $|x.h - y.h| < \beta$, where $.c$ and $.h$ denote the NMR measurements for carbon and hydrogen respectively.*

Note that a single peak of a spectrum can match several peaks from another spectrum. Given two spectra A and B, the subset of peaks from A which find matching partners in B is denoted as $matches(A, B) = \{x \colon x \in A, \exists y \in B \colon x \text{ matches } y\}$. The function $matches$ is not symmetric, but helps to define a symmetric similarity measure

Definition 3 *Let be A and B two given spectra and $A' = matches(A, B)$ and $B' = matches(B, A)$, so the **similarity** is defined as*

$$sim(A, B) = \frac{|A'| + |B'|}{|A| + |B|}$$

The measure is close to one if most peaks of both spectra are matching peaks. Otherwise the similarity drops towards zero.

An important application of similarity search is the detection of duplicates to increase the data quality of a collection of 2D-NMR-spectra. Clearly a naive definition of duplicates does not work, like two duplicate spectra A and B need to have the same size and the peaks at the same positions. The reason is that the spectra are measured experimentally and so the peak positions differ even if the same probe is analyzed twice. So flexibility should be allowed for the peak positions. Another problem appears when two spectra of the same substance are measured with different resolutions. In case a spectrum is measured with low resolution it may happen that neighboring peaks are merged to a single one. A restriction to an one-to-one relationship between matching peaks can not handle such cases.

We propose a definition of fuzzy duplicates based on the direct similarity measure, which can deal with both of the mentioned problems.

Definition 4 *A pair of 2D-NMR-spectra A and B are **fuzzy duplicates**, iff $sim(A, B) = 1$.*

By that definition it is only required that every peak of a spectrum finds at least one matching peak in the other spectrum.

Group	#Spectra	#Peaks
Pregnans	11	17–26
Anthrquinones	8	3–6
Aconitanes	8	22–26
Triterpenes	17	24–31
Flavonoids	18	5–8
Isoflavonoids	16	5–7
Aflatoxins	8	8–10
Steroids	12	16–23
Cardenolides	15	18–25
Coumarins	19	3–8

Table 1: Groups with number of spectra and range of peaks

5 Evaluation and Results

In this section we present the results for duplicate detection, a comparison of the effectiveness of the mappings for similarity search, and mining aspects of 2D-NMR-data.

5.1 2D-NMR-Data

The substances included in the database are mostly secondary metabolites of plants and fungi. They cover a representative area of naturally occurring compounds and originate either from experiments or from simulations[2] based on the known structure of the compound. The database includes about 587 spectra, each has about 3 to 35 peaks. The total number of peaks is 7029. Ten small groups of chemically similar compounds are included in the database for controlled experiments. The groups with the number of spectra and number of peaks are listed in table 1 left. The peak space with all peaks in the database is shown in figure 3 right. Two groups, steroids and flavonoids, are selected as examples and shown with their peak distribution within figure 3 right.

Natural steroids occur in animals, plants and fungi. They are vitamins, hormones or cardioactive poisons like digitalis or oleander. The steroids in the database ar mostly hormones like androgens and estrogens. Flavonoids are aromatic substances (rings). Some flavonoids decrease vascular permeability or possess antioxidant activity which can have an anticarcinogenic effect.

5.2 Detection of Duplicates

We used the direct similarity function introduced in section 4 to detect duplicates in the database. With a setting

[2] ACD/2D NMR predictor, version 7.08, http://www.acdlabs.com/

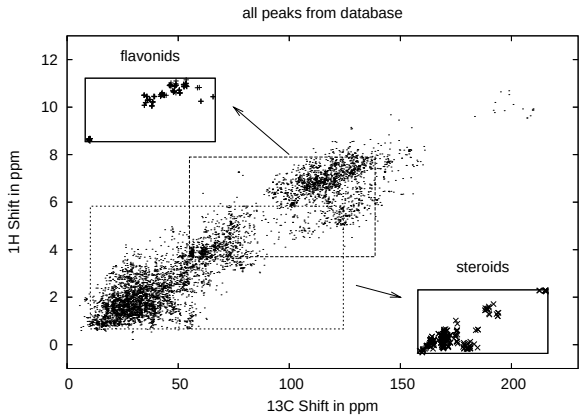


Figure 3: Distribution of the peaks of all spectra with the distribution within the groups of flavonoids and steroids.

of $a = 3$ppm and $b = 0.3$ppm, which are reasonable tolerances, 54 of 171991 possible pairs are reported as fuzzy duplicates. An inspection by hand revealed that 30 pairs are just very similar spectra, but 24 are candidates for real duplicates. Many of the found pairs come from the groups shown in table 1. Some pairs consist of an experimental and a simulated spectrum of the same substance which confirms the usefulness of the definition. There was also a surprise, namely the pair Thalictrifoline/Cavidine. Both structures differ only in the stereochemical orientation of one methyl group. Evidently, in this case the commercial software package used for the simulation is not able to reflect the different stereochemistry in calculated spectra. In the future, fuzzy duplicates will be used to improve the quality of collections of 2D-NMR spectra.

5.3 Performance Evaluation

The different methods for similarity search of 2D-NMR-spectra are compared using recall-precision curves. The search quality is high, when both – recall and precision – are high. So the upper curves are the best.

First, a series of experiments is conducted using our proposed mapping functions in combination with the vector space model. Each spectrum from the ten groups is used as a query while the rest of the respective group should be found as answers. The plots in figure 4 and 5 show averages over all queries. The results for the simple grid-based mapping are shown in figure 4a. The sizes of the grid cells are varied over $w_c = 4, 6, 8, 10$ and $w_h = 0.4, 0.6, 0.8, 1.0$ respectively. Small sizes give the best results.

The use of shifted grids improves the performance substantially over simple grids, as shown in figure 4b,c. The plots show the experiments for $k = 2, 3$. The results for $k = 2$ and $k = 3$ are almost identical. However, the vocabulary for $k = 2$ is much smaller. In practise, the smaller model with $k = 2$ shifts is favored.

Also the mapping based on grids with different grid cell sizes are assessed. Due to lack of space, only the results from combinations of $w_c^{(1)} = 4, w_h^{(1)} = 0.4$ with other sizes are reported, because those performed best among all combinations. Figure 4d shows that also the mapping based on different grid cell sizes outperforms the simple grid-based mapping. But the improvement is not as much as for shifted grids. The set of resolutions $\{(w_c^{(1)} = 4, w_h^{(1)} = 0.4), (w_c^{(2)} = 10, w_h^{(2)} = 1.0)\}$ performs best.

Also, experiments are performed with the combination of the previous two mappings, namely a combination of shifted grids with those of different resolutions. The performance results are shown in figure 4e which indicates that the best combination, namely the resolution set $\{(w_c^{(1)} = 4, w_h^{(1)} = 0.4), (w_c^{(2)} = 10, w_h^{(2)} = 1.0)\}$ with $k = 2$ shifts, outperforms both previous mappings. This is more clearly seen in figure 4f which compares the best performing settings from the above experiments.

Next, a series of similar experiments is conducted using our proposed mapping functions in combination with PLSI. Random initialization is used for the EM training algorithm described in section 2. All curves are averages from cross validation over all groups. As PLSI is trained on the data beforehand, we used cross validation where the current query is not included in the taining data. As the groups are very small, the leave-one-out cross validation scheme is employed. The results for PLSI are shown in figure 5a-f. PLSI requires to chose the number of hidden aspects. For the experiments reported so far, the PLSI model is used with 20 hidden aspects. Also different numbers of aspects are tested using the best combination of mappings. Figure 5g shows that the performance with 10 aspects drops a bit The increase in the numbers of aspects from 20 to 32 is only marginally reflected in increase of search performance. So 20 is a reasonable number of aspects for the given data.

In summery, the experiments with both text retrieval methods show, that the mappings based on shifted grids and those with different resolutions perform significantly better than the simple grid-based mapping. In both cases, the combination of shifted grids and grids with different resolutions is even better than the individual mappings. The comparison between PLSI and the vector space model (figure 5h) shows that both have similar performance for small recall but for large recall PLSI has a better precision.

Last, the direct similarity function is tested (figure 5i). The size of the matching neighborhood is varied over $\alpha = 4, 6, 8, 10$ and $\beta = 0.4, 0.6, 0.8, 1.0$ respectively. The search quality is quite low. In fact on average, it fails to deliver a spectrum from the answer set in the top ranks which is indicated by the hill-like shape of the curves.

In conclusion, the results prove experimentally that the vector space model as well as the PLSI model, which are designed for text retrieval, are indeed effective for similarity search of 2D-NMR spectra from naturally occurring products.

5.4 Analysis of the latent Aspects

We analyzed the latent aspects learned by the PLSI model using the mapping based on the combination of shifted grids with different resolutions. The grid cells (words) with high probability for a given aspect are plotted together to describe the aspects meaning. Some aspects specialized on certain regions in the peak space which are typical for distinct molecule fragments like aromatic rings or alkane skeletons. However, also more subtle details of the data are captured by the aspect model. For example, the main aspect for the group of flavonoids specializes not only on the region for aromatic rings which are the main part of flavonoids. It also includes a smaller region which indicates oxygen substitution. A closer inspection of the database revealed that indeed many of the included flavonoids do have several oxygen substituents. The main aspect for flavonoids with the respective peak distribution of the flavonoid group

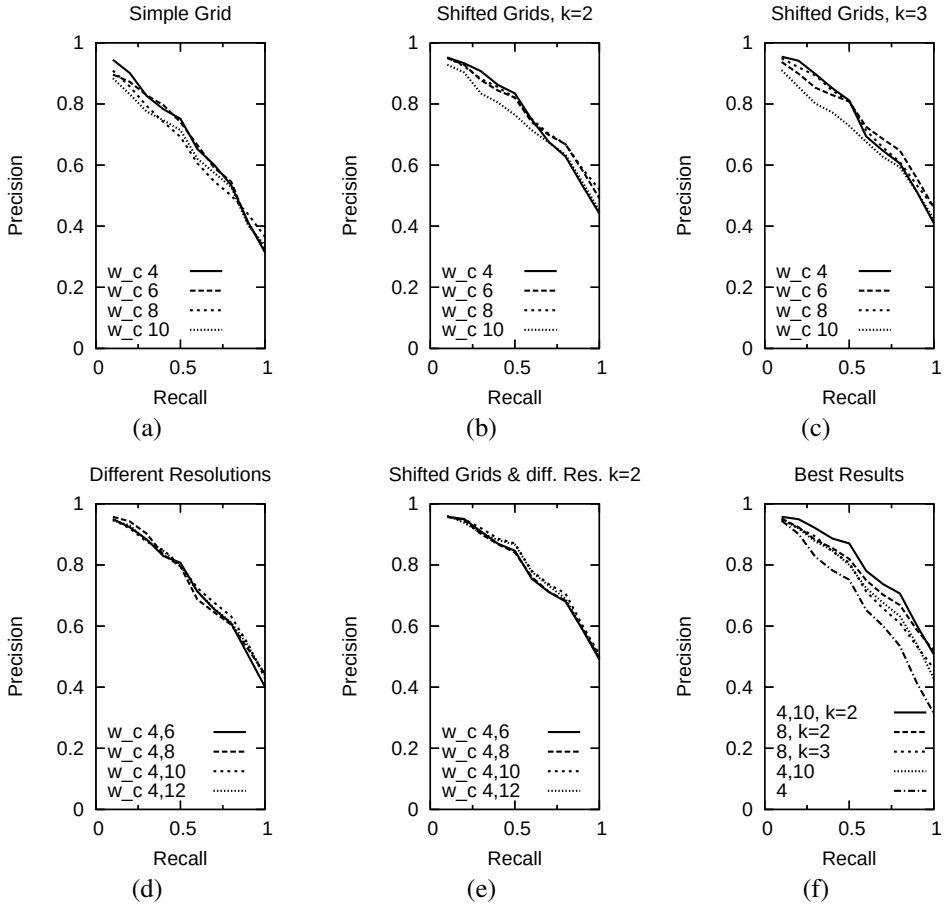


Figure 4: Average recall-precision curves using the vector space model

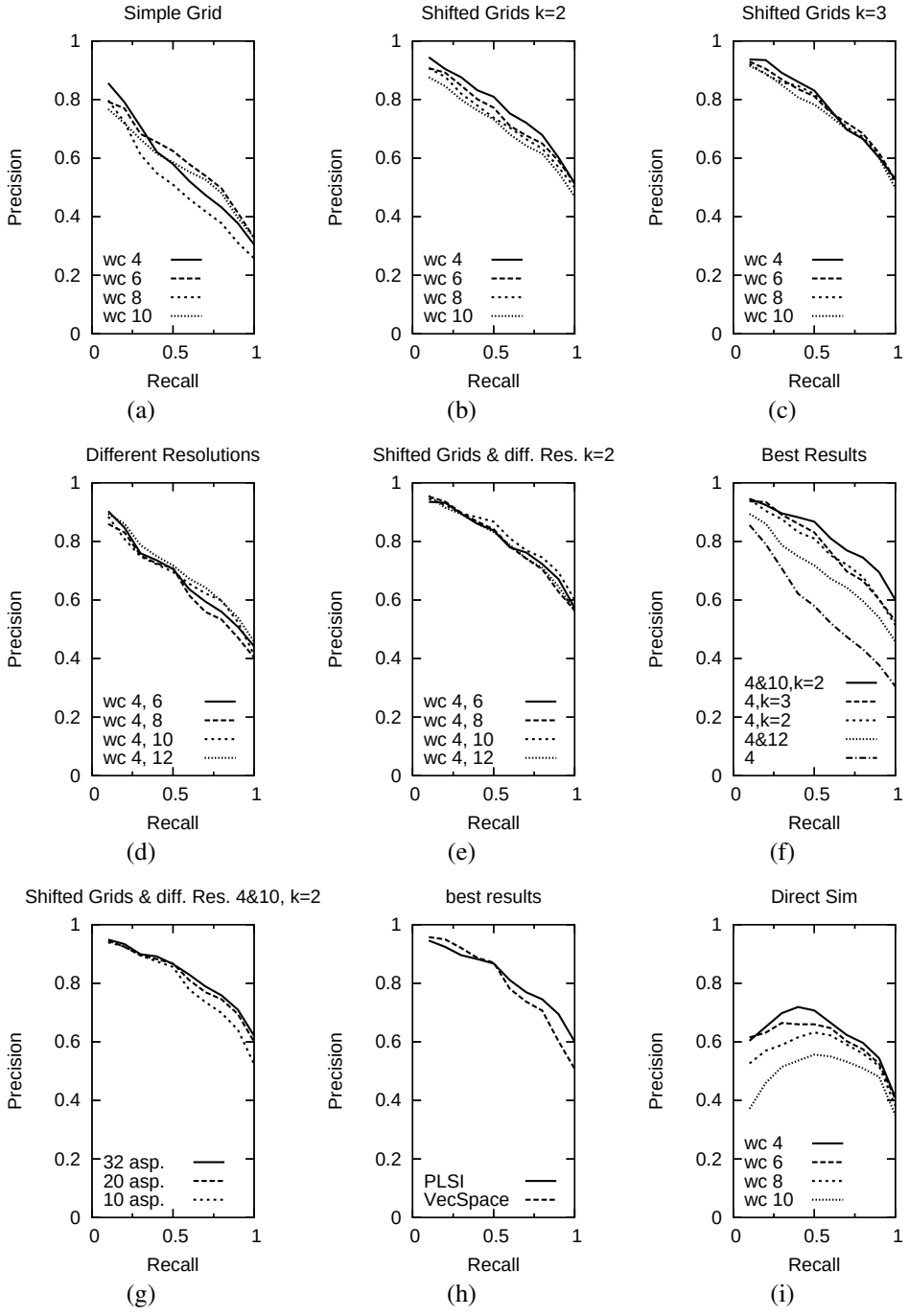


Figure 5: Average recall-precision curves from leave-one-out cross validation experiments with the PLSI model (a-g), best results of PLSI and vector space model (h) and results for the direct similarity (i).

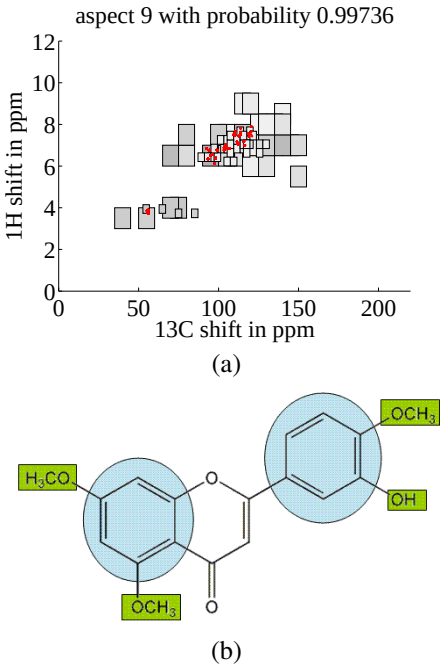


Figure 6: (a) Main aspect of the flavonoid group which includes the region of aromatic rings (upper right cluster) and the region for oxygen substituents (lower left cluster). The gray shades indicate the strength of the association between grid cell and aspect. (b) An example of an flavonoid (3'-Hydroxy-5,7,4'-trimethoxyflavone) where the aromatic rings and the oxygen substituents (methoxy groups in this case) are marked.

is shown in figure 6a. We believe a detailed analysis of the aspects found by the model may help to investigate unknown structures of new substances when their NMR-spectra are included in the training set.

6 Conclusion

We proposed redundant mappings from continuous 2D-NMR spectra to discrete text-like data which can be processed by any text retrieval method. We demonstrated experimentally the effectiveness of the our mappings in combination with the vector space model and PLSI. Further analysis revealed that the aspects found by PLSI are chemically relevant. In future research we will study more recent text models like LDA [3] in combination with our mapping methods.

References

[1] A. Tsipouras, J. Ondeyka, C. Dufresne et al. Using similarity searches over databases of estimated c-13 nmr spectra for structure identification of natural products. *Analytica Chimica Acta*, 316:161–171, 1995.

[2] A. S. Barros and D. N. Rutledge. Segmented principal component transform-principal component analysis. *Chemometrics & Intelligent Laboratory Systems*, 78:125–137, 2005.

[3] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

[4] L. Cai and T. Hofmann. Text categorization by boosting automatically extracted concepts. In *SIGIR '03*, 2003.

[5] T. Hofmann. Probabilistic latent semantic indexing. In *SIGIR '99*, 1999.

[6] P. Krishnan, N. J. Kruger, and R. G. Ratcliffe. Metabolite fingerprinting and profiling in plants using nmr. *Journal of Experimental Botany*, 56:255–265, 2005.

[7] M. Farkas, J. Bendl, D. H. Welti et al. Similarity search for a h-1 nmr spectroscopic data base. *Analytica Chimica Acta*, 206:173–187, 1988.

[8] Q. Mei and C. Zhai. Discovering evolutionary theme patterns from text: an exploration of temporal text mining. In *KDD '05*.

[9] A. Popescul, L. H. Ungar, D. M. Pennock, and S. Lawrence. Probabilistic models for unified collaborative and content-based recommendation in sparse-data environments. In *UAI '2001*.

[10] G. Salton, A. Wong, and C. S. Yang. A vector space model for automatic indexing. *Commun. ACM*, 18(11):613–620, 1975.

[11] C. Steinbeck, S. Krause, and S. Kuhn. Nmrshiftdb-constructing a free chemical information system with open-source components. *J. chem. inf. & comp. sci.*, 43:1733 –1739, 2003.

[12] M. Steyvers, P. Smyth, M. Rosen-Zvi, and T. Griffiths. Probabilistic author-topic models for information discovery. In *KDD '04*.

[13] K. Wolfram, A. Porzel, and A. Hinneburg. Similarity search for multi-dimensional nmr-spectra of natural products. In *PKDD '06*, 2006.

Frequent Subgraph Mining in Outerplanar Graphs*

Tamás Horváth
Dept. of Computer Science III
University of Bonn and
Fraunhofer IAIS
Sankt Augustin, Germany
tamas.horvath@iais.fraunhofer.de

Jan Ramon
Dept. of Computer Science
Katholieke Universiteit Leuven
Belgium
janr@cs.kuleuven.be

Stefan Wrobel
Fraunhofer IAIS
Sankt Augustin and
Dept. of Computer Science III
University of Bonn, Germany
stefan.wrobel@iais.fraunhofer.de

Abstract

In recent years there has been an increased interest in frequent pattern discovery in large databases of graph structured objects. While the frequent connected subgraph mining problem for tree datasets can be solved in incremental polynomial time, it becomes intractable for arbitrary graph databases. Existing approaches have therefore resorted to various heuristic strategies and restrictions of the search space, but have not identified a practically relevant tractable graph class beyond trees. In this paper, we define the class of so called *tenuous outerplanar graphs*, a strict generalization of trees, develop a frequent subgraph mining algorithm for tenuous outerplanar graphs that works in incremental polynomial time, and evaluate the algorithm empirically on the NCI molecular graph dataset.

1 Introduction

The discovery of *frequent patterns* in a database is one of the central tasks considered in data mining. In addition to be interesting in their own right, frequent patterns can also be used as features for predictive data mining tasks (see, e.g., [Deshpande *et al.*, 2005]). For a long time, work on frequent pattern discovery has concentrated on relatively simple notions of patterns and elements in the database as they are typically used for the discovery of association rules (simple sets of atomic items). In recent years, however, due to the significance of application areas such as the analysis of chemical molecules or graph structures in the WWW, there has been an increased interest in algorithms that can perform frequent pattern discovery in databases of structured objects such as *trees* or arbitrary *graphs*.

While the frequent pattern problem for trees can be solved in *incremental polynomial time*, i.e., in time polynomial in the *combined size* of the input and the set of frequent tree patterns *so far* computed, the frequent pattern problem for graph structured databases in the general case cannot be solved in *output polynomial time*, i.e., in time polynomial in the *combined size* of the input and the set of *all* frequent patterns. Existing approaches to frequent pattern discovery for graphs have therefore resorted to various heuristic strategies and restrictions of the search space (see, e.g., [Cook and Holder, 1994; Deshpande *et al.*, 2005; Inokuchi *et al.*, 2003; Yan and Han, 2002]), but have not identified a practically relevant tractable graph class beyond trees.

In this paper, we define the class of so called *tenuous outerplanar graphs*, which is the class of planar graphs that can be embedded in the plane in such a way that all of its vertices lie on the outer boundary, i.e. can be reached from the outside without crossing any edges, and which have a fixed limit on the number of inside diagonal edges. This class of graphs is a strict generalization of trees, and is motivated by the kinds of graphs actually found in practical applications. In fact, in one of the popular graph mining data sets, the NCI data set[1], 94.3% of all elements are tenuous outerplanar graphs. We develop an incremental polynomial time algorithm for enumerating frequent tenuous outerplanar graph patterns.

Our approach is based on a canonical string representation of outerplanar graphs which may be of interest in itself, and further algorithmic components for mining frequent biconnected outerplanar graphs and candidate generation in an Apriori style algorithm. To map a pattern to graphs in the database, we define a special notion of *block and bridge preserving* (BBP) subgraph isomorphism, which is motivated by application and complexity considerations, and show that it is decidable in polynomial time for outerplanar graphs. We note that for trees, which form a special class of outerplanar graphs, BBP subgraph isomorphism is equivalent to subtree isomorphism. Thus, BBP subgraph isomorphism generalizes subtree isomorphism to graphs, but is at the same time more specific than subgraph isomorphism. Since in many applications, subgraph isomorphism is a non-adequate matching operator (e.g., when pattern matching is required to preserve certain type of fragments in molecules), by considering BBP subgraph isomorphism we take a first step towards studying the frequent graph mining problem w.r.t. non-standard matching operators as well. Beside complexity results, we present also empirical results which show that the favorable theoretical properties of the algorithm and pattern class also translate into efficient practical performance.

The paper is organized as follows. In Sections 2 and 3, we define the necessary notions and the problem setting for this work, respectively. Section 4 describes our algorithm for mining frequent tenuous outerplanar graphs. Section 5 contains our experimental evaluation and finally, Section 6 concludes and discusses some open problems. Due to space

*A longer version of this paper appeared in the in the *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 197–206, ACM Press, New York, NY, 2006. This short version has also been accepted to the ECML/PKDD Workshop on *Mining and Learning with Graphs*(MLG 2006).

[1] http://cactus.nci.nih.gov/

limitations, proofs are omitted in this short version.

2 Preliminaries

We recall some notions related to graphs [Harary, 1971]. An *undirected graph* is a pair (V, E), where $V \neq \emptyset$ is a finite set of *vertices* and $E \subseteq \{e \subseteq V : |e| = 2\}$ is a set of *edges*. A *labeled undirected graph* is a quadruple (V, E, Σ, λ), where (V, E) is an undirected graph, $\Sigma \neq \emptyset$ is a finite set of *labels* associated with some total order, and $\lambda : V \cup E \rightarrow \Sigma$ is a function assigning a label to each element of $V \cup E$. Unless otherwise stated, in this paper by graphs we always mean *labeled undirected* graphs and denote the set of vertices, the set of edges, and the labeling function of a graph G by $V(G)$, $E(G)$, and λ_G, respectively. Let G and G' be graphs. G' is a *subgraph* of G, if $V(G') \subseteq V(G)$, $E(G') \subseteq E(G)$, and $\lambda_{G'}(x) = \lambda_G(x)$ for every $x \in V(G') \cup E(G')$. For a vertex $v \in V(G)$, $N(v)$ denotes the set of vertices of G connected by an edge with v.

A graph G is *connected* if there is a path between any pair of its vertices; it is *biconnected* if for any two vertices u and v of G, there is a simple cycle containing u and v. A *block* (or *biconnected component*) of a graph is a maximal subgraph that is biconnected. Edges not belonging to blocks are called *bridges*. The definitions imply that the blocks of a graph are pairwise edge disjoint and that the set of bridges forms a forest. For the set of blocks and the forest formed by the bridges of a graph G it holds that their cardinalities are bounded by $|V(G)|$ and they can be enumerated in time $O(|V(G)| + |E(G)|)$ [Tarjan, 1972].

Let G_1 and G_2 be graphs. G_1 and G_2 are *isomorphic*, denoted $G_1 \simeq G_2$, if there is a *bijection* $\varphi : V(G_1) \rightarrow V(G_2)$ such that (i) $\{u, v\} \in E(G_1)$ iff $\{\varphi(u), \varphi(v)\} \in E(G_2)$, (ii) $\lambda_{G_1}(u) = \lambda_{G_2}(\varphi(u))$, (iii) and if $\{u, v\} \in E(G_1)$ then $\lambda_{G_1}(\{u, v\}) = \lambda_{G_2}(\{\varphi(u), \varphi(v)\})$ hold for every $u, v \in V(G_1)$. In this paper, two graphs are considered to be the same if they are isomorphic. G_1 is *subgraph isomorphic* to G_2 if G_1 is isomorphic to a subgraph of G_2. Deciding whether a graph is subgraph isomorphic to another graph is NP-complete, as it generalizes e.g. the Hamiltonian path problem.

Outerplanar Graphs Informally, a graph is *planar* if it can be drawn in the plane in such a way that no two edges intersect except at a vertex in common. An *outerplanar graph* is a planar graph which can be embedded in the plane in such a way that all of its vertices lie on the boundary of the outer face. Throughout this work we consider connected outerplanar graphs and denote the set of connected outerplanar graphs over an alphabet Σ by $\mathcal{O}_\Sigma$. Clearly, trees are outerplanar graphs and hence, a graph is outerplanar iff each of its blocks is outerplanar [Harary, 1971]. Furthermore, as the blocks of a graph can be computed in linear time [Tarjan, 1972] and outerplanarity of a block can be decided also in linear time [Lingas, 1989; Mitchell, 1979], one can decide in linear time whether a graph is outerplanar.

A biconnected outerplanar graph G with n vertices contains at most $2n - 3$ edges and has a unique Hamiltonian cycle which bounds the outer face of a planar embedding of G [Harary, 1971]. This unique Hamiltonian cycle can be computed efficiently [Lingas, 1989]. Thus, G can be considered as an n-polygon with at most $n - 3$ non-crossing diagonals. Below we state a bound for the number of cycles of G. Due to space limitation, we omit the proof.

Proposition 1 *A biconnected outerplanar graph with d diagonals has at most 2^{d+1} cycles.*

Given outerplanar graphs G and H, deciding whether H is subgraph isomorphic to G is an NP-complete problem. This follows from the fact that outerplanar graphs generalize forests and deciding whether a forest is subgraph isomorphic to a tree is NP-complete [Garey and Johnson, 1979]. The following stronger negative result is shown in [Syslo, 1982].

Theorem 2 *Deciding whether a connected outerplanar graph H is subgraph isomorphic to a biconnected outerplanar graph G is NP-complete.*

If, however, H is also biconnected, the following positive result holds [Lingas, 1989].

Theorem 3 *Let G, H be biconnected outerplanar graphs. Then one can decide in time $O\left(|V(H)| \cdot |V(G)|^2\right)$ whether H is subgraph isomorphic to G.*

For the special case of trees, the following positive result holds [Matula, 1978].[2]

Theorem 4 *The problem whether a tree H is subgraph isomorphic to a tree G can be decided in time $O\left(|V(H)|^{1.5} \cdot |V(G)|\right)$.*

3 The Problem Setting

In this section we define the frequent subgraph mining problem for a practically relevant class of outerplanar graphs with respect to a matching operator that preserves the pattern graph's bridge and block structure. To define the mining problem, we need the notions of tenuous outerplanar graphs and BBP subgraph isomorphism.

Tenuous Outerplanar Graphs Let $d \geq 0$ be some integer. A *d-tenuous* outerplanar graph G is an outerplanar graph such that each block of G has at most d diagonals. For an alphabet Σ and integer $d \geq 0$, $\mathcal{O}_\Sigma^d$ denotes the set of connected d-tenuous outerplanar graphs labeled by the elements of Σ. The class of d-tenuous outerplanar graphs forms a practically relevant graph class e.g. in chemoinformatics. As an example, out of the 250251 pharmacological molecules in the NCI dataset, 236180 (i.e., 94.3%) compounds have an outerplanar molecular graph. Furthermore, among the outerplanar compounds, there is no molecular graph having a block with more than 11 diagonals. In fact, there is only one compound containing a block with 11 diagonals; 236083 (i.e., 99.99%) compounds among the outerplanar graphs have at most 5 diagonals per block.

BBP Subgraph Isomorphism We continue our problem definition by introducing a matching operator between outerplanar graphs. Let $G, H \in \mathcal{O}_\Sigma$. A *bridge and block preserving* (BBP) subgraph isomorphism from H to G, denoted $H \preccurlyeq_{BBP} G$, is a subgraph isomorphism from H to G mapping (i) the set of bridges of H to the set of bridges of G and (ii) different blocks of H to different blocks of G. Notice that for trees, which are special outerplanar graphs (i.e., block-free), BBP subgraph isomorphism is equivalent

[2] The bound in Theorem 4 is improved by a log factor in [Shamir and Tsur, 1999]. For the sake of simplicity, we generalize the algorithm in [Matula, 1978] to outerplanar graphs in the long version of this paper. We note that the complexity of our algorithm can also be improved using the idea of [Shamir and Tsur, 1999].

to the ordinary subtree isomorphism. Thus, BBP subgraph isomorphism can be considered as a generalization of subtree isomorphism to outerplanar graphs which is more specific than ordinary subgraph isomorphism.

Besides complexity reasons raised by Theorem 2, the use of BBP subgraph isomorphism as matching operator is motivated by recent results in chemoinformatics which indicate that more powerful predictors can be obtained by considering matching operators that map certain fragments of the pattern molecule to certain fragments of the target molecule. One natural step towards this direction is to require that only ring structures (i.e., blocks) can be mapped to ring structures and that edge disjoint ring structures are mapped to edge disjoint ring structures.

The FTOSM Problem Using the above notions, we define the *frequent d-tenuous outerplanar subgraph mining problem* (FTOSM) as follows: *Given* (i) an alphabet Σ, (ii) a finite set $\mathcal{D} \subseteq \mathcal{O}_\Sigma^d$ of *transactions* for some integer $d \geq 0$, and (iii) an integer threshold $t > 0$, *enumerate* the set of all connected d-tenuous outerplanar graphs in $\mathcal{O}_\Sigma^d$ that match at least t graphs in $\mathcal{D}$ w.r.t. BBP subgraph isomorphism, i.e., enumerate the set

$$\mathcal{F}_{\Sigma,d}^t(\mathcal{D}) = \{H \in \mathcal{O}_\Sigma^d : \Pi_t(\mathcal{D}, H)\} \ , \qquad (1)$$

where $\Pi_t(\mathcal{D}, H)$ is the *frequency property* defined by

$$\Pi_t(\mathcal{D}, H) = |\{G \in \mathcal{D} : H \preccurlyeq_{BBP} G\}| \geq t \ . \qquad (2)$$

By definition, $\mathcal{F}_{\Sigma,d}^t(\mathcal{D})$ does not contain isomorphic graphs. Furthermore, it is closed downwards w.r.t. BBP subgraph isomorphism, i.e., $G_1 \in \mathcal{F}_{\Sigma,d}^t(\mathcal{D})$ whenever $G_2 \in \mathcal{F}_{\Sigma,d}^t(\mathcal{D})$ and $G_1 \preccurlyeq_{BBP} G_2$. Given $\mathcal{D}$ and t, we call a graph H satisfying (2) *t-frequent*.

The *parameters* of the FTOSM problem are the cardinality of the transaction dataset (i.e., $|\mathcal{D}|$) and the size of the largest graph in $\mathcal{D}$ (i.e., $\max\{|V(G)| : G \in \mathcal{D}\}$). Since d is usually small, it is assumed to be a *constant*. Note that the cardinality of $\mathcal{F}_{\Sigma,d}^t(\mathcal{D})$ can be exponential in the above parameters of $\mathcal{D}$. Clearly, in such cases it is impossible to enumerate $\mathcal{F}_{\Sigma,d}^t(\mathcal{D})$ in time polynomial in the parameters of $\mathcal{D}$. We therefore ask whether the FTOSM problem can be solved in *incremental polynomial time* (see, e.g., [Johnson *et al.*, 1988]), that is, whether there exists an enumeration algorithm listing the first k elements of $\mathcal{F}_{\Sigma,d}^t(\mathcal{D})$ in time polynomial in the *combined size* of $\mathcal{D}$ and the set of these k elements for every $k = 1, \ldots, |\mathcal{F}_{\Sigma,d}^t(\mathcal{D})|$.

We note that in the literature (see, e.g., [Johnson *et al.*, 1988]) one usually considers also the notion of *output polynomial time* (or *polynomial total time*) complexity for enumeration algorithms. Algorithms in this more liberal class are required to enumerate a set S in the combined size of the input and the *entire* set S. Thus, in contrast to incremental polynomial time, an output polynomial time algorithm may have in worst-case a delay time exponential in the size of the input before printing the kth element for some $k \geq 1$.

Although several algorithms mining frequent connected subgraphs from datasets of arbitrary graphs w.r.t. subgraph isomorphism have demonstrated their performance empirically, we note that this general problem cannot be solved in output polynomial time, unless P = NP. On the other hand, the frequent graph mining problem is solvable in incremental polynomial time when the graphs in the dataset are restricted to forests and the patterns to trees. This follows e.g. from the results in [Chi *et al.*, 2005b]. Since tenuous outerplanar graphs form a practically relevant graph class that naturally generalizes trees, by considering the FTOSM problem we take a step towards going beyond trees in frequent subgraph mining.

Algorithm 1 FREQUENTOUTERPLANARGRAPHS

Require: $\mathcal{D} \subseteq \mathcal{O}_\Sigma^d$ for some alphabet Σ and integer $d \geq 0$, and integer $t > 0$
Ensure: $\mathcal{F}_{\Sigma,d}^t(\mathcal{D})$ defined in Eq. (1)

1: $\mathcal{L}_1 = \mathcal{F}_v \cup \mathcal{F}_b$, where
$\mathcal{F}_v = \{H \in \mathcal{O}_\Sigma : |V(H)| = 1 \wedge \Pi_t(\mathcal{D}, H)\}$
$\mathcal{F}_b = \{H \in \mathcal{O}_\Sigma^d : H \text{ is biconnected} \wedge \Pi_t(\mathcal{D}, H)\}$
2: $\mathcal{L}_2 = \mathcal{F}_e \cup \mathcal{F}_{bb} \cup \mathcal{F}_{be}$, where
$\mathcal{F}_e = \{H \in \mathcal{O}_\Sigma : |E(H)| = 1 \wedge \Pi_t(\mathcal{D}, H)\}$
$\mathcal{F}_{bb} = \{H \in G_1 \bowtie G_2 : G_1, G_2 \in \mathcal{F}_b \wedge \Pi_t(\mathcal{D}, H)\}$
$\mathcal{F}_{be} = \{H \in G_1 \bowtie G_2 : G_1 \in \mathcal{F}_b \wedge G_2 \in \mathcal{F}_e \wedge \Pi_t(\mathcal{D}, H)\}$
3: $k = 2$
4: **while** $\mathcal{L}_k \neq \emptyset$ **do**
5: $k = k + 1$
6: $\mathcal{C}_k = \text{GENERATECANDIDATES}(\mathcal{L}_{k-1})$
7: $\mathcal{L}_k = \{H \in \mathcal{C}_k : \Pi_t(\mathcal{D}, H)\}$
8: **endwhile**
9: **return** $\cup_{i=1}^k \mathcal{L}_i$

4 The Mining Algorithm

In this section we present Algorithm 1, an Apriori-like [Agrawal *et al.*, 1996] algorithm, that solves the FTOSM problem in incremental polynomial time. For a set $\mathcal{D} \subseteq \mathcal{O}_\Sigma^d$ and integer $t \geq 0$, the algorithm computes iteratively the set of t-frequent k-patterns from the set of t-frequent $(k-1)$-patterns. A *k-pattern* is a graph $G \in \mathcal{O}_\Sigma^d$ such that the sum of the number of blocks of G and the number of vertices of G not belonging to any block is k.

In step 1 of the algorithm, we first compute the set of t-frequent 1-patterns, that is, the set of t-frequent graphs consisting of either a single vertex or a single block. The first set, denoted by $\mathcal{F}_v$ in step 1, can be computed in linear time. The second set, denoted $\mathcal{F}_b$, can be computed in time polynomial in the parameters of $\mathcal{D}$; an efficient Apriori-based algorithm for this problem is presented in Section 4.2.

In step 2 of the algorithm, we then compute the set of t-frequent 2-patterns, i.e., the set of graphs in $\mathcal{O}_\Sigma^d$ consisting of either (1) a single edge or (2) two blocks having a common vertex or (3) a block and a bridge edge having a common vertex. We denote the corresponding three sets in step 2 by $\mathcal{F}_e$, $\mathcal{F}_{bb}$, and $\mathcal{F}_{be}$, respectively. In the definitions of $\mathcal{F}_{bb}$ and $\mathcal{F}_{be}$, $G_1 \bowtie G_2$ denotes the set of graphs that can be obtained from the union of G_1 and G_2 by contracting[3] a vertex from G_1 with a vertex from G_2 that have the same label. Clearly, $G_1 \bowtie G_2 \subseteq \mathcal{O}_\Sigma^d$ for every $G_1, G_2 \in \mathcal{O}_\Sigma^d$. The set $\mathcal{F}_e$ of t-frequent edges can be computed in linear time. Since the cardinalities of both $\mathcal{F}_{bb}$ and $\mathcal{F}_{be}$ are polynomial in the parameters of $\mathcal{D}$, and BBP subgraph isomorphism

[3]The contraction of the vertices u and v of a graph G is the graph obtained from G by introducing a new vertex w, connecting w with every vertex in $N(u) \cup N(v)$, and removing u and v, as well as the edges adjacent to them.

between outerplanar graphs can be decided in polynomial time by the result of Section 4.4 below, it follows that both $\mathcal{F}_{bb}$ and $\mathcal{F}_{be}$ and hence, the set $\mathcal{L}_2$ of t-frequent 2-patterns can be computed in time polynomial in the parameters of $\mathcal{D}$.

In the loop 4–8, we compute the set of t-frequent k-patterns for $k \geq 3$ in a way similar to the Apriori algorithm [Agrawal *et al.*, 1996]. The crucial steps of the loop are the generation of candidate k-patterns from the set of t-frequent $(k-1)$-patterns (step 6) and the decision of t-frequency of the candidate patterns (step 7). In Sections 4.3 and 4.4 below we describe these steps in detail.

Putting together the results given in Theorems 7 – 10 stated in Sections 4.1 – 4.4, respectively, we can formulate the main result of this paper:

Theorem 5 *Algorithm 1 is correct and solves the FTOSM problem in incremental polynomial time.*

Before going into the technical details in Sections 4.1 – 4.4, we first describe a transformation on outerplanar graphs by means of block contraction that is used in different steps of the mining algorithm. More precisely, for a graph $G \in \mathcal{O}_\Sigma$, let $\widetilde{G}$ denote the graph over the alphabet $\Sigma \cup \{\#\}$ derived from G by the following transformation: For each block B in G, (i) introduce a new vertex v_B and label it by $\#$, (ii) remove each edge belonging to B, and (iii) for every vertex v of B, connect v with v_B by an edge labeled by $\#$, if v is adjacent to a bridge or to another block of G; otherwise remove v. In the following proposition we state some basic properties of $\widetilde{G}$.

Proposition 6 *Let $G \in \mathcal{O}_\Sigma$. Then*

(i) $|V(\widetilde{G})| = 1$ *iff* $|V(G)| = 1$ *or G is biconnected,*

(ii) for every $e \in E(\widetilde{G})$, at most one vertex of e is labeled by $\#$, and

(iii) $\widetilde{G}$ is a free tree.

Since $\widetilde{G}$ is a tree, we call it the *block and bridge tree (BB-tree)* of G.

4.1 Canonical String Representation

One time consuming step of mining frequent d-tenuous outerplanar graphs is to test whether a particular graph $H \in \mathcal{O}_\Sigma^d$ belongs to some subset S of $\mathcal{O}_\Sigma^d$. To apply advanced data structures that allow fast search in large subsets of $\mathcal{O}_\Sigma^d$, we need to define a total order on $\mathcal{O}_\Sigma^d$. Similarly to many other frequent graph mining algorithms, we solve this problem by assigning a *canonical string* to each element of $\mathcal{O}_\Sigma$ such that (i) two graphs have the same canonical string iff they are isomorphic and (ii) for every $G \in \mathcal{O}_\Sigma$, the canonical string of G can be computed efficiently. Using some canonical string representation satisfying the above properties, a total order on $\mathcal{O}_\Sigma$ and thus, on $\mathcal{O}_\Sigma^d$ as well, can be defined by some total order (e.g. lexicographic) on the set of strings assigned to the elements of $\mathcal{O}_\Sigma$. Furthermore, property (i) allows one to decide isomorphism between two outerplanar graphs by comparing their canonical strings.

Although the canonical string representation for outerplanar graphs may be of some interest in itself, due to space limitations we omit its definition which is based on the BB-tree $\widetilde{G}$ of G. By (iii) of Proposition 6, $\widetilde{G}$ is a free tree. Utilizing this property, we can generalize the depth-first canonical representation for free trees (see, e.g., [Chi *et al.*, 2005a]) to outerplanar graphs, and state the following result:

Algorithm 2 FREQUENTBICONNECTEDGRAPHS

Require: $\mathcal{D} \subseteq \mathcal{O}_\Sigma^d$ for some alphabet Σ and integer $d \geq 0$, and integer $t > 0$
Ensure: $\mathcal{F}_b$ defined in step 1 of Algorithm 1

1: **let** $\mathcal{L}_0 \subseteq \mathcal{O}_\Sigma^0$ be the set of t-frequent cycles in $\mathcal{D}$
2: **for** $k = 1$ **to** d **do**
3: **let** $\mathcal{C}_k \subseteq \mathcal{O}_\Sigma^k \setminus \mathcal{O}_\Sigma^{k-1}$ be the set of biconnected graphs H such that $H \ominus \Delta \in \mathcal{L}_{k-1}$ for every diagonal Δ of H
4: $\mathcal{L}_k = \{H \in \mathcal{C}_k : \Pi_t(\mathcal{D}, H)\}$
5: **endfor**
6: **return** $\bigcup_{k=0}^{d} \mathcal{L}_k$

Theorem 7 *A canonical string representation of a graph in $\mathcal{O}_\Sigma$ with n vertices can be computed in time $O\left(n^2 \log n\right)$.*

4.2 Mining Frequent Biconnected Graphs

In this section we present Algorithm 2, an Apriori-like algorithm, that computes the set $\mathcal{F}_b$ of t-frequent d-tenuous biconnected graphs used in step 1 of Algorithm 1. Since d is constant, Algorithm 2 runs in time polynomial in the parameters of $\mathcal{D}$.

In step 1 of Algorithm 2, we first compute the set $\mathcal{L}_0$ of t-frequent cycles as follows: We list the cycles of G for every $G \in \mathcal{D}$ and count their frequencies. Proposition 1 in Section 2 implies that the number of cycles of a d-tenuous outerplanar graph G is bounded by $O\left(|V(G)|\right)$ if d is assumed to be constant. Furthermore, from [Read and Tarjan, 1975; Tarjan, 1972] it follows that the cycles of a graph can be listed with linear delay. Since isomorphism between cycles can be decided efficiently, these results together imply that $\mathcal{L}_0$ can be computed in time polynomial in the parameters of $\mathcal{D}$.

In loop 2–5 of Algorithm 2, we compute the sets of t-frequent biconnected graphs containing k diagonals for every $k = 1, \ldots, d$. In particular, in step 3 we compute the set $\mathcal{C}_k$ of candidate biconnected graphs $H \in \mathcal{O}_\Sigma^k$ satisfying the following conditions: H has exactly k diagonals and the removal of any diagonal from H, denoted by $\ominus$ in step 3, results in a t-frequent biconnected graph. Putting the above results together, we can state the following theorem. (We omit the proof in this short version.)

Theorem 8 *Algorithm 2 is correct and computes the set of t-frequent d-tenuous biconnected outerplanar graphs in time polynomial in the parameters of $\mathcal{D}$.*

4.3 Candidate Generation

In step 6 of Algorithm 1, we generate the set of candidate k-patterns. In this section we give Algorithm 3, a generalization of the candidate generation algorithm for free trees described in [Chi *et al.*, 2005b], that computes the set of candidate k-patterns from the set of frequent $(k-1)$-patterns. Applying the candidate generation principle of the Apriori algorithm [Agrawal *et al.*, 1996], each candidate is obtained by joining two frequent $(k-1)$-patterns that have an isomorphic $(k-2)$-pattern core.

In the outer loop 2–12 of the algorithm, we consider each possible pair G_1, G_2 of frequent $(k-1)$-patterns, and in loop 3–11, each pair g_1 and g_2 of *leaf* subgraphs of G_1 and G_2, respectively. By a leaf subgraph of a k-pattern H for $k \geq 2$ we mean the subgraph of H represented by a leaf of the BB-tree $\widetilde{H}$. If G_1 and G_2 are the same graphs then,

Algorithm 3 GENERATECANDIDATES

Require: set $\mathcal{L}_{k-1}$ of frequent $k-1$-patterns for some $k > 2$

Ensure: set $\mathcal{C}_k$ of candidate k-patterns

1: $\mathcal{C}_k = \emptyset$
2: **forall** $G_1, G_2 \in \mathcal{L}_{k-1}$ **do**
3: **forall** $g_1 \in \text{Leaf}(G_1)$ and $g_2 \in \text{Leaf}(G_2)$ **do**
4: **if** $G_1 \ominus g_1 \simeq G_2 \ominus g_2$ **then**
5: **forall** $g_1' \in \text{Leaf}(G_1 \ominus g_1)$ **do**
6: **if** g_2 is attachable to g_1' consistently with G_2 **then**
7: attach g_2 in G_1 to g_1' consistently with G_2 and denote the obtained graph by C
8: **if** g_1, g_2 have the top two string encodings in C, $C \notin \mathcal{C}_k$, and $C \ominus g \in \mathcal{L}_{k-1}$ for every $g \in \text{Leaf}(C)$
9: **then** add C to $\mathcal{C}_k$
10: **endfor**
11: **endfor**
12: **endfor**
13: **return** $\mathcal{C}_k$

for completeness, we consider also the case when g_1 and g_2 are isomorphic leaf subgraphs. We remove g_1 and g_2 from G_1 and G_2, respectively, denoted by $\ominus$ in the algorithm, and check whether the obtained graphs G_1' and G_2' are isomorphic (step 4). The removal of a biconnected component means the deletion of each of its edges and vertices except the distinguished vertex which is adjacent to a bridge or to another block.

If G_1' and G_2' are isomorphic then we consider every leaf subgraph g_1' of G_1' (loop 5–10) and check whether g_2 can be attached to g_1' in G_1 consistently with G_2 (step 6). More precisely, let g_2' be a block or a vertex not belonging to a block in G_2 such that g_2 is hanging from g_2', i.e., the only edge adjacent to g_2 is adjacent also to g_2'. We say that g_2 can be attached to g_1' in G_1 consistently with G_2 if g_1' is isomorphic to g_2'. Thus, if the condition in step 6 holds then we attach g_2 to g_1' consistently with G_2 and denote the obtained graph by C (step 7).

Notice that C can be generated in many different ways, depending on the particular choice of g_1 and g_2. To reduce the amount of unnecessary computation, we consider only those pairs which are among the top leaf subgraphs of C, i.e., which have the top two string encodings w.r.t. a center of $\tilde{C}$. By definition, a vertex representing a leaf subgraph of C is always a leaf in $\tilde{C}$. If this condition holds then we add C to the set of candidates in step 9 if for every leaf subgraph g of C, the $(k-1)$-pattern obtained from C by removing g is frequent (see step 8). We omit the proof of the following theorem.

Theorem 9 *Let $\mathcal{C}_k$ be the output of Algorithm 3 and $\mathcal{L}_k$ the set of frequent k-patterns for any $k > 2$. Then $\mathcal{L}_k \subseteq \mathcal{C}_k$, the cardinality of $\mathcal{C}_k$ is polynomial in the cardinality of $\mathcal{L}_{k-1}$, and $\mathcal{C}_k$ can be computed in time polynomial in the size of $\mathcal{L}_{k-1}$.*

4.4 BBP Subgraph Isomorphism

Algorithms 1 and 2 contain the steps of deciding whether a candidate pattern $H \in \mathcal{O}_\Sigma^d$ is t-frequent, i.e., whether it is BBP subgraph isomorphic to at least t graphs in $\mathcal{D}$. While subgraph isomorphism between outerplanar graphs is NP-complete even for very restricted cases (see Theorem 2), Theorem 10, the main result of this section, states that BBP subgraph isomorphism can be decided efficiently between outerplanar graphs if the pattern graph H is connected. The connectivity is necessary, as otherwise the problem would generalize the NP-complete subforest isomorphism problem [Garey and Johnson, 1979]. We note that the result of Theorem 10 generalizes the positive result on subtree isomorphism given in Theorem 4 and may thus be of some interest in itself.

Theorem 10 *Let $G, H \in \mathcal{O}_\Sigma$ such that H is connected. Then $H \preccurlyeq_{BBP} G$ can be decided in polynomial time.*

Due to space limitations, we omit the proof of the above theorem. We only note that the algorithm first computes the BB-trees of the input graphs G and H, and then combines the subgraph isomorphism algorithms between labeled trees (generalization of [Matula, 1978]) and labeled biconnected outerplanar graphs (generalization of [Lingas, 1989]).

5 Experimental Evaluation

In our experiments, we used the NCI dataset consisting of 250251 chemical compounds. For our work, it was important to recognize that 236180 (i.e., 94.3%) of these compounds have outerplanar molecular graph. Thus, outerplanar graphs form a practically relevant class of graphs. Among the outerplanar molecular graphs, there are 21963 trees (i.e., 8.8% of the outerplanar subset). In the experiments, we have removed the non-outerplanar graphs from the dataset. Altogether, the outerplanar molecules contain 423378 blocks, with up to 11 diagonals per block. However, 236083 (i.e., 99.99%) of the outerplanar molecular graphs have at most 5 diagonals per block. This empirical observation validates our approach to assume the number of diagonals to be constant.

The database contains a wide variety of structures, and a low relative frequency threshold is needed to mine a significant number of patterns. E.g. though there are 15426 pairwise non-isomorphic cycles in the database, only a few of them are really frequent; the only one above 10% is the benzene ring with frequency 66%.

Our results are given in Table 1. It shows the number of candidate (#C) and frequent (#FP) k-patterns discovered for $k = 1, \ldots, 15$, as well as the runtime (T) in seconds for the computation and evaluation of the candidates using the frequency thresholds 10%, 5%, 2% and 1%. As expected, the number and the size of the discovered patterns is much larger when the frequency threshold is lower. Even though the embeddings of $(k-1)$ patterns are computed (again) in level k, the time needed to complete one level does not necessarily increase with k. It is interesting to note that after the number of frequent k-patterns drops a bit when k gets larger then 8, this number again increases when k exceeds 12, and the number of frequent patterns gets close to the number of candidate patterns. This is because this particular dataset contains large subsets with molecules sharing large biconnected structures (such as the HIV active substance dataset). The time needed for candidate generation is always smaller than 1% of the total time. The time needed for coverage testing per pattern depends on how much structure these patterns share. If the number of patterns is large, the time needed per pattern is usually lower.

One can make several conclusions. First, our algorithm can mine an expressive class of molecular patterns from a relatively large database. Although the presented

Table 1: Number of patterns (#C), number of frequent patterns (#FP), and runtime in seconds for candidate generation and evaluation (T) with frequency thresholds 10%, 5%, 2%, and 1%

size	10%			5%			2%			1%		
(k)	#C	#FP	T	#C	#FP	T	#C	#FP	T	#C	#FP	T
1	86	7	107	144	11	169	582	25	380	2196	55	824
2	74	16	446	216	24	570	1332	61	1118	6208	174	2554
3	139	41	1133	234	74	1393	510	170	2123	1516	659	5653
4	133	77	1232	266	154	2038	642	356	4079	2554	1776	11899
5	139	91	1071	319	222	2268	909	644	5603	4550	3886	20411
6	107	72	754	332	252	1847	1212	918	6105	7314	6490	28811
7	61	41	472	295	195	1168	1266	990	4964	10165	9058	34967
8	37	25	354	182	137	741	1086	893	3384	11479	10396	36391
9	20	13	205	137	116	602	956	803	2282	11129	10194	31721
10	8	5	130	131	119	594	828	700	1635	9370	8623	23412
11	0	0	0	131	117	565	697	604	1360	7276	6818	15530
12	0	0	0	115	107	536	707	665	1483	5533	5184	9345
13	0	0	0	78	64	412	1027	1022	2017	4395	4145	5252
14	0	0	0	27	21	250	1702	1700	2858	4303	4194	3707
15	0	0	0	4	3	89	2725	2715	3957	5422	5376	4089

experiments happened entirely in memory (taking about 600Mb), our approach does not depend on storing intermediate results in memory between the different passes over the database. This means that we could also perform this algorithm with a database on disk. In our application e.g., this would bring an overhead of about 15 seconds per pass over the database. Second, we can conclude that the complexity of the coverage testing scales well as the pattern size grows, as predicted by theory. In this application, due to the implementation exploiting shared structure among patterns, the time needed for evaluation per pattern does not even depend in a clear systematic way on the pattern size.

6 Conclusion and Open Problems

We have defined the FTOSM problem motivated by chemical datasets and presented an Apriori-based algorithm solving this enumeration problem in incremental-polynomial time. To the best of our knowledge, no fragment of the frequent subgraph mining problem *beyond trees* has so far been identified, for which the problem can be solved in incremental polynomial time. Our algorithm is based on a canonical string representation of outerplanar graphs and further algorithmic components for mining frequent biconnected outerplanar graphs and candidate generation in an Apriori style algorithm. Motivated by application and complexity considerations, we introduced a special kind of subgraph isomorphism which generalizes subtree isomorphism but is at the same time more specific than ordinary subgraph isomorphism, and which is decidable in polynomial time for outerplanar graphs. We presented also empirical results with a large dataset indicating the effective practical performance of our algorithm. We believe that the identification of tractable practical fragments of the frequent subgraph mining problem is an important challenge for the data mining community.

Besides working on optimization of the algorithm, e.g., on improving the time complexity of the coverage testing, it is natural to ask whether the positive result of this paper can be generalized to arbitrary outerplanar graphs. Notice that our algorithm exploits the constant bound on the number of diagonals only in the computation of the set $\mathcal{F}_b$ of frequent biconnected graphs in step 1 of Algorithm 1. Therefore, to generalize the result of this paper to arbitrary outerplanar graphs, it is sufficient to consider the following special problem: *Given* a finite set $\mathcal{D} \subseteq \mathcal{O}_\Sigma$ of biconnected outerplanar graphs and a non-negative integer t, compute the set of t-frequent patterns in $\mathcal{D}$ w.r.t. BBP subgraph isomorphism. Notice that this problem definition implicitly requires t-frequent patterns to be biconnected because by definition, there is no BBP subgraph isomorphism from a non-biconnected graph to a biconnected outerplanar graph. We do not know whether this special problem can be solved in incremental or at least in output polynomial time.

Acknowledgments

Tamás Horváth and Stefan Wrobel were partially supported by the DFG project (WR 40/2-1) *Hybride Methoden und Systemarchitekturen für heterogene Informationsräume*. Jan Ramon is a post-doctoral fellow of the Fund for Scientific Research (FWO) of Flanders.

References

[Agrawal *et al.*, 1996] R. Agrawal, H. Mannila, R. Srikant, H. Toivonen, and A. I. Verkamo. Fast discovery of association rules. In *Advances in Knowledge Discovery and Data Mining*, pp. 307–328. AAAI/MIT Press, 1996.

[Chi *et al.*, 2005a] Y. Chi, R. R. Muntz, S. Nijssen, and J. N. Kok. Frequent subtree mining - An overview. *Fundamenta Informaticae*, 66(1-2):161–198, 2005.

[Chi *et al.*, 2005b] Y. Chi, Y. Yang, and R. R. Muntz. Canonical forms for labelled trees and their applications in frequent subtree mining. *Knowledge and Information Systems*, 8(2):203–234, 2005.

[Cook and Holder, 1994] D. J. Cook and L. B. Holder. Substructure discovery using minimum description length and background knowledge. *J. of Artificial Intelligence Research*, 1:231–255, 1994.

[Deshpande *et al.*, 2005] M. Deshpande, M. Kuramochi, N. Wale, and G. Karypis. Frequent substructure-based approaches for classifying chemical compounds. *IEEE Transactions on Knowledge and Data Engineering*, 17(8):1036–1050, 2005.

[Garey and Johnson, 1979] M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to NP-Completeness*. Freeman, San Francisco, CA, 1979.

[Harary, 1971] F. Harary. *Graph Theory*. Addison–Wesley, Reading, Massachusetts, 1971.

[Inokuchi *et al.*, 2003] A. Inokuchi, T. Washio, and H. Motoda. Complete mining of frequent patterns from graphs: Mining graph data. *Machine Learning*, 50(3):321–354, 2003.

[Johnson *et al.*, 1988] D. S. Johnson, M. Yannakakis, and C. H. Papadimitriou. On generating all maximal independent sets. *Information Processing Letters*, 27(3):119–123, 1988.

[Lingas, 1989] A. Lingas. Subgraph isomorphism for biconnected outerplanar graphs in cubic time. *Theoretical Computer Science*, 63:295–302, 1989.

[Matula, 1978] D. W. Matula. Subtree isomorphism in $O(n^{\frac{5}{2}})$. *Annals of Discrete Mathematics*, 2:91–106, 1978.

[Mitchell, 1979] S. L. Mitchell. Linear algorithms to recognize outerplanar and maximal outerplanar graphs. *Information Processing Letters*, 9(5):229–232, 1979.

[Read and Tarjan, 1975] R. C. Read and R. E. Tarjan. Bounds on backtrack algorithms for listing cycles, paths, and spanning trees. *Networks*, 5(3):237–252, 1975.

[Shamir and Tsur, 1999] R. Shamir and D. Tsur. Faster subtree isomorphism. *J. of Algorithms*, 33(2):267–280, 1999.

[Syslo, 1982] M. M. Sysło. The subgraph isomorphism problem for outerplanar graphs. *Theoretical Computer Science*, 17:91–97, 1982.

[Tarjan, 1972] R. E. Tarjan. Depth first search and linear graph algorithms. *SIAM J. on Computing*, 1(2):146–160, 1972.

[Yan and Han, 2002] X. Yan and J. Han. gSpan: Graph-based substructure pattern mining. In *Proc. of the 2002 IEEE Int. Conference on Data Mining (ICDM)*, pp. 721–724, IEEE Computer Society, 2002.

Semantic Network Analysis of Ontologies

Bettina Hoser[†], **Andreas Hotho**[‡], **Robert Jäschke**[‡,*], **Christoph Schmitz**[‡], **Gerd Stumme**[‡,*]
[†] Chair of Informationservices and Electronic Markets, Universität Karlsruhe
[‡] Knowledge & Data Engineering Group, Universität Kassel
[*] Research Center L3S, Hannover

Abstract

A key argument for modeling knowledge in ontologies is the easy re-use and re-engineering of the knowledge. However, current ontology engineering tools provide only basic functionalities for analyzing ontologies. Since ontologies can be considered as graphs, graph analysis techniques are a suitable answer for this need. Graph analysis has been performed by sociologists for over 60 years, and resulted in the vivid research area of Social Network Analysis (SNA). While social network structures currently receive high attention in the Semantic Web community, there are only very few SNA applications, and virtually none for analyzing the structure of ontologies.

We illustrate the benefits of applying SNA to ontologies and the Semantic Web, and discuss which research topics arise on the edge between the two areas. In particular, we discuss how different notions of centrality describe the core content and structure of an ontology. From the rather simple notion of degree centrality over betweenness centrality to the more complex eigenvector centrality, we illustrate the insights these measures provide on two ontologies, which are different in purpose, scope, and size.

1 Introduction

A key argument for modeling knowledge in ontologies is the easy re-use and re-engineering of the knowledge. However, beside consistency checking, current ontology engineering tools provide only basic functionalities for analyzing ontologies. Since ontologies can be considered as (labeled, directed) graphs, graph analysis techniques are a promising tool. Sociologists have performed graph analysis since for over 60 years. In the late 1970ies, Social Network Analysis (SNA) emerged as a research area out of this work. Its aim is to analyze the structures of social communities. Typical applications include the analysis of relationships like friendship, communication patterns (e. g., phone call graphs), and the distribution of attendants over several events. While social structures are currently a steeply rising topic within the Semantic Web community (e. g., friend-of-a-friend networks,[1] social tagging systems like del.icio.us.org or www.bibsonomy.org, or semantics-based P2P networks [22]), Social Network *Analysis* has only been applied marginally up to now on ontologies and the Semantic Web.

In this paper, we will discuss the use of SNA for analyzing ontologies and the Semantic Web. While the SNA community has already discovered the internet and the Web as fruitful application domains for their techniques a while ago (e. g., analysing the link structure of the internet [17], and email traffic [18; 23; 26]), SNA applications for the Semantic Web are only emerging slowly. We advocate here a systematic development of *Semantic Network Analyis (SemNA)*, as the adoption of SNA to ontologies and the Semantic Web. In this paper, we show that the application of both basic and advanced SNA techniques to ontologies provide a powerful tool for analyzing the structure of the ontology. We adapt SNA tools to ontology analysis, and discuss the findings. In particular, we discuss how different notions of centrality describe the core content and structure of an ontology. From the rather simple notion of degree centrality over betweenness centrality to the more complex eigenvector centrality based on Hermitian matrices, we illustrate the insights these measures provide on two ontologies, which are different in purpose, scope, and size. The results may be used for selecting the right ontology for a specific application, as well as for re-engineering ontologies.

SemNA is a sub-area of Semantic Web Mining [4]. that addresses the mining of the Semantic Web. To this end, we consider ontologies as (both vertex- and edge-)labeled, directed graphs. As we will discuss below, the existence of different types of nodes and edges (which are reflected in the labels) is a problem for standard SNA approaches. We will discuss solutions for this problem. In this paper, we present two selected applications, and discuss the use of different SNA techniques for analyzing ontologies. The examples will illustrate the deep insights we were able to gain from the two ontologies.

Testcases: SWRC and SUMO ontologies.

The SWRC ontology[2] provides a vocabulary about publications, authors, academic staff and the like. It consists of 54 concepts and 70 relations. Figure 1 shows a graphical representation of the ontology. Rectangles represent concepts, relations are shown as rounded boxes.

We selected the SWRC ontology as our first example, as it is a handy size, and as we know its structure rather well, since some of the authors have contributed to its construction. We are thus able to validate the resulting SNA findings (which were computed independently by the non-ontology author) with our insight in the history of the SWRC ontology. The promising results (which were also

[1] http://www.foaf-project.org/

[2] http://ontobroker.semanticweb.org/ontologies/swrc-onto-2001-12-11.oxml

surprising for the authors) motivated us to consider a larger ontology, the SUMO ontology, where we only knew about its general purpose, but no details about its structure nor its content.

The aim of the Suggested Upper Merged Ontology (SUMO)[3] is to express the most basic and universal concepts for creating a framework for merging ontologies of different domains. With its 630 concepts and 236 relations, SUMO is significantly larger than the SWRC ontology. This information is about all we knew about SUMO when performing our analysis. We are thus in exactly the situation of an ontology engineer who wants to gain deeper insights to a previously unknown ontology.

Organization of the paper.
This paper is organized as follows. In the next section, we will provide a brief overview over the history and main lines of research in Social Network Analysis. In Section 3, we will apply a representative selection of SNA techniques to a representative set of ontologies with different structures. In particular, we will analyse the most central parts of the ontology, and will study the eigenvector system assigned to the ontology. Section 4 addresses further applications of SNA for the Semantic Web. In the conclusion, we summarize our experiences, and will discuss the research issues that arise when applying SNA to ontologies and the Semantic Web.

This paper has first been published at ESWC 2006 [15].

2 Social Network Analysis

Already as early as the 1930's Moreno [20] started to describe social relationships within groups using so called *sociograms*. A sociogram is a graph where the members of an observed population are represented as nodes and the relationships among members as edges. The step from modelling relationships between entities of a graph to a structural analysis of these graphs started by using the results from graph theory as early as the 1960's. Pioneers in this field are Harary, Norman and Cartwright [7]. To use the tools of graph theory to analyze and thus describe structures of social networks and to interpret these results in the context of anthropological and sociological contexts was the major achievement of these researchers. The notion of *Social Network Analysis (SNA)* was used to subsume all tools for methodological as well as functional analysis of such group structures.

The two aspects of SNA, the functional aspect and the structural aspect, each highlight a different perspective of research. The functional view focuses on how the function of a network is determined by the structure of a given network. Thus the question of flow between nodes is very prominent. The structural view on the other hand is more interested in the question of structure per se and what statements about a given network can be made based on the analysis of structure alone. Both aspects can be viewed separately, but for some objects of interest, such as organizations, a combined approach may be more appropriate. Since the use of SNA tools in the semantic web environment is just starting out, we will focus in this paper on the structuralist view on SNA, in particular on different notions of centrality. The concept of centrality has many different branches. Just to name a few: in/out degree centrality, betweenness centrality, information centrality, eigenvector centrality. For a good overview see [9].

[3] http://www.ontologyportal.org/

Wasserman and Faust [27, p.205-219] describe to a great extent the history of rank prestige index, which is an eigenvector centrality based concept. This index is based on the idea, that the rank of a group member depends on the rank of the members he or she is connected to. Stated in mathematical terms this yields the eigenvalue equation (for an eigenvalue equal to 1). The components of the principal eigenvector are the rank prestige indices of each group member. This concept is implemented in the hub-and-authority algorithms of Kleinberg [16] and also in the PageRank algorithm proposed by Page and Brin [6].

There have been different approaches to the analysis of unbalanced graphs. All concepts work very well on undirected and unweighted graphs. But if none of these restrictions apply for a given graph, diffuculties arise. Freeman [10] proposed to use the possibility to split any asymmetric square matrix into its symmetric and skew-symmetric part, perform a singular value decomposition of the skew-symmetric matrix, and showed, that the result could be interpreted as a ranking of dominance. Tyler et al. [26] could identify subgroups in unbalanced email networks by analyzing betweenness centrality in the form of inter-community edges with a large betweenness value. These edges are then removed until the graph decomposes into separate communities, thus re-organizing the graph structure.

Barnett and Rice [2] showed that the transformation of asymmetrical data into matrices that avoid negative eigenvalues may result in the loss of information. This is one of the reasons why we will transform the adjacency matrix into a Hermitian matrix in Subsection 3.3.

Beside considering the direction of links as discussed, the notion of a graph can be refined in several ways. One-mode graphs consider just one type of nodes (e. g., participants of an email network), while two-mode graphs distinguish between two types of nodes (but still have only one type of edges), forming thus a bipartite graph (e. g., persons and events they are visiting). More general, n-mode graphs distinguish n types of nodes. The edges may also be typed. Extending the definition of [27], we call a *n-mode multigraph with k edge types* a graph where the nodes may be labeled with n different types and the edges with k different labels. This reflects exactly the structure of an (RDFS-based) ontology. Since the interpretation of such complex graphs is more difficult, one often tries to preprocess the data in order to obtain a 1- or 2-mode graph with only one relation, i. e., with one type of edges. Of course the chosen preprocessing transformation has to be taken into account when interpreting the results.

To analyze networks more easily, several software tools have been developed. These packages include, but are not limited to UCINet,[4] Pajek,[5] and Visone.[6] There are also packages for R[7] and also some implementations in Java.[8] For a good overview on SNA and its history refer to Wasserman and Faust [27] and Freeman [11].

3 Network Analysis of Ontologies

Ontologies can be considered as n-mode multi-graphs with k edge types. As argued above, n-mode multi-graphs with

[4] http://www.analytictech.com/ucinet.htm

[5] http://vlado.fmf.uni-lj.si/pub/networks/pajek/

[6] http://www.visone.info/

[7] http://www.stat.ucl.ac.be/ISdidactique/Rhelp/library/sna/html/00Index.html

[8] http://jung.sourceforge.net/index.html

k edge types are hard to analyze when n is larger than 2 or 3, and k is larger than 1. Therefore we follow the usual approach of projecting them first to a 1- or 2-mode 1-plex network. In the sequel of this section, we will first illustrate the benefit of some basic SNA approaches, before performing a more sophisticated analysis, based on the analysis of the eigenvectors of the adjacency matrix. To show the diversity of results that can be expected from such an analysis, we will apply the basic techniques to two different ontologies: the SWRC and the SUMO ontology, which differ in purpose, scope, and size.

3.1 Preprocessing the Ontologies

As SNA works on graphs, we first transform the ontology into a suitable graph. As in all knowledge discovery (KDD) applications (and probably more so than in the average KDD scenario), the interpretation of the final results is highly sensitive to the decisions made during preprocessing.

A standard approach (which we use also) of turning n-mode networks with k edge types into a (directed or undirected) graph is to collect all types of nodes into just one set of nodes, and to ignore the edge types.[9] We will keep the typing information, though, and refer to it during the analysis.

As a first step, we set up a directed graph for the input ontology in the following way: Technical artifacts were pruned from the ontology. In the KAON ontology API,[10] which we used, these comprise the artificial root concept present in all ontologies, and entities for lexical information such as labels and word stems. Each concept and each property became a node in the graph. Between two concepts C_1 and C_2, a directed edge (C_1, C_2) was added if C_1 is a direct subconcept of C_2. For each property node, edges are added from the each domain concept to the property node, and from the property node to each range concept (unless the property is scalar-valued or untyped), as well as from the property to each superproperty.

The adjacency matrix A of this graph has one row and one column for each node. If there is an edge from the ith to the jth node, then $a_{ij} := 1$, else $a_{ij} := 0$. This matrix is the subject of our subsequent analysis. For the SWRC ontology, A has thus 54+70 rows and 54+70 columns, with entries 0 and 1. The matrix for the SUMO ontology is structured in the same way, with $630 + 236 = 866$ rows and columns in total.

3.2 Basic Methods of Network Analysis

The intuitive approach to analyze a network, represented as a graph $G := (V, E)$ with nodes (or vertices) $v \in V$ and edges $e \in E$, is to start with the number of connections each node has. A node that has many connections is presumed to be important, while a node without connections is presumed to be irrelevant. This concept is called *degree centrality*. In the adjacency matrix A the degree centrality c_k of a vertex in an undirected graph can be calculated as the row or column sum $c_k = \sum_l^n a_{kl}$ of A. If the connection between two nodes has no directional preference this is just called *degree*. If the relationship has an inherent direction, like in 'person A called person B' then the degree is categorized into *in-* (column sums) and *outdegree* (row sums) depending on whether the connection ends at a node or starts at a given node.

[9] A more frequent way for handling different edge types is to perform a sequence of analyses, one for each edge type. For ontologies, however, this approach is not suitable, as most edge types (beside 'is_a' and eventually 'part_of') appear only once.

[10] http://kaon.semanticweb.org/

	# concepts	# relations	diameter	density
SWRC	54	70	16	0.015
SUMO	630	236	27	0.0024

Table 1: Size, diameter, and density of SWRC and SUMO

The *betweenness centrality* is the (normalized) number of shortest paths between any two nodes that pass through the given node. The betweenness centrality provides often a high degree of information, as it describes the location of a node in the graph in a global sense, while in- and outdegree consider the direct neighbor nodes only.

Based on the degree centrality we can define the *density* d of a network. Let the network describe a non-directional relationship between nodes, then the density is defined as the number of existing connections divided by the number $N := \frac{|V|(|V|-1)}{2}$ of all possible edges as $d = \frac{\sum_{kl} a_{kl}}{N}$. Thus a completely connected network has a density of 1. In the directed case one has to keep in mind that at most two connections are possible between two nodes. Thus the density d_d becomes $d_d = \frac{\sum_{kl} a_{kl}}{|V|(|V|-1)}$. This concept is not useful anymore when multiple connections are allowed or when the connections become valued or weighted, because no total number of possible connections can be given in that case.

Another measure of how well a graph is connected is its *diameter*. For all pairs A, B of nodes, we calculate the shortest path from A to B, and take then the maximum over their lengths. The well-known *small-world phenomenon* states that social networks have a small diameter. Diameter and density are used for comparing networks.

Global comparison of SWRC and SUMO.

To analyze the given ontologies, we calculated for each of them the diameter and the density of the network. The results are shown in Table 1. These indices were generated using Pajek.

Compared to typical social networks, the density of the SWRC ontology (0.015) is very sparse. SUMO has an even sparser density with 0.0024. The fact that the difference between both ontologies is approx. one magnitude, which is in the same ratio as their difference in size, indicates that the concepts in both ontologies have a similar number of properties attached in average. It might be interesting to analyze more ontologies to check whether this is some kind of constant stemming from ontology engineering principles. We assume that ontologies are scale-free networks because of their construction.

For studying both ontologies in more details, we computed as next step for all their nodes indegree, outdegree and betweenness centrality.

The SWRC ontology in detail.

Table 2 shows the indegrees, outdegrees and betweenness centralities of the nodes in the graph extracted from the SWRC ontology. While the degrees could still be read from Fig. 1, the betweenness centrality has to be listed.

Considering the degrees only, one observes that the BibTeX part of the ontology was modeled with the highest level of detail: BibTeX-related concepts such as 'Book'

#	Label	d_o	d_i	b_c
1	Academic Staff	10	4	0.102
2	Administrative Staff	1	0	0.
3	Article	7	0	0.
4	Assistant Professor	1	0	0.
5	Associate Professor	1	0	0.
6	Association	1	0	0.
7	Book	13	0	0.
8	Booklet	5	0	0.
9	Conference	2	0	0.
10	Department	2	0	0.
11	Development Project	1	1	0.004
12	Employee	2	4	0.025
13	Enterprise	1	0	0.
14	Event	6	9	0.019
15	Exhibition	1	0	0.
16	Faculty Member	1	3	0.013
17	Full Professor	1	0	0.
18	Graduate	1	1	0.016
19	In Book	13	0	0.
20	In Collection	14	0	0.
21	In Proceedings	12	0	0.
22	Institute	3	0	0.
23	Lecture	2	0	0.
24	Lecturer	1	0	0.
25	Manager	1	0	0.
26	Manual	6	0	0.
27	Master Thesis	1	0	0.
28	Meeting	4	1	0.001
29	Misc	3	0	0.
30	Organization	8	10	0.134
31	Person	7	5	0.024
32	PhDStudent	4	1	0.024
33	Ph DThesis	1	0	0.
34	Proceedings	9	0	0.
35	Product	2	3	0.017
36	Project	7	7	0.12
37	Project Meeting	1	0	0.
38	Project Report	2	0	0.
39	Publication	5	14	0.022
40	Report	2	2	0.005
41	Research Group	3	1	0.01
42	Research Project	1	1	0.004

#	Label	d_o	d_i	b_c
43	Research Topic	3	1	0.079
44	SoftwareComponent	2	0	0.
45	Software Project	2	0	0.
46	Student	2	3	0.027
47	Technical Report	3	1	0.014
48	TechnicalStaff	1	0	0.
49	Thesis	6	2	0.01
50	Topic	1	1	0.
51	Undergraduate	1	0	0.
52	University	3	2	0.041
53	Unpublished	3	0	0.
54	Workshop	2	0	0.
55	Abstract	0	1	0.
56	address	0	9	0.
57	Affiliation	1	1	0.019
58	AtEvent	1	1	0.
59	author	0	10	0.
60	booktitle	0	2	0.
61	carried Out By	1	1	0.009
62	carriesOut	1	1	0.033
63	Chapter	0	2	0.
64	cooperate With	0	2	0.
65	Date	0	2	0.
66	Dealt With In	1	1	0.004
67	Describes Project	1	1	0.004
68	Developed By	1	1	0.017
69	develops	1	1	0.006
70	edition	0	4	0.
71	editor	0	6	0.
72	Email	0	1	0.
73	employs	1	1	0.013
74	Event Title	0	1	0.
75	fax	0	1	0.
76	financedBy	1	1	0.009
77	Finances	1	1	0.033
78	Given By	1	1	0.
79	Has Part Event	1	1	0.
80	Has Parts	0	3	0.
81	hasPrice	0	1	0.
82	head	0	2	0.
83	Head Of	1	1	0.011
84	head Of Group	1	1	0.008

#	Label	d_o	d_i	b_c
85	homepage	0	1	0.
86	howpublished	0	2	0.
87	institution	0	0	0.
88	Is About	1	1	0.078
89	IsWorkedOnBy	1	1	0.069
90	Isbn	0	1	0.
91	Journal	0	1	0.
92	Keywords	0	1	0.
93	Location	0	2	0.
94	member	0	2	0.
95	member Of PC	1	1	0.01
96	month	0	11	0.
97	name	0	6	0.
98	Note	0	1	0.
99	number	0	6	0.
100	organization	0	4	0.
101	organizer Or Chair Of	1	1	0.01
102	Pages	0	4	0.
103	participant	1	1	0.001
104	phone	0	1	0.
105	Photo	0	1	0.
106	Price	0	1	0.
107	product	1	1	0.001
108	projectInfo	1	1	0.009
109	publication	0	2	0.
110	Publisher	0	5	0.
111	publishes	1	1	0.01
112	School	1	1	0.012
113	series	0	8	0.
114	source	0	1	0.
115	has student	1	1	0.002
116	Studies At	1	1	0.024
117	Supervises	1	1	0.023
118	supervisor	1	1	0.006
119	TechnicalReport	1	1	0.017
120	Title	0	2	0.
121	Type	0	3	0.
122	Volume	0	6	0.
123	Works AtProject	0	2	0.
124	Year	0	1	0.
	Mean (Degree)	1.82	1.82	–
	Std (Degree)	2.84	2.55	–

Table 2: Degree and betweenness centrality of concepts (# 1–54) and relations (# 55–124)

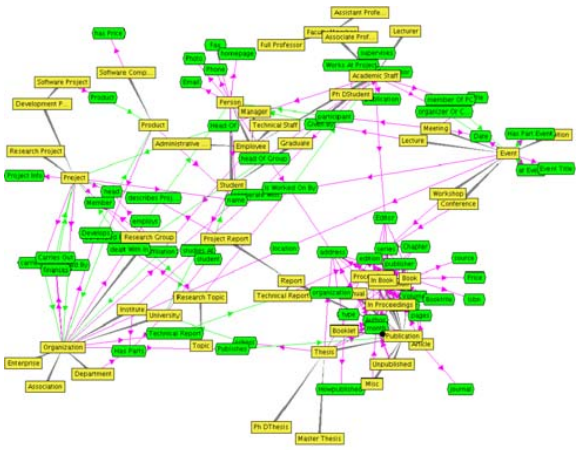


Figure 1: The SWRC Ontology

and 'InCollections' have high outdegrees (i. e. a large number of properties) but no indegree, while the related properties such as 'author', 'month', and 'address' have large indegree.

Properties which apply to all kinds of publications, such as 'title' and 'year', have a low degree, as they are attached to 'Publication' only and are inherited by its subclasses. This is a result of the way we set up the adjacency matrix. An alternative way of setting up the matrix is to model explicitly also the inherited attributes. This is an example for the fact that the modeling step has to be taken into account for the interpretation of the SNA results.

The betweenness centrality gives us a more global description of the roles of nodes in the graph. For SWRC, it returns first of all 'Organization' and 'Project', followed in short distance by Academic Staff' and 'Research Topic'. These are thus the concepts that play a 'bridging role' in SWRC; they are used for describing (chains of) other objects (these are the incoming edges), and they are described by (chains of) other objects (the outgoing edges). From a database perspective, these are typical candidates for joins in a query.

The SUMO ontology in detail.

We also computed the list of in, out and between degrees of the concepts and relations of the SUMO ontology. Due to space restrictions, we omit this list. The means of in- and outdegree (which are obviously equal, as each outgoing edge has to go in somewhere) are at 2.07. The standard deviation is 1.67 for the outdegrees, and 5.8 for the indegrees. The large difference of the standard deviations indicates a heterogeneity in the way of modeling.

When looking at the concepts and relations with out- and indegrees differing largely from the mean, this heterogene-

	Outdegree	Indegree
Process	20	10
Object	15	21
RealNumber	13	15

	Outdegree	Indegree
BinaryObject	3	102
AsymmetricRelation	2	71
UnaryFunction	3	54

Table 3: Highest out- and indegrees of SUMO concepts.

ity can be explained. The highest indegree has the concept 'BinaryPredicate' ($d_i = 102$), and the highest outdegree has the concept 'Process' ($d_o = 20$). The former shows that this technical notions is important for the designers of the ontology. However, this concept is conceptually not part of the domain of interest of the ontology, but rather a meta-construct. If the KR language permitted different arity relations, this would be modeled with language constructs and not by reification. The latter, on the other hand, indicates that 'Process', which is indeed a concept of the domain of interest, is modelled in a high level of detail by providing many properties that a process can have. As in the SWRC ontology, the betweenness centrality emphasizes more on the conceptual part of the ontology: the top node according to this measure is 'Object', followed by 'Formula', 'Entity', 'Physical', 'List', 'Process'. These are the central nodes of the SUMO ontology.

3.3 Eigensystem Analysis

Compared to the centrality measures described so far, the eigensystem of the adjacency matrix provides an overall view of the network, while still allowing a very detailed structure analysis of its parts.

Eigenvector centrality measurements have become a standard procedure in the analysis of group structures. Mostly symmetric (dichotomized) data has been used. Bonacich and Lloyd [5] present an introduction of the use of eigenvector-like measurements of centrality for asymmetric data. The analysis of directed, weighted, asymmetric relationships within a social network poses some difficulties. In this paper we will use a method based on the status (rank prestige) index method [27, p.205-219], that was adapted by the first author to complex adjacency matrices. We sketch the principal approach here (the technical details are presented in [13] and [14]) and adapt it to the analysis of ontologies.

In the following, we consider an ontology as a network which can be modeled as a directed, weighted graph $G = (V, E)$ with V denoting the set of nodes or members and E denoting the set of edges, links or communications between different members. Self references (loops) are excluded.

We use the following construction rules for a complex adjacency matrix H of the initial graph G: First, we construct a square complex adjacency matrix C with n nodes from the possibly weighted real valued adjacency matrix A of graph G by $C = A + iA^t$ with $a_{kl} = m + ip$ where m is the number of outbound edges (or equivalently the weight of the outbound edge) from node k to node l, p is the number of inbound edges (or equivalently the weight of the inbound edge) from node l to node k, and i is representing the imaginary unit ($i^2 = -1$). As can be seen, $c_{kl} = i\overline{c_{lk}}$ holds. Then we rotate C by multiplying it with $e^{-i\frac{\pi}{4}}$ in order to obtain a Hermitian matrix H, i. e., $H := C \cdot e^{-i\frac{\pi}{4}}$. For the proof see [14].

The fact that the resulting matrix is Hermitian has the advantage that it has full rank and thus a complete orthogonal eigenbasis can be found. The consequence is that H can be represented by a Fourier sum as the sum of all orthogonal projectors $P_k = \mathbf{x_k}\mathbf{x_k^*}$, weighted by the corresponding eigenvalue λ_k: $H = \sum_{k=1}^{n} \lambda_k P_k$. Since all eigenvalues are real, they can be sorted by absolute value. In addition the eigenvalue can be used to calculate the covered data variance. These characteristics can be used to analyze a network structure at different levels of relevance as will be shown later in this paper.

Under this similarity transformation the coordinate independent characteristics of the original directional patterns are kept, no information is lost. For instance, more outbound than inbound links lead to a negative sign of the imaginary part of h_{kl}, while more inbound than outbound links lead to a positive sign of the imaginary part of h_{kl}. Now one can analyze the eigensystem of the matrix H in order to gain insights into the structure of the underlying ontology.

Eigensystem analysis of the SWRC ontology.
We start by using the adjacency matrix A for the SWRC ontology from subsection 3.2, and construct the matrix H as described above. This matrix is the subject of further examination.

Let us first have a look at the distribution of the eigenvalues of H as shown in Fig. 2. The diagram suggests a symmetry in the spectrum. This indicates that major components of the network are star like in structure. As the concept hierarchy of SWRC is a tree, this hierarchy has a snowflake structure if considered as graph. Hence our observation that stars are predominant indicates that the concept hiararchy has a more important influence on the overall structure of the SWRC ontology than the non-hierarchical relationships.

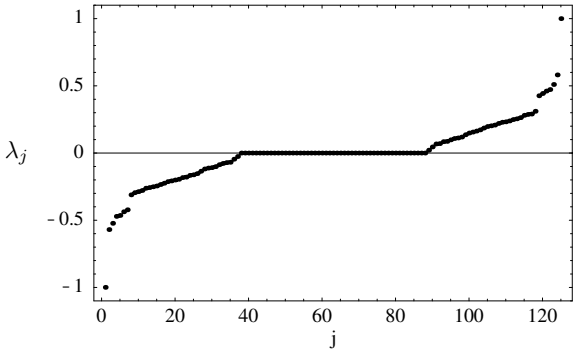


Figure 2: Eigenspectrum of the ontology sorted by value

Fig. 3 displays the cumulative covered variance of the ontology. One can see that the first two eigenvalues cover already 29 % of the variance of the system, that it has a clear distance to the following eigenvalue, and that the first 14 eigenvalues cover approx. 70 % of the overall variance. The remaining eigenvalues contribute marginally only.

In Fig. 4 we now take a more detailed look at the eigenvectors and their components. The lefthand side gives the eigenvalues of each eigenvector, the righthand side gives covered data variance, each eigenvector is represented horizontally with the components numbered 1 through 125 on the bottom, and each eigenvector component is represented as a colored (or gray scaled) field.

The eigenvector components are complex valued, indicating in the phase of the complex number the direction of

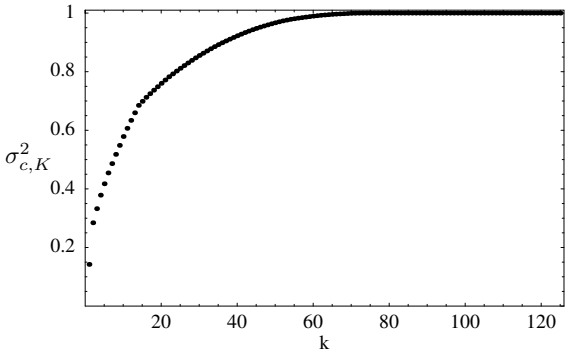


Figure 3: Cumulative covered variance $\sigma^2_{c,K}$ of the SWRC ontology by eigenvalues λ_k

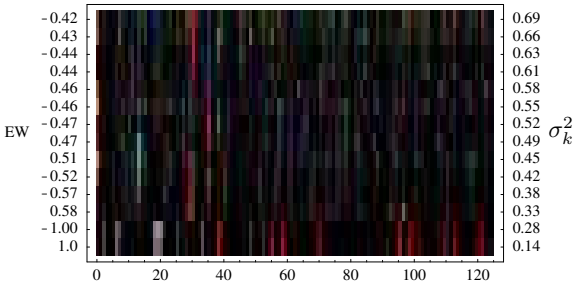


Figure 4: The 14 strongest eigenvectors of the ontology

the connection with respect to the central node,and in the absolute value the relevance of the node in this eigenvector. The color representation lends itself naturally. The absolute value of the component is given by the brightness of the colored field. In gray scales an absolute value of 0 or near 0 is black, while an absolute value close to 1 is bright or has a saturated color. The phase of the complex number is represented by color where a phase of 0 is given as red and counter clockwise $\frac{\pi}{4}$ is yellow, $\frac{\pi}{2}$ is yellow-green, $\frac{3}{4}\pi$ green, $-\pi$ cyan, $-\frac{3}{4}\pi$ blue, $-\frac{\pi}{2}$ blue-magenta, $-\frac{\pi}{4}$ magenta and coming back to red. Thus for example the field with the coordinates $1.0, 39$ is bright red which indicates an eigenvalue with high absolute value and phase 0.

By checking for the largest eigenvector component in each of the eigenvectors (colored red) corresponding to these eigenvalues we can see which concept/relation of the ontology is most central: In the first eigenvector (i. e., the lowest row in Fig. 4, with eigenvalue $+1$), the brightest color is in column 39, which is the concept 'Publication'. The fact that the same column shows in the eigenvector for the negative of the eigenvalue (i. e., in the second row from below, with eigenvalue -1) the same phase (as it is red as well) indicates that the concept 'Publication' is the center of a star like structure. The concept 'Organization' (= column 29) follows (at some distance) with the third and fourth eigenvector. This confirms that publications were in the key focus of the developers of SWRC – a finding we were already pointed to when analysing the in- and outdegrees in the previous subsection. In fact, this fits with the history of the development of the SWRC ontology, which started by transforming the BibTeX format into an ontology.

When looking further down the eigenvalues, we observe that of the three concepts 'Academic Staff', 'Employee' and 'Person', 'Academic Staff' already becomes relevant in the fifth eigenvector, while 'Person' becomes relevant as late as the 11th eigenvector. 'Employee' does not feature as a central concept in any eigenvector. This observation raises the question if the concepts 'Employee' and 'Person' are really needed by the applications the SWRC ontology is targeted to, or if they eventually have just been added because 'one is usually doing so' when designing an ontology.

In Fig. 4, we observe also that the concept 'Academic Staff' *interlaces* with 'Organisation', 'Project' and 'Person'. This behavior is visible by observing that while 'Academic Staff' is colored red in the fifth eigenvector (eigenvalue -0.52), it changes color already in the next line and goes back to red again in line 10 and again in line 14. The three other concepts are colored red in the remaining eigenvectors in between. The absolute values of the eigenvalues do not come in strict pairs of equal absolute value but different sign, thus the three star like structures can not be clearly separated into blocks. The pattern of connection of AcademicStaff to the rest of the network is not easily explained. The pattern of AceademicStaff is distrubed by other structures that have approximately the same amount of connections, thus seperating the eigenvalues.

When considering the eigenvectors of the 36th to 44th eigenvalue (which are out of Fig.4 due to space restrictions), we observe that the concepts 'Assistant Professor', 'Associate Professor', and 'Full Professor' (columns 4, 5, and 17) behave identically with respect to 'Faculty Member' (column 16). As these three concepts are also very similar from an ontology engineering point of view, we take this as a hint that, in a re-engineering step, they should be unified to a single concept, with an additional attribute like 'status'.

Finally, we take a look at the partial sums as described earlier. In Fig. 5 we see the partial sum of the Fourier sum of the first two eigenprojectors weighted by their eigenvalues and rotated back ($\sum_{k=1}^{2} \lambda_k P_k$). This figure was generated by using an adapted k-means cluster algorithm based on the eigensystem. To define the initial cluster centers we use the eigenvector components with the highest absolute value of those eigenvectors that have a negative eigenvalue. We further restrict the selection to all those eigenvectors where the eigenvalues add up to explain data variance to a predefined level of 70%. Thus we do not need to set the number of clusters ex ante. An approximated block matrix is generated when we then sort the eigenvectors and rearrange the eigenvector components accordingly before calculating the eigenprojector. Since the matrices are hermitian, the blocks are symmetric but different in color. The color-coding is the same as in Fig. 4. What is clearly visible is the BibTeX structure as a block in the upper left hand corner. It shows a very strong outbound connection from concepts like 'Book', 'InBook', etc. to Publication', 'address' and 'edition' for example.

If we now take the partial sum of the first 14 eigenprojectors we bring more detail to the picture. In Fig. 6 we see in addition to the BibTeX block five right angles in the matrix plot. These five structures belong to the concepts of 'Organisation', 'Academic Staff', 'Project', 'Event' and 'Person'. As this matrix can be read as a 'partial adjacency

Figure 5: Back rotated partial sum of first two eigenprojectors

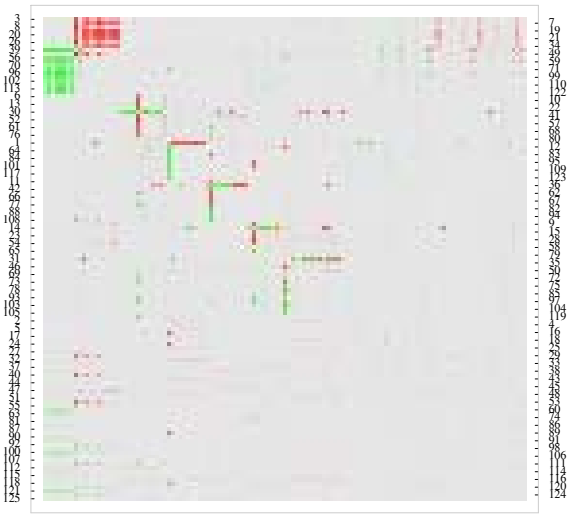


Figure 6: Back rotated partial sum of first 14 eigenprojectors

matrix', such right angles are the structure one expects for stars in the graph: one central node pointing from/to several nodes around it. Different to the BibTeX block that is visible in the upper left hand corner, these concepts play thus a central role in their surroundings. The color of the horizontal part of the angle indicates the direction: for 'Organization', it is green, hence this concept has many inbound edges – its subconcepts. The red color for 'Acadamic Staff' comes from its many outbound properties. 'Project', 'Event' and 'Person' have both incoming and outcoming edges/properties.

Eigensystem analysis of the SUMO ontology.
The eigensystem of the SUMO ontology differs significantly from the one of SWRC. Not only because the SUMO ontology is modeled as a graph with more then 800 nodes, but if differs in that this ontology does not have such a very prominent center.

The spectrum of SUMO (given in Fig. 7) shows – as in the SWRC case – a very strong symmetry, thus suggesting star like structures which come again from the concept hierarchy where several subconcepts all point to their common superconcept. Different to SWRC, the cumulative covered variance (Fig. 8) shows a rather slow incline. While the first two eigenvalues of the SWRC ontology covered already 29% of the data variance, the first two eigenvalues of SUMO cover only about $10\,\%$. The incline then goes without any obvious steps. This suggests that many concepts need to be taken into account to explain the complete ontology. Otherwise said, the degree of detail in SUMO seems to be more balanced than in SWRC.

Due to space restrictions, we cannot display the equivalents of Figs. 4 to 6 for SUMO here. We only present the major insights of our analysis verbally. The concept 'Binary Predicate' contributes most to the interpretation of the first two eigenvectors. 'Asymmetric Relation' seems to follow the same pattern in connecting to other nodes. Thus it is the second strongest concept in the first two eigenvectors. The fact that these two concepts also have a high absolute value in the following six eigenvectors further indicates that these two concepts also contribute to a high extent to the interpretation of these patterns. This might tells us that, in SUMO, these two concepts play a predominant role.

The third and fourth eigenvectors are most strongly influenced by the concepts 'Unary Function', 'Total Valued Relation' and 'Unit of Measure'. These three concepts have similar incoming connections from many concepts which are all of the form'...Fn'. This can be taken as a hint that these bundles of relations could be unified if there were a suitable construct in the KR formalism.

Concluding this section, we summarize that the out-/indegree analysis (and in particular the different differences of the standard deviations for out- and indegree) showed us that SUMO is more heterogenous in its way of modeling (due to the lack of a construct for higher-arity relations in the KR language) than SWRC, but that it is – according to the eigensystem analyis – more homogenous in the distribution of the coverage of different sub-domains of interest.

4 Other Applications of SNA in the Semantic Web Context

There are interesting first results from emerging SNA applications in the Semantic Web context. Mike [19] defines a model of semantic-social networks for extracting lightweight ontologies from folksonomies. Besides calculating such measures as the clustering coefficient, (local) betweenness centrality or the network constraint on the extracted one-mode network, Mika uses co-occurence techniques for clustering the concept network. Stuckenschmidt [24] uses network analysis to partition an ontology into a disjoint and covering set of concepts. After creating a dependency graph of the ontology and computing the strength of the dependencies the line island method [3] is used to determine strongly related concepts. These are then used to form a partition of the ontology graph. The tool Ontocopi described in [1] performs what is called Ontology Network Analysis for initially populating an organizational memory. Several network analysis methods are applied to

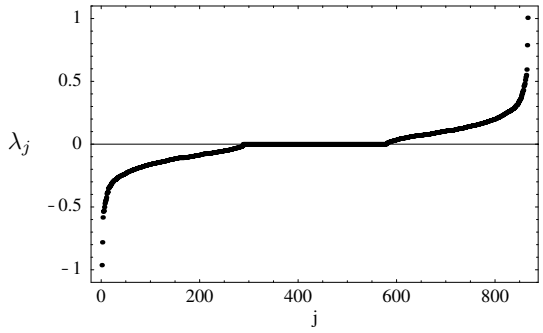


Figure 7: Eigenspectrum of the SUMO ontology sorted by value

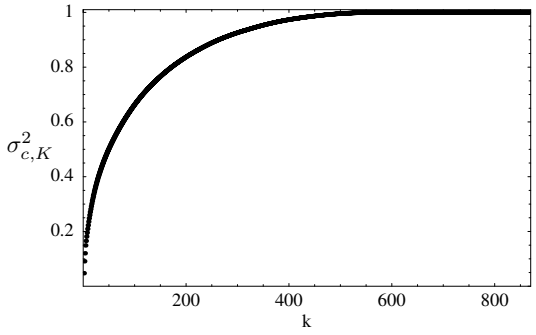


Figure 8: Cumulative covered variance $\sigma^2_{c,K}$ of the SUMO ontology by eigenvalues λ_k

an already populated ontology to extract important objects. In particular, a PageRank-like [6] algorithm is used to find communities of practice of individuals represented in the ontology.

Another field of interest regarding SemNA are Friend Of A Friend (FOAF)[11] networks which are studied for instance in [21] and [8]. Both articles focus on analysing the structure of the social network yielded by a large collection of FOAF documents.

5 Conclusion

In this paper, we have shown that Social Network Analysis provides a promising set of tools for analyzing ontologies and Semantic Web applications, providing deep insights into the structure of ontologies and knowledge bases. In particular, we have seen that the analysis of a given ontology can be done very thoroughly at different levels of granularity.

While the degree based measures from SNA already give an insight into the importance of certain concepts and properties of the ontology, the eigenvector analysis provides a detailed analysis of the importance of entities and the structure of the ontology. Little used "dummy" concepts, as well as candidates for concept fusion can be detected, and the topical clusters within the ontology and their structure can be shown using the eigenprojectors. The analysis is also useful for selecting the right ontology for reuse from a set of candidate ontologies. The eigenvalue analysis provides deep insights into the structure and focus of each ontology and supports the selection of the most suitable result.

[11] http://www.foaf-project.org/

As the two research areas Semantic Web and Semantic Network Analysis met only recently, open issues are still abundant, and provide a rich domain of research for the coming years:

- As seen above, SNA deals well with one- to n-mode networks with one relation. However, ontologies typically consist of more than one or two concepts, and of more than just one kind of relation. A systematic analysis of preprocessing steps which transform an ontology into a one- or two-mode network, as well as the interpretation of the results, is thus needed.
- One step further in this direction is the interesting and far from trivial research question how to expand existing SNA approaches to n-mode multigraph data sets.
- The interpretation of the standard eigenvector analysis needs currently some experience. Future work includes the use of cluster algorithms for rearranging the dimensions of the vector space such that similar dimensions are visualized together.
- (Description) Logics based ontologies describe relations (such as the subsumption hierarchy) implicitly only. It has to be studied whether these relations have to be computed explicitly before SNA techniques can be applied in a meaningful way.
- The next step after analyzing the ontologies is to turn the outcome into support for search, navigation, browsing, and restructuring ontologies and knowledge bases. Seeing the large field of SNA techniques, though, we expect a lot more techniques and tools to come up within the next years.
- Another direction of research is the comparison with philosophical aspects of ontology engineering. The OntoClean [12] method provides a framework for the evaluation of ontological decisions bsaed on philosophical notions e.g. of Identity or Polysemy. Correlations between the structural and philosophical properties of ontologies will have to be researched.

References

[1] Harith Alani, Srinandan Dasmahapatra, Kieron O'Hara, and Nigel Shadbolt. Identifying Communities of Practice through Ontology Network Analysis. *IEEE Intelligent Systems*, 18(2):18–25, March/April 2003.

[2] George A. Barnett and Ronald E. Rice. Longitudinal non-euclidean networks: Applying galileo. *Social Networks*, 7:287–322, 1985.

[3] Vladimir Batagelj. Analysis of large networks - Islands. Presented at Dagstuhl seminar 03361: Algorithmic Aspects of Large and Complex Networks, August/September 2003.

[4] B. Berendt, A. Hotho, and G. Stumme. Towards semantic web mining. In *Proc Int. Semantic Web Conference*, Sardinia, Italy, 2002.

[5] P. Bonacich and P. Lloyd. Eigenvector-like measurement of centrality for asymmetric relations. *Social Networks*, 23:191 – 201, 2001.

[6] Sergey Brin and Lawrence Page. The Anatomy of a Large-Scale Hypertextual Web Search Engine. *Computer Networks and ISDN Systems*, 30(1-7):107–117, April 1998.

[7] Frank Harary ; Robert Z. Norman ; Dorwin Cartwright. *Structural models : an introduction to the theory of directed graphs*. Wiley, New York, 1965.

[8] Li Ding, Lina Zhou, Timothy W. Finin, and Anupam Joshi. How the Semantic Web is Being Used: An Analysis of FOAF Documents. In *HICSS*. IEEE Computer Society, 2005.

[9] M.G. Everett and S.P. Borgatti. The centrality of groups and classes. *Journal of Mathematical Sociology*, 23(3):181–201, 1999.

[10] Linton C. Freeman. Uncovering organizational hierarchies. *Computational & Mathematical Organization Theory*, 3(1):5 – 18, 1997.

[11] Linton C. Freeman. *The Development of Social Network Analysis: A Study in the Sociology of Science*. BookSurge Publishing, 2004.

[12] Nicola Guarino and Christopher A. Welty. Evaluating ontological decisions with OntoClean. *Commun. ACM*, 45(2):61–65, 2002.

[13] Bettina Hoser. *Analysis of Asymmetric Communication Patterns in Computer Mediated Communication Environments*. PhD thesis, Universität Karlsruhe, 2005.

[14] Bettina Hoser and Andreas Geyer-Schulz. Eigenspectralanalysis of Hermitian Adjacency Matrices for the Analysis of Group Substructures. *Journal of Mathematical Sociology*, 29(4):265–294, 2005.

[15] Bettina Hoser, Andreas Hotho, Robert Jäschke, Christoph Schmitz, and Gerd Stumme. Semantic network analysis of ontologies. In *Proceedings of the 3rd European Semantic Web Conference*, Budva, Montenegro, June 2006.

[16] Jon M. Kleinberg. Authoritative sources in a hyperlinked environment. In *Ninth Annual ACM-SIAM Symposium*, pages 668 – 677, Jan 1998.

[17] Jon M. Kleinberg. Authoritative sources in a hyperlinked environment. *JACM*, 46(5):604–632, sep 1999.

[18] Barry Wellman Laura Garton. Social impacts of electronic mail in organizations: A review of research literature. *Communication Yearbook*, 18:434–453, 1995.

[19] Peter Mika. Ontologies Are Us: A Unified Model of Social Networks and Semantics. In Yolanda Gil, Enrico Motta, V. Richard Benjamins, and Mark A. Musen, editors, *ISWC 2005*, volume 3729 of *LNCS*, pages 522–536, Berlin Heidelberg, November 2005. Springer-Verlag.

[20] J.L. Moreno. *Who shall survive? : a new approach to the problem of Human Interrelations*, volume 58 of *Nervous and mental disease monograph series*. Nervous and Mental Disease Publ., Washington, 1934.

[21] John C. Paolillo, Sarah Mercure, and Elijah Wright. The Social Semantics of LiveJournal FOAF: Structure and Change from 2004 to 2005. In Stumme et al. [25].

[22] Christoph Schmitz. Self-organization of a small world by topic. In *Proc. 1st International Workshop on Peer-to-Peer Knowledge Management*, Boston, MA, August 2004.

[23] Michael F. Schwartz and David C. M. Wood. Discovering Shared Interests Using Graph Analysis. *Communications of the ACM*, 36(8):78 – 89, Aug 1993.

[24] Heiner Stuckenschmidt. Network Analysis as a Basis for Ontology Partitioning. In Stumme et al. [25].

[25] Gerd Stumme, Bettina Hoser, Christoph Schmitz, and Harith Alani, editors. *Proc. ISWC 2005 Workshop on Semantic Network Analysis*, Galway, Ireland, November 2005.

[26] J. R. Tyler, D. M. Wilkinson, and B. A. Huberman. Email as spectroscopy: Automated discovery of community structure within organizations. *cond-mat/0303264*, 2003.

[27] Stanley Wasserman and Katherine Faust. *Social Network Analysis: Methods and Applications*, volume 8 of *Structural Analysis in the Social Sciences*. Cambridge University Press, Cambridge, 1 edition, 1999.

On Trading Off Consistency and Coverage in Inductive Rule Learning

Frederik Janssen and **Johannes Fürnkranz**
TU Darmstadt
D-64289, Darmstadt, Deutschland
[janssen,juffi]@ke.informatik.tu-darmstadt.de

Abstract

Evaluation metrics for rule learning typically, in one way or another, trade off consistency and coverage. In this work, we investigate this trade-off for three different families of rule learning heuristics, all of them featuring a parameter that implements this trade-off in different guises. These heuristics are the *m*-estimate, the F-measure, and the Klösgen measures. The main goals of this work are to extend our understanding of these heuristics by visualizing their behavior via isometrics in coverage space, and to determine optimal parameter settings for them. Interestingly, even though the heuristics use quite different ways for implementing this trade-off, their optimal settings realize quite similar evaluation functions. Our empirical results on a large number of datasets demonstrate that, even though we do not use any form of pruning, the quality of the rules learned with these settings outperforms standard rule learning heuristics and approaches the performance of Ripper, a state-of-the-art rule learning system that uses extensive pruning and optimization phases.

1 Introduction

Evaluation metrics for rule learning typically, in one way or another, have to trade off consistency and coverage. On the one hand, rules should only cover a small percentage of negative examples, on the other hand, rules that cover more examples tend to be more reliable, even though they might be less precise on the training examples than alternative rules with lower coverage. An increase in coverage of a rule typically goes hand-in-hand with a decrease in consistency, and vice versa. Thus, many successful rule learning heuristics try to trade off these two aspects. For example, the well-known information gain heuristic of FOIL [Quinlan, 1996] uses a logarithmic difference between a rule and its predecessor as a measure of the increase in consistency of a rule, and multiplies this with the rule's coverage on the positives examples.

In this work, we will show that three well-known evaluation metrics, the *m*-estimate, the F-measure, and the Klösgen measures, may be interpreted as different ways for trading off consistency and coverage. Following the framework laid out in [Fürnkranz and Flach, 2005], we will first visualize their behavior in coverage space in order to demonstrate the way they implement this trade-off. Subsequently, we will report on an extensive experimental study with the goal of determining optimal values for each of these three parameters.

A more detailed description of the results of this paper, and some additional material can be found in [Janssen, 2006].

2 Inductive Rule Learning

The goal of an inductive rule learning algorithm is to automatically learn rules that allow to map the examples of the training set to their respective classes. Different algorithms implement different ways for finding individual rules, but most of them employ a *separate-and-conquer* or *covering* strategy for combining rules into a rule set.

Separate-and-conquer rule learning can be divided into two main steps: In the first one a single rule is learned from the data (the *conquer* step). Then all the (positive) examples which are covered by the learned rule are being removed from the training set (the *separate* step). The next rule is learned on the remaining examples. The two steps are repeated as long as (positive) examples are left in the training set. This ensures that every positive example is covered at least by one rule (*completeness*) and no negative example is included (*consistency*). The origin of this strategy is the AQ-Algorithm [Michalski, 1969] but it is still used in many algorithms [Fürnkranz, 1999].

3 Heuristics and the Coverage Space

In [Fürnkranz and Flach, 2005] it was suggested to visualize the behavior of rule learning heuristics by plotting their isometrics in coverage space, an un-normalized version of ROC-space. In this section, we briefly review the main concepts.

Some notational conventions

In the remainder of this paper the following notations are used:

- p and $n \equiv$ the positive/negative examples covered by the rule (local)
- P and $N \equiv$ the total amount of positive/negative examples in the training set (global)

Rule evaluation heuristics are denoted by the letter h with a subscript to differentiate between them. All heuristics depend only on the number of positive and negative examples that are covered by the rule, and are thus unable to discriminate between rules that cover the same number of positive and negative examples. So it follows that $h(R_i) \equiv h(n_i, p_i)$ holds for all rules R_i. Furthermore it is obvious that $R_1 \neq R_2 \nrightarrow h(R_1) \neq h(R_2)$.

Coverage Space
In distinction to ROC-spaces the coverage space plots the absolute number of positive examples on the *y*-axis and the absolute number of negative ones on the *x*-axis. For example the point $(0, 0)$ represents the empty theory where no example is covered at all. A good algorithm should navigate the learning process in the direction of the point $(0, P)$. It represents the optimal theory because all positive examples are covered and no negative is included. The point $(N, 0)$ represents the opposite theory, and the universal theory, covering all positive and negative examples, is located at (N, P).

Isometrics in Coverage Space
A good method to visualize the peculiarities of a heuristic is to plot their isometrics into a coverage space. A single line of an isometric connects different points (n_i, p_i) with $n_i \in N$ and $p_i \in P$ in this space. Each of these points represents a rule R_i which covers a certain amount of positive (p_i) and negative (n_i) examples. Note that two different rules R_1 and R_2 which covers the same amount of examples, receive the same evaluation value. Isometrics connects those rules $R_1, ..., R_m$ that have the same evaluation value but covering a different amount of examples. Figure 1 shows an example of a coverage space which contains isomterics with their evaluation values.[1]

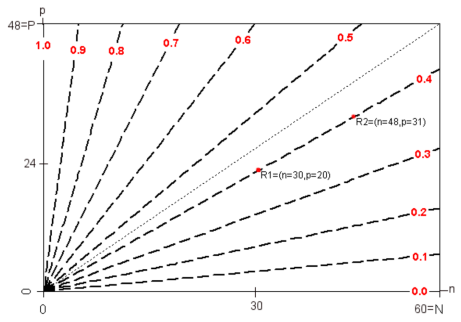


Figure 1: Isometrics in coverage space

Note that there are fewer positive than negative examples in Figure 1 which is necessary for pointing out some special differences between the heuristics (it is important that the positive and negative examples are not equally distributed). In this example the steepest line (the y-axis) represents the greatest evaluation value, which is assigned to all rules that cover some positive but no negative examples.

Based on their isometric structure, we can discern three basic types of heuristics:

- linear isometrics that are not parallel (like the one in Figure 1),
- linear ones that are parallel and
- non-linear ones.

It was shown in [Fürnkranz and Flach, 2005] that linear isometrics may be reduced to two fundamental prototypes: The first one is precision, which tries to optimize the Area under the ROC-Curve for unknown costs and the second one is a cost based optimization for known or expected costs.

[1]For visualization, one is primarily interested in the shape of the isomterics. In this case, the evaluation value is usually ommitted from the graph.

4 Overview of the Used Heuristics

A heuristic is a function that tries to find promising rules by evaluating their coverage of positive and negative examples of the training set. There are two main goals which should be taken into account if an appropriate heuristic is constructed:

- on the one hand the number of positive examples that are covered by the rule should be maximized and
- on the other hand the amount of negative examples that are covered by the rule should be minimized.

A simple way of achieving both objectives is to subtract the number of covered negatives from the covered positives. The resulting heuristic ($h_{Accuracy} = p - n$) is equivalent to accuracy, which computes the percentage of correctly classified examples in all training examples. Other heuristics employ more complex ways to reach these two objectives.

Note that $h_{Accuracy}$ may already be interpreted as a simple way of trading off coverage (represented through the maximization of *p*) and consistency (represented through the minimization of *n*). However, this trade-off is fixed and corresponds to a cost assumption (false positives and false negatives have equal costs) that does not necessarily hold in practice, and, more importantly, may not lead to a good choice of rules.

In the following, we will have a closer look at three heuristics which implement a parametrized form to trade off between coverage and consistency. In the remainder they are called the parametrized heuristics. All three heuristics measure consistency with the same metric, *Precision*, but employ different ways for measuring coverage.

In the following section we will describe these basic heuristics before we will discuss the parametrized heuristics in Section 4.2.

4.1 The basic heuristics

- Precision $h_{Precision} = \frac{p}{p+n}$

A rule is beeing evaluated with the amount of correctly classified examples (*p*) among all covered examples (*p*+*n*). This heuristics picks the steepest line in the PN-space. Its isometrics rotating around the origin as can be seen in Figure 1 which plots those of *Precision*.

- Recall $h_{Recall} = \frac{p}{P}$

This one evaluates a rule with the fraction of covered positive examples in all positive examples of the training set. This estimation is independet of the covered negative examples which results in horizontal parallel lines. The toggling line receives the highest evaluation value because the rules that are located on this one cover the most positive examples.

- Coverage $h_{Coverage} = \frac{p+n}{P+N}$

The idea of this heuristic is similiar to the concept of *Recall*, but the covered negative examples are taken into account as well. The maximum heuristic value is reached if all examples of the training set are covered. In that case the rule corresponds to the point (N, P) of the coverage space and represents the universal theory. The isometrics are lines with a slope of -1.

- WRA $h_{WRA} = \frac{p+n}{P+N} \cdot \left(\frac{p}{p+n} - \frac{P}{P+N} \right) \sim \frac{p}{P} - \frac{n}{N}$

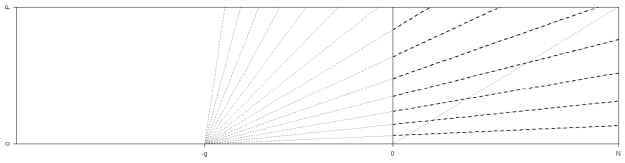


Figure 2: General behavior of the F-Measure

The basic idea of *weighted relative accuracy* (WRA) [Lavrac *et al.*, 1999] is to compute accuracy on a normalized distribution of positive and negative examples. As a result, the lines of the isometrics are now parallel to the diagonal of the coverage space instead of those of $h_{Accuracy}$ which have a slope of 1 (and are independet from the *a priori* class distribution).

WRA differs from the other two coverage-heuristics because it does not directly optimize coverage alone. In fact, like accuracy, it already implements a fixed trade-off between consistency and coverage. However, the experimental evidence of [Todorovski *et al.*, 2000] (which is consistent with our own experience) suggests that this measure has a tendency to over-generalize, i.e., that it places too strong emphasis on coverage.

4.2 The parametrized heuristics

The heuristics that we consider in this work all have a parameter that allows to gradually transform the isometrics of $h_{Precision}$ into one of the three coverage-based metrics that we discussed in the previous section. In the following, we will analzye the changes which happen during this process. If we are able to see how the preferences of the heuristic are modified, we can develop a better understanding of these heuristics and the trade-off they implement.

- F-measure $\qquad h_{F\text{-}Measure} = \frac{\left(\beta^2+1\right)\cdot h_{Precision}\cdot h_{Recall}}{\beta^2\cdot h_{Precision}+h_{Recall}}$

The F-measure [Salton and McGill, 1986] has its origin in Information Retrieval and trades off the basic heuristics $h_{Precision}$ and h_{Recall}. There are some common parametrizations which either focus on the influence of *Precision* or *Recall* or trade them off equally. If $\beta \to 0$ the isometrics correspond to those of $h_{Precision}$ as shown in Figure 2 for $g = 0$. The more the parameter is increased the more the origin of the isometrics is shifted in the direction of the negative N-axis. The observable effect is that the lines in the isometrics becomes flatter and flatter. Finally if $\beta \to \infty$ the resulting isometrics approach those of h_{Recall} which are horizontal parallel lines.

- m-estimate $\qquad h_{m\text{-}estimate} = \frac{p+m\cdot\frac{P}{P+N}}{p+n+m}$

The idea of this parametrized heuristic [Cestnik, 1990] is to presume that a rule covers m training examples *a priori*, which are assumed to be distributed according to the distribution of the examples in the training set $\frac{P}{P+N}$. There is a common parameter setting of $m = 2.0$. In this case – assuming an equal example distribution – we get the *Laplace* heuristic:

$h_{Laplace} = \frac{p+2.0\cdot\frac{1}{2}}{p+n+2.0} = \frac{p+1.0}{p+n+2.0}$

If we inspect the isometrics in relation to the pass through the different parameter settings, we observe a shift of the origin of the coverage space. Related to the situation that was described at the F-measure the origin is moved to the point $(-n_m, -p_m)$ with $p_m = m \cdot \frac{P}{P+N}$ and $n_m = m - p_m$. Here it is shifted in the direction of the negative diagonal of the coverage space as can be seen in Figure 3.

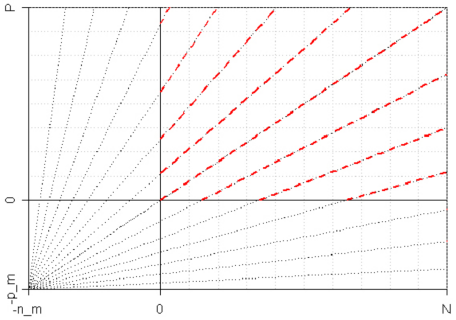


Figure 3: General behavior of the m-estimate

The more the parameter is increased the more the lines become parallel. If $m \to \infty$ the lines are parallel to the diagonal and match the isometrics of h_{WRA}. Thus, the m-estimate performs a trade-off between *Precision* and WRA.

- Klösgen $h_{Klösgen} = \left(h_{Coverage}\right)^{\omega} \cdot \left(h_{Precision} - \frac{P}{P+N}\right)$

This family of measures was first proposed by Klösgen [Klösgen, 1992] and trades off *Precision Gain* (the increase in precision compared to the default distribution $P/(P+N)$) and *Coverage*. Thus *Precision Gain*, as opposed to *Precision*, takes the *a priori* distribution into account.

Klösgen suggested the parameter settings $\omega = 0.5$ and $\omega = 1$, the parameter $\omega = 2$ was investigated by [Wrobel, 1997]. Setting $\omega = 1$ results in WRA, and $\omega = 0$ yields *Precision Gain*, which has the same isometric structure as *Precision* because they only differ by the substraction of a constant. Thus, the Klösgen measure starts with the isometrics of $h_{Precision}$ and first evolves into those of h_{WRA}, just like the m-estimate. However, the transformation takes a different route, as shown in Figure 4. Graph A shows that the lines in the area of low coverage are bent towards the diagonal of the coverage space if the parameter is increased. This indicates a preference for rules which cover few examples. The bending of the lines decreases with an increase of the parameter until they are parallel. The isometrics of picture B comply with those of h_{WRA}.

If the parameter is increased further on, the isometrics converge to $h_{Coverage}$, as shown in Figure 5. Graph A reflects the parameter setting suggested by Wrobel. Here the region of few examples is avoided because the influence of the coverage is increased. Thus the evaluation value is higher the more the lines move away from the diagonal. In picture B this effect is further strenghtend by increasing the parameter to 7.0. Additionally the rules which are evaluated better move towards the point (N, P). If one is looking at a single line in graphic B it starts with a certain amount of positive examples and no negatives. It then shows an almost linear decrease of covered positives and

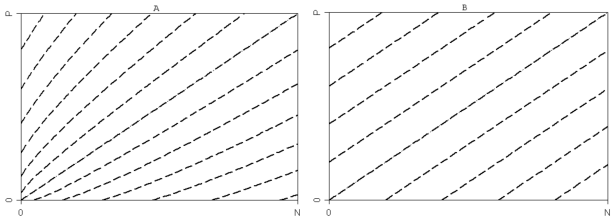


Figure 4: Klösgen-Measure for $\omega \leq 1$

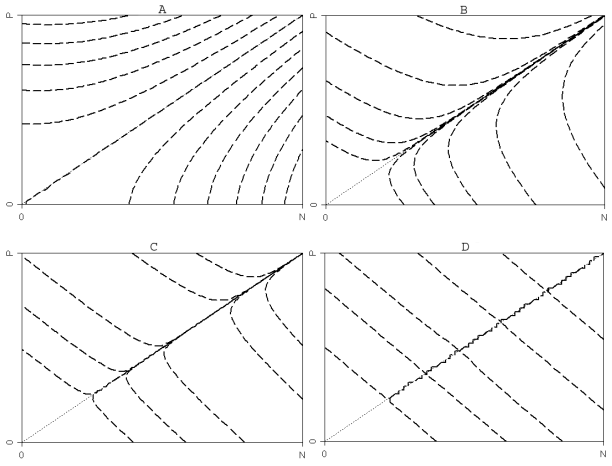


Figure 5: Klösgen-Measure for $\omega > 1$

increase of covered positives, very similar to *Coverage*. However, near the diagonal, the coverage of positive examples suddenly increases as well as those from the negative ones. This behavior is known from WRA. The influence of this heuristic is decreased more and more. This effect is visualized near the diagonal where the lines of the isometric becomes increasingly parallel (graph C in Figure 5). Finally, for $\omega \to \infty$, the influence of WRA is abjured and the isometrics match those of $h_{Coverage}$.

Another interesting variation of the Klösgen measure is to divide $h_{Coverage}$ by $1 - h_{Coverage}$ instead of raising it to the ω-th power. This turns out to be equivalent to the heuristic *Correlation* ($h_{Corr} = \frac{p \cdot (N-n) - n \cdot (P-p)}{\sqrt{P \cdot N \cdot (p+n) \cdot (P-p+N-n)}}$) as has been shown in [Klösgen, 1992].

5 Experimental setup

The primary goal of our experimental work was to determine settings for the parametrized heuristics that are optimal in the sense that they will result in the best overall performance on a wide variety of datasets. Clearly, the optimal setting for individual datasets may vary.

There are some important points that we have to keep in mind when performing the search for the best parameter. First, a large amount of datasets ought to be employed. This is necessary to be independent of special characteristics of them. Second, the parameters should be searched on some datasets and then be tested on an independent set of datasets. This step is important to assure that the obtained parameters are universally valid.

5.1 The datasets

We have used 27 datasets of the UCI-Repository [Newman *et al.*, 1998] for the search of the parameters. It was not important how the datasets were constituted (the number of attributes, classes and examples are indifferent). We chose the following ones because the quantity of examples varies from 24 to 8124, the number of attributes moves between 3 and 69 and finally the lowest amount of classes is 2 and the highest 24:

> *anneal*, *audiology*, *breast-cancer*, *cleveland-heart-disease*, *contact-lenses*, *credit*, *glass2*, *glass*, *hepatitis*, *horse-colic*, *hypothyroid*, *iris*, *krkp*, *labor*, *lymphography*, *monk1*, *monk2*, *monk3*, *mushroom*, *sick-euthyroid*, *soybean*, *tic-tac-toe*, *titanic*, *vote-1*, *vote*, *vowel*, *wine*.

Then the obtained parameters were tested on 30 different datasets that were also taken from the UCI-Repository. Here similiar contraints were valid. In this datasets the number of examples diversifies from 57 to 2310, the count of attributes goes from 4 to 60 and the number of classes is between 2 and 22. The sets were:

> *auto-mpg*, *autos*, *balance-scale*, *balloons*, *breast-w*, *breast-w-d*, *bridges2*, *colic*, *colic.ORIG*, *credit-a*, *credit-g*, *diabetes*, *echocardiogram*, *flag*, *hayes-roth*, *heart-c*, *heart-h*, *heart-statlog*, *house-votes-84*, *ionosphere*, *labor-d*, *lymph*, *machine*, *primary-tumor*, *promoters*, *segment*, *solar-flare*, *sonar*, *vehicle*, *zoo*.

All given accuracies are calculated with a *1x10-stratified Cross Validation* implemented in *weka* [Witten and Frank, 2005].

5.2 The rule learner

We have used a separate-and-conquer rule-learner that is implemented within the SECO-Framework [Fürnkranz, 1999; Thiel, 2005], which is a modular architecture for rule learning that is under development in group. The framework defines a generic separate-and-conquer rule learner that allows to configure specific variations by specifying appropriate modules. In our study, we only varied the heuristics and kept all other options simple and stable. It is not a fundamental point which rule-learner was used because we aim more at an empiric study about different rule learning heuristics than at experiments about various methods of rule learning. The search strategy was chosen to be *Top-Down Hill-Climbing* and no special stopping criterion was used to avoid overfitting because we wanted to solely focus on the heuristics' abilities to evaluate the quality of a rule.

5.3 The evaluation methods

As we have a large number of different individual results, a key issue is how to determine which parameter performed best on average. We have experimented with several choices.

Our primary method was the *Macro-Averaged-Accuracy* of one parametrization of a parametrized heuristic on all of the datasets. Assume that there are m datasets overall. The correctly classified examples of dataset *i* are denoted by $corr_i$ and the total amount of examples of dataset *i* is called $total_i$.

Defintion 5.1 (Macro-Averaged-Accuracy) *The* Macro-Averaged-Accuracy *is computed in two steps: First the accuracy of a heuristic on a single dataset is calculated by dividing the correctly classified by the total number of examples in the corresponding set. Then the accuracy of all datasets is averaged.*

$$Av\_Acc_{macro} = \frac{\sum_{i=1}^{m} \frac{corr_i}{total_i}}{m}$$

However, there are other sensible choices for combining individual results. For examples, *Macro-Averaged-Accuracy* gives the same weight to all datasets. Alternatively, one could assign the same weight to each misclassified example, which results in *Micro-Averaged-Accuracy*. This method assigns a higher weight to datasets with many examples and those with few examples get a minor weight.

Defintion 5.2 (Micro-Averaged-Accuracy) Micro-Averaged Accuracy *is computed by dividing the number of correctly classified examples on* all *different datasets by the total number of examples in all datasets.*

$$Av\_Acc_{micro} = \frac{\sum_{i=1}^{m} corr_i}{\sum_{i=1}^{m} total_i}$$

As there are large differences in the variances of the accuracies of the individual datasets, one could also focus only on the *Ranking* of the heuristics and neglect the magnitude of the accuracies on different datasets. For example, if one heuristic achieves 90.25 % and another one gets 90.23 % on the dataset this difference is not really taken into account when calculating the *Macro-Averaged-Accuracy* on a great number of datasets. In this case the Ranking method provides a better separation because the first heuristic gets rank number 1 and the second rank number 2.[2] Small variations will cancel out over multiple datasets, but if there is a constant small advantage of one heuristic over the other it may be better observed on a combined ranking than on on an averaged accuracy value.

The rankings of the heuristics are combined by adding up their individual ranks.

Defintion 5.3 (Average Rank) *The* Average Rank *is the average of the individual ranks r_i on each dataset.*

$$Av\_Rank = \frac{\sum_{i=1}^{m} r_i}{m}$$

During the search for the optimal setting we selected a large set of interesting parameter settings. All of these parameters are taken as individual heuristics described by their name and the corresponding parameter which leads to a total of 45 parametrized heuristics. For example the general Klösgen measure which are initialized with a parameter of 2.0 is called *Klösgen2.0*. As a result of the evaluation, we have created two tables containing all the heuristics with their *Macro-*, their *Micro-Averaged-Accuracy*, and their *rank*. In addition, we also measured the total *number of rules and conditions*. The first table is produced on the results obtained on the 27 sets on which the parameters have been searched and the second one corresponds to the outcomes on the 30 sets used to evaluate the heuristics (cf. Section 5.1). The correlation of the two tables is an indicator for the universality of the determined parameters. The higher the correlation value, the more reliably will the parameters work on arbitrary datasets. The comparison is made by a correlation value calculated with the *Spearman Rank Correlation*.

Defintion 5.4 (Spearman Rank Correlation) *Given two (averaged and rounded) rankings r_i and r'_i for the heuristics $h_i, i = 1 \ldots m$, the* Spearman Rank Correlation *is defined as*

$$\rho = 1 - \frac{6 \cdot \sum_{i=1}^{m} (r_i - r'_i)^2}{m \cdot (m^2 - 1)}$$

[2]Ties in the ranking are handled by assigning the average rank to all tied accuracies. For examples, if the heuristics on the ranks 5–8 all have equal accuracies, they all receive the rank 6.5 on this dataset.

It computes a correlation value between -1 (which stands for a perfect negative correlation) and 1 (which represents the perfect positive correlation). A result of 0 means no correlation at all. There are some advantages of using this method:

- it is robust against anomalies and
- it is praticable for variables whose relation is non-linear.

6 Searching for the optimal parameter

This section describes our method for searching for the optimal parameter setting. First, we tested a wide range of intuitively appealing parameter settings to get an idea of the general behavior and the differences of the three parametrized heuristics. The promising parameters were restricted further on. Our expectation was to have a general behavior like the one shown in Figure 6.

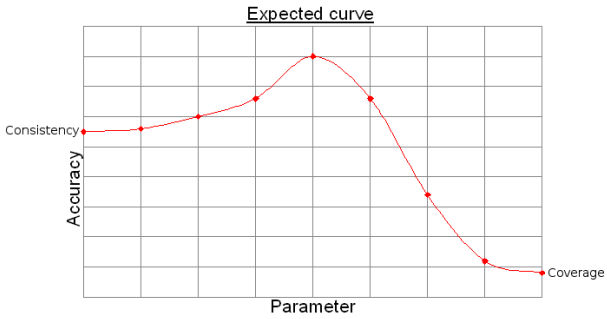


Figure 6: Expected curve

The start point, where the parameter is close to 0, represents the accuracy reached by *Precision*. Then the correctness should raise with an increase of the parameter until the optimal one is found (where the *Macro-Averaged-Accuracy* is the highest). If the parameter is increased further on, the accuracy should decrease again and should converge towards the value achieved by the used coverage heuristic. In Figure 6 the accuracy of *Precision* (which represents *Consistency*) is higher than those of the related coverage heuristic. As we will see, this holds for the Klösgen measures and the F-measure. At the m-estimate the situation is reversed because the coverage heuristic WRA achieves a higher accuracy than *Precision*.

6.1 The search strategy

As mentioned above, we first used a fixed set of parameters in order to identify the basic regions where the optimal parameter can be. After this first test, the general search algorithm is started.

There are some constraints which underlay the search method. First it should be clear that the only way to find the optimal parameter is to perform an exhaustive search through the space of all possible parameter settings. Due to limited computational power it is of course not possible to use this kind of search in practice. Another way of searching the above-mentioned space is to simply test different parameters in a certain interval and analyze promising ones further more. A good method to do this is to use nested intervals. We start with the best parameter found so far, and

Algorithm 1 The search algorithm

```
PROCEDURE SearchBestParameter (a, b, i, currHeur, dataSets)
{
   p_best = 0
   acc_best = 0
   acc_former = acc_best
   t = 0.001
   # create a list (params) containing the parameters to
   # search
   params = createList(a, b, i)
   # find the best parameter in this list
   p_best = getBestParam(currHeur, params, dataSets)
   # get the highest accuracy
   acc_best = getHighestAcc(currHeur, params, dataSets)
   # if no significant improvement is yielded return the
   # best parameter and break
   IF (acc_best − acc_former < t)
   {
     RETURN (p_best)
     BREAK
   }
   # call the procedure recursively with the new borders
   # and the new increment
   SearchBestParameter(p_best − i/2, p_best + i/2, i/10, currHeur,
   dataSets)
}
```

divide the previous increment by a predefined value. Next, a certain interval around this best parameter is searched for a better value. If one is found, it is refined using the same method as above. So basically a certain value is selected out of an interval and a new interval is created around the value. Then the next interval is selected out of the previous one. If the length of the intervall is converging towards 0, a real number is yielded which lies in every intervall. An example search is shown in Table 1.

There are several constraints of setting the parameters of the search. For example, the farther the lower border *a* and the upper border *b* of the related interval are away from the best parameter p_{best}, the higher the probability is that the global optimal parameter will be found. Due to the restricted calculation power the constraints are defined as follows (*i* stands for the increment):

$$a = p_{best} - \frac{i}{2},\ b = p_{best} + \frac{i}{2} \text{ and } i = \frac{i}{10}$$

Additionally a threshold *t* for minimum accuracy differences has to be initialized. Suitable values could be derived from significance tests, but we simply set this value to 0.001. A schematic description of the search algorithm is given in pseudo code at Algorithm 1.

There are some problems resulting out of the proposed method:

- there is a possibility to simply miss the best parameter due to the fact that the global best parameter may lie under or above the borders (if the best one so far is 1 for example, the interval that would be searched is $[0.5, 1.5]$; if the global optimum is 0.4, it would not be detected)
- there is only one possible optimum that is closer examined (the global optimum could hide between two apparently lower values)

The latter can be addressed by keeping a list of candidate parameters that all be refined and from which the best one is selected. One has to define how many candidates should be maintained. Therefore it is necessary to introduce a threshold that discriminates between a normal and a candidate parameter. It is not trivial to determine such a threshold. Due to this the number of candidate parameters is limited to 3 (all experiments confirmed that this is sufficient). The first problem remains unresolved. Because of complexity issues the borders have to be adjusted as proposed. The focus of this work is not to find the global optimum definitely.[3] Instead, we aim at identifying interesting intervals of the 3 parametrized heuristics. If we can find the region that is likely to contain the best parameter, independent from the datasets, this would already be a sufficiently interesting result.

Table 1: A sample parameter search

Run	set which has to be searched	increment	best parameter	Accuracy
1	$\{0.1, \ldots, 1.0\}$	0.1	0.4	84.5658
2	$\{0.35, \ldots, 0.45\}$	0.01	0.42	84.6852
3	$\{0.415, \ldots, 0.425\}$	0.001	0.418	84.7015
4	$\{0.4175, \ldots, 0.4185\}$	0.0001	0.4176	84.7045
5	$\{0.41755, \ldots, 0.41765\}$	0.00001	0.4176	84.7045

6.2 Optimal parameters for the three heuristics

In this section we focus on the results of the search and describe the different parameters we have found for the three heuristics. In addition, we introduce graphs in which we plot curves that show interpoloated accuracy values over various parameter settings. These curves illustrate the behavior of the different parameter settings.

Klösgen measures

Figure 7 (a) shows the results for the Klösgen measures. The curve corresponds to our expectations (cf. Figure 6). In the region from 0.1 to 0.4 the accuracy increases continuously until it reaches a global optimum at 0.4323, which achieves an accuracy of $84.9909\,\%$. After the second run of the search algorithm no better candidate parameters were found. The accuracy decreases again with parametrizations greater than 0.6. The parameter setting of 1.0 represents WRA. Larger values are not shown, as they turned out to further decrease the accuracy. As illustrated in Figure 4, the shown interval $[0, 1]$ describes the trade-off between *Precision* and WRA. So one can say that the trade-off between WRA and *Coverage*, which is obained for values of $\omega > 1$, does not reach a sufficient accuracy and can therefore be ignored.

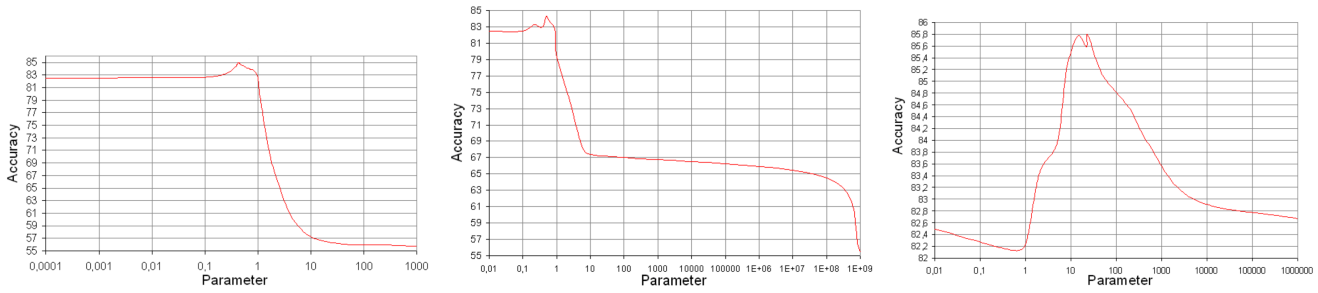
F-measure

For this heuristic the same interval as with the Klösgen measures is of special interest (Figure 7 (b)). Already after the first run the parameter 0.5 got the highest accuracy of $82.2904\,\%$. A better one could not be found during the following runs of the algorithm. After the second pass two other candidate parameters, namely 0.493 with $84.1025\,\%$ and 0.509 with $84.2606\,\%$ were found. But both of them could not be refined to achieve a higher accuracy and were therefore ignored. The main difference between the Klösgen measures and the F-measure was, that for the latter, the accurary has a steep descent at a very high parametrization of $1 \cdot E^9$. At this point it reaches the same value as the Klösgen measures (about $55\,\%$).

m-estimate

The behavior of this heuristic differs from the other two parametrized heuristics in several ways. For example, we even noticed a decrease for low parameter settings (Figure 7 (c)). The main problem is that the first run exhibited no clear tendencies. So the region in which the best parameter should be could not be restricted, and we had to search a larger interval. In Figure 8 we zoom into the range

[3] The optimal parameter will change anyhow if it is searched on different datasets.



(a) Klösgen-measures (b) F-measure (c) m-estimate

Figure 7: Accuracy over parameter values for the three parametrized heuristics

$[1, 100]$ to give a better impression of the heuristic's behavior in that critical region. The figure also shows many variations, which complicated the identification of an optimal parameter range. A significant deterioration in the accuracy cannot be detected before a value of 90 where it falls permately below $82.1\,\%$.

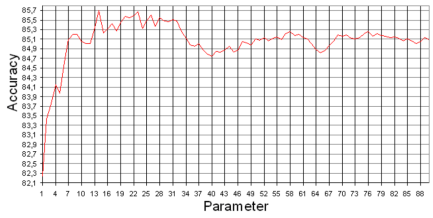


Figure 8: The curve of the m-estimate for a big interval

Due to this, the interval $[0, 35]$ had to be searched with an increment of 1 because all parameters greater than 35 got accuracies under $85.3\,\%$ and we had to restrict the area of interest. After this first run there were 3 candidate parameters, from which 14 achieves the greatest accuracy. After a second run, 23.5 became optimal, which illustrates that it was necessary to maintain a list of candidate parameters. After a sufficient amount of runs we found the opimal parameter at 22.466. The achieved accuracy of 85.8003 was the best value of all heuristics.

6.3 Behavior of the optimal heuristics

In this section, we compare the parameters which have been found for the three heuristics. The best heuristic was the m-estimate. The next one was the generalized Klösgen measures which was approximately $1\,\%$ worse, followed by the F-measure whose optimal value lagged about $0.7\,\%$ behind the generalized Klösgen measures. It is interesting to look at the isometrics of the best parameter settings of the heuristics. Interestingly, the optimal values of the m-estimate and the Klösgen measures implement a very similar heuristic, as can be seen in subfigures (b) and (c) of Figure 9. Minor differences are detectible in the low coverage region near the origin, where the isometrics of the Klösgen measures are slightly bended.

The F-measure produces somewhat different isometrics, which mostly results from its bias towards parallel lines near the N-axis, because the origin of the isometrics can only move along this axis. Therefore it can never reach an isometric structure similiar to this of the other two measures.

6.4 Validity of the results

In order to make sure that we do not overfit the datasets that were used for this study, we compared the rankings of 15 different parametrizations per heuristic on the original datasets with their rankings on new datasets, which were not used for finding the optimal values. We also added some standard heuristics (*Correlation*, *WRA*, *Precision*, *Laplace* and *Accuracy*), as well as JRIP, WEKA's implementation of RIPPER [Cohen, 1995], which, in contrast to our algorithms, uses sophisticated pruning mechanisms. In total, 52 heuristics were compared.

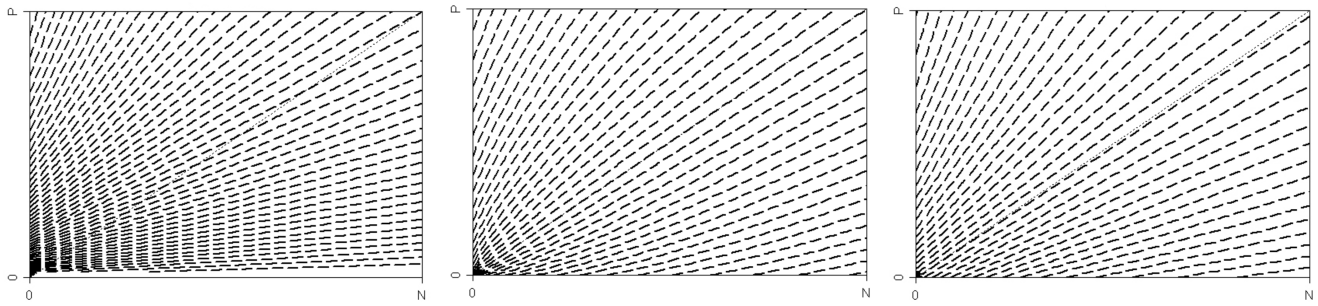
The results for the original datasets are summarized in Table 2 and for the test sets in Table 3. The numbers in braces describes the rank of each heuristic according to the measure of the respective column. The correlation value that describes the similarity between the two tables was 0.92 for Macro-, 0.91 for Micro-Averaged-Accuracy, 0.99 for the number of conditions and 0.99 for the number of rules which is not displayed in the tables. This is a very

Table 2: Different evaluations

	average accuracy		average	
Heuristic	Macro	Micro	Rank	# conditions
m-estimate22.466	85.80 (1)	93.87 (2)	16.13 (2)	36.81
Klösgen0.4323	84.99 (7)	93.62 (7)	18.69 (7)	46.89
JRip	84.45 (11)	93.80 (4)	17.37 (5)	16.93
F-measure0.5	84.29 (12)	92.94 (14)	19.07 (8)	40.78
JRip-P	83.88 (17)	93.55 (9)	21.93 (13)	45.52
Correlation	83.66 (19)	92.39 (24)	25.15 (25)	37.11
WRA	82.71 (29)	90.43 (37)	28.26 (35)	14.41
Precision	82.50 (33)	92.21 (28)	27.89 (31)	99.93
Laplace	82.28 (34)	92.26 (27)	27.30 (30)	91.04
Accuracy	82.28 (35)	91.31 (33)	28.19 (34)	84.07

Table 3: Different evaluations on the "Test Set"

	average accuracy		average	
Heuristic	Macro	Micro	Rank	# conditions
JRip	78.98 (1)	82.42 (1)	16.60 (1)	12.20
m-estimate22.466	78.68 (2)	81.72 (3)	17.97 (3)	47.27
JRip-P	78.50 (5)	82.04 (2)	18.47 (5)	49.80
Klösgen0.4323	78.49 (6)	81.33 (14)	19.87 (12)	62.67
F-measure0.5	78.14 (12)	81.52 (9)	18.27 (4)	52.43
Correlation	77.57 (22)	80.91 (21)	24.70 (26)	47.50
Laplace	76.89 (28)	79.76 (30)	26.27 (31)	118.83
Precision	76.22 (33)	79.53 (35)	29.80 (40)	129.17
WRA	75.80 (37)	79.35 (37)	27.03 (34)	12.13
Accuracy	75.60 (41)	78.47 (39)	31.23 (42)	104.77



(a) F-measure (b) Klösgen (c) m-Estimate

Figure 9: Isometrics of the best parameter settings

high correlation, which makes us confident that the found parameters will also work well on new datasets.

7 Conclusions

In this work, we investigated three different ways for trading of consistency and coverage for rule learners, in the form of three different parametrized heuristics. For each heuristic, we determined an optimal parameter, which proved to be quite stable over multiple domains. The best trade-off was achieved for the m-estimate, but the other heuristics produced quite similar behavior, which we confirmed by visualizing their isometrics in coverage space. While the exact value for this trade-off is certainly not that important, our experiments provide evidence that the optimal parameters are located in the interval $[0.3, 0.5]$ for both the Klösgen measures and the F-measure, and $[13, 27]$ for the m-estimate.

As further work we could examine other evaluation methods to find the optimal parameters. Another promising way is to re-adjust the trade-off every time a rule is learned and the examples are removed from the training set. This approach is located in the domain of Meta-Learning. Finally, we intend to look at different trade-offs between consistency and coverage, most notably to a parametrized cost metric. For these, the isometrics are always parallel lines, so that the behavior of the optimal value will necessarily be different from those studied here.

Acknowledgements

Part of this research was supported by the *German Science Foundation (DFG)* under grant no. FU 580/2-1.

References

[Cestnik, 1990] Bojan Cestnik. Estimating probabilities: A crucial task in Machine Learning. In L. Aiello, editor, *Proceedings of the 9th European Conference on Artificial Intelligence (ECAI-90)*, pages 147–150, Stockholm, Sweden, 1990. Pitman.

[Cohen, 1995] William W. Cohen. Fast Effective Rule Induction. In Armand Prieditis and Stuart Russell, editors, *Proceedings of the 12th International Conference on Machine Learning*, pages 115–123, Tahoe City, CA, July 9–12, 1995. Morgan Kaufmann.

[Fürnkranz and Flach, 2005] Johannes Fürnkranz and Peter A. Flach. ROC 'n' Rule Learning - Towards a Better Understanding of Covering Algorithms. *Machine Learning*, 58(1):39–77, January 2005.

[Fürnkranz, 1999] Johannes Fürnkranz. Separate-and-Conquer Rule Learning. *Artificial Intelligence Review*, 13(1):3–54, February 1999.

[Janssen, 2006] Frederik Janssen. Eine Untersuchung des Trade-Offs von Precision und Coverage bei Regel-Lern-Heuristiken, July 2006.

[Klösgen, 1992] Willi Klösgen. Problems for Knowledge Discovery in Databases and their Treatment in the Statistics Interpreter Explora. *International Journal of Intelligent Systems*, 7:649–673, 1992.

[Lavrac *et al.*, 1999] Nada Lavrac, Peter A. Flach, and Blaz Zupan. Rule evaluation measures: A unifying view. In *ILP*, pages 174–185, 1999.

[Michalski, 1969] Ryszard S. Michalski. On the Quasi-Minimal Solution of the Covering Problem. In *Proceedings of the 5th International Symposium on Information Processing (FCIP-69)*, volume A3 (Switching Circuits), pages 125–128, Bled, Yugoslavia, 1969.

[Newman *et al.*, 1998] D.J. Newman, C.L. Blake, S. Hettich, and C.J. Merz. UCI Repository of Machine Learning databases, 1998.

[Quinlan, 1996] J.R. Quinlan. Learning First-Order Definitions of Functions. *Journal of Artificial Intelligence Research*, 5:139–161, 1996.

[Salton and McGill, 1986] Gerard Salton and Michael J. McGill. *Introduction to Modern Information Retrieval*. McGraw-Hill, Inc., New York, NY, USA, 1986.

[Thiel, 2005] Matthias Thiel. Separate and Conquer Framework und disjunktive Regeln, 2005.

[Todorovski *et al.*, 2000] Ljupco Todorovski, Peter Flach, and Nada Lavrac. Predictive performance of weighted relative accuracy. In Djamel A. Zighed, Jan Komorowski, and Jan Zytkow, editors, *4th European Conference on Principles of Data Mining and Knowledge Discovery (PKDD2000)*, pages 255–264. Springer-Verlag, September 2000.

[Witten and Frank, 2005] Ian H. Witten and Eibe Frank. *Data Mining — Practical Machine Learning Tools and Techniques with Java Implementations*. Morgan Kaufmann Publishers, 2nd edition, 2005.

[Wrobel, 1997] Stefan Wrobel. An Algorithm for Multi-relational discovery of Subgroups. In Jan Komorowski and Jan Zytkow, editors, *Proc. First European Symposion on Principles of Data Mining and Knowledge Discovery (PKDD-97)*, pages 78–87, Berlin, 1997. Springer Verlag.

Constraining the Search Space in Temporal Pattern Mining

Andreas D. Lattner and Otthein Herzog
TZI – Center for Computing Technologies, Universität Bremen
PO Box 330 440, D-28334 Bremen, Germany
{adl|herzog}@tzi.de

Abstract

Agents in dynamic environments have to deal with complex situations including various temporal interrelations of actions and events. Discovering frequent patterns in such scenes can be useful in order to create prediction rules which can be used to predict future activities or situations. We present the algorithm *MiTemP* which learns frequent patterns based on a time interval-based relational representation. Additionally the problem has also been transfered to a pure relational association rule mining task which can be handled by *WARMR*. The two approaches are compared in a number of experiments. The experiments show the advantage of avoiding the creation of impossible or redundant patterns with *MiTemP*. While less patterns have to be explored on average with *MiTemP* more frequent patterns are found at an earlier refinement level.

1 Introduction

Agents in dynamic environments have to deal with complex situations including various temporal interrelations of actions and events. If more elaborated technologies like planning should be used, the representation of the agent's belief including background knowledge for its behavior decision can become very complex, too. It is necessary to represent knowledge about object classes and their properties, actual scenes with objects, their attributes and relations. If even more complex scenes with temporal extents shall be described this additional dimension must also be incorporated in the formalism.

Discovering frequent patterns in dynamic scenes can be useful in order to create prediction rules which can be used to predict future activities of other agents or to predict future situations and, thus, adapt the own behavior by taking into account this additional knowledge. In previous work a relational representation with temporal validity intervals and algorithms for mining temporal patterns have been introduced in [Lattner *et al.*, 2006]. Here, we present a new algorithm which is also based on such a representation but avoids the creation of redundant patterns by defining an optimal refinement operator similar to the one in [Lee, 2006]. Additionally to the own implementation the problem has also been transfered to a pure relational association rule mining task which can be handled by *WARMR* [Dehaspe and Toivonen, 1999].

2 Related Work

Association rule mining addresses the problem of discovering association rules in data. One typical example is the mining of rules in basket data [Agrawal *et al.*, 1993]. Different algorithms have been developed for the mining of association rules in item sets (e.g., Apriori [Agrawal and Srikant, 1994]). [Mannila *et al.*, 1997] extended association rule mining by taking event sequences into account. They describe algorithms which find all relevant episodes which occur frequently in the sequence. Höppner presents an approach for learning rules about temporal relationships between labeled time intervals [Höppner, 2003]. The time intervals consist of propositional events and temporal relations are described by Allen's interval logic [Allen, 1983].

Dehaspe and Toivonen combine association rule mining algorithms with ILP techniques. Their system *WARMR* is an extension of Apriori for mining association rules over multiple relations [Dehaspe and Toivonen, 1999]. The generated rules consist of sets of logical atoms. This more expressive representations (compared to itemset mining) allows for discovering relational association rules.

[Kaminka *et al.*, 2003] introduce an approach which creates a sequence of certain events or behaviors from objects' positions in RoboCup soccer matches and searches for frequent sequences in the data. In their work they compare two approaches based on frequency counts and statistical dependencies. The events in the sequences do not have a temporal extent and the learned patterns do not abstract from concrete objects in the events.

[Lee, 2006] presents an approach to mine first-order logical (*SeqLog*) patterns from sequential relational data. He defines optimal refinement operators and algorithms for finding all frequent patterns. The support is defined by the number of event sequences that match the pattern.

The approaches of Höppner, Kaminka et al., and Lee are quite similar to the one presented here. In contrast to Höppner and Kaminka our approach can learn relational patterns with variables which is also supported by [Lee, 2006]. Like Höppner we take an interval-based representation as input and mine temporal patterns with temporal inter-relations between these intervals. We also use a similar support definition which is based on the probability to find a pattern at a random sliding window position in the sequence. In contrast to our work, Kaminka et al. and Lee's approaches are based on event sequences without temporal extent.

Our approach combines and extends these existing approaches. To the best of our knowledge no approach has addressed the mining of frequent temporal patterns from multi-relational time interval-based data. Our approach allows for taking hierarchical class information into account

(while existing approaches just provide types for variables). Reasoning techniques are used to exploit the knowledge about temporal relations and about classes in order to reduce the number of patterns to be generated and to avoid checking inconsistent patterns.

3 Definitions and Problem Statement

The goal of the mining task is to find the set of all frequent temporal patterns from a dynamic scene. Before the approach is described in detail we provide some definitions. Let $\mathcal{V}$, $\mathcal{O}$, $\mathcal{C}$, and $\mathcal{IR}$ be the sets of variables, objects, classes, and temporal interval relations, respectively.

Definition 3.1 (Dynamic Scene) *A dynamic scene is described by the 4-tuple $ds = (\mathcal{P}, \mathcal{O}, i, \mathcal{DSS})$ where $\mathcal{P}$ is the set of predicate instances, $\mathcal{O}$ is the set of objects in the dynamic scene, $i : \mathcal{O} \rightarrow \mathcal{C}$ maps the objects to classes (instance-of relation), and $\mathcal{DSS}$ is the dynamic scene schema.* □

Definition 3.2 (Dynamic Scene Schema) *The schema of a dynamic scene $\mathcal{DSS} = (\mathcal{C}, sc, \mathcal{PD}, \mathcal{IR})$ consists of all schematic information. $\mathcal{C}$ is the set of classes and $sc : \mathcal{C} \rightarrow \mathcal{C}$ maps classes to their super classes and thus describes the class hierarchy. $\mathcal{C}$ consists of at least one element which denotes the most general class ($object$). $\mathcal{PD}$ is the set of predicate definitions and $\mathcal{IR}$ the set of the temporal interval relations.* □

Predicate definitions consist of the identifier, the arity, and the allowed ranges for the objects in their instances.

Definition 3.3 (Predicate Definition) *A predicate definition pd is defined as $pd = (pd_{name}, pd_{arity}, pd_{classes})$ with $pd_{classes} = (c_1, c_2, \ldots, c_{pd_{arity}})$. All c_i denote classes in the dynamic scene schema, i.e., $c_i \in \mathcal{C}$ with $1 \leq i \leq pd_{arity}$.* □

Definition 3.4 (Predicate Instance) *Predicate instances $pi = (pd, p_{objects}, \langle s, e\rangle)$ are instances of predicate definition pd, consist of a list of object identifiers $p_{objects} = (o_1, o_2, \ldots, o_{pd_{arity}})$ with $\forall o_i : o_i \in \mathcal{O}$ of the dynamic scene, and additionally contain an interval of validity $\langle s, e\rangle$ with start time s and end time e.* □

For a better understanding we denote predicate instances in a more readable way: $holds(predicate(o_1, o_2, \ldots, o_{pd_{arity}}), \langle s, e\rangle)$ represents a predicate with $pd_{name} = predicate$, $p_{objects} = (o_1, o_2, \ldots, o_{pd_{arity}})$, start time s, and end time e. An example for a predicate in this notation is: $holds(inBallControl(p7), \langle 17, 42\rangle)$.

Definition 3.5 (Interval Relation Function) *The interval relation function $ir : \langle \mathbb{N}, \mathbb{N}\rangle \times \langle \mathbb{N}, \mathbb{N}\rangle \mapsto \mathcal{IR}$ maps time interval pairs to interval relations.* □

It depends on the used interval relations $\mathcal{IR}$ how the actual mapping from the interval pairs to the interval relation has to be performed. Using, for instance, Allen's interval relations $ir(\langle s_1, e_1\rangle, \langle s_2, e_2\rangle) = b$ (*before*) if (and only if) $e_1 < s_2$ [Allen, 1983].

An atomic pattern consists only of one predicate pattern. The difference to predicate instances is that the list of arguments do not need to denote objects. In the general case the elements of the pattern are variables that can be bound to objects while pattern matching. However, it is also allowed to have arguments bound to objects in the pattern already.

Definition 3.6 (Atomic Pattern) *An atomic pattern is defined as $p = (pd, p_{arg})$ where pd denotes a predicate definition and p_{arg} specifies a list of terms $p_{arg} = (v_1, v_2, \ldots, v_{pd_{arity}})$. All v_i are either elements of $\mathcal{O}$ as defined in the dynamic scene or are elements of $\mathcal{V}$, the set of variables, i.e., it holds $\forall v_i \in \mathcal{V} \cup \mathcal{O}$.* □

Definition 3.7 (Conjunctive Pattern) *A conjunction of atomic patterns is called conjunctive pattern. It connects the atomic patterns by a conjunction (logical AND): $p_1 \wedge p_2 \wedge \ldots \wedge p_n$ where the p_i are atomic patterns with $1 \leq i \leq n$; n is called the size of the pattern.* □

Similarly to the predicate instances above we introduce a short notation for conjunctive patterns: $predicate\_1(v_{1_1}, \ldots, v_{1_{pd_{arity}}}) \wedge \ldots \wedge predicate\_n(v_{n_1}, \ldots, v_{n_{pd_{arity}}})$. An example of a conjunctive pattern with two predicates is $uncovered(X) \wedge pass(Y, X)$.

Definition 3.8 (Class Restriction) *The class restriction defines for each variable v_i of a conjunctive pattern its least general class c_i. For a given variable list $(v_1, v_2, \ldots, v_n)$ the class restriction is represented by a class list $(c_1, c_2, \ldots, c_n)$.* □

Variable unifications define if certain variables in a (conjunctive) pattern should refer to the same object in the assignment during pattern matching, i.e., if variables are unified.

Definition 3.9 (Variable Unification) *A variable unification of a pattern p is defined as the unification of two different arguments v_1 and v_2 of one or two predicates of p, i.e., it must hold that $v_1 = v_2$.* □

Binding a variable to a constant (i.e., to an instance) is denoted as instantiation:

Definition 3.10 (Instantiation) *A variable v_i is instantiated if it is bound to an instance of the set of objects in the dynamic scene, i.e., if $v_i = o$ with $o \in \mathcal{O}$.* □

A temporal restriction defines the constraints w.r.t. the validity intervals of two predicates in a conjunctive pattern. The order of the predicates in a pattern defines a temporal order implicitly already. A predicate must have an earlier or identical start time as all its succeeding predicates. Therefore, we define $\mathcal{IR}_{older} \subseteq \mathcal{IR}$ including those temporal relations where the start time of the first interval s_1 is before the start time of the second interval s_2, i.e., $s_1 < s_2$ and for the "head to head" temporal relations we define $\mathcal{IR}_{\models} \subseteq \mathcal{IR}$ where the start times are equal, i.e., $s_1 = s_2$.

Definition 3.11 (Temporal Restriction) *The temporal restriction $\mathcal{TR} = \{\mathcal{TR}[1,2], \ldots, \mathcal{TR}[n-1, n]\}$ of a conjunctive pattern p with size n is defined as the set of pairwise temporal relations between all predicates. For each predicate pair $(pred_i, pred_j)$ of the pattern p where $pred_i$ appears before $pred_j$ in the pattern, i.e., $i < j$, the possible temporal relations between these two intervals are defined by the set $\mathcal{TR}[i, j]$. It must hold that $\forall tr_k \in \mathcal{TR}[i, j] : tr_k \in \mathcal{IR}_{older} \cup \mathcal{IR}_{\models}$ with $1 \leq i < n$ and $i < j \leq n$ due to the implicit temporal order of the predicates. If the name $pd_{name,j}$ of $pred_j$ is smaller than $pd_{name,i}$ of $pred_i$ w.r.t. a lexicographic order it must hold that $\forall tr_k \in \mathcal{TR}[i, j] : tr_k \in \mathcal{IR}_{older}$ in order to have a canonical representation of the sequences.* □

In the experiments described in section 6 we use just five temporal relations which can be seen as a condensed subset

B r_2 C / A r_1 B	$<$	$<_c$	$\models$	$>_c$	$>$
$<$	$<$	$<$	$<$	$<$, $<_c$, $\models$, $>_c$	$<$, $<_c$, $\models$, $>_c$, $>$
$<_c$	$<$, $<_c$	$<$, $<_c$	$<_c$	$<_c$, $\models$, $>_c$	$<_c$, $\models$, $>_c$, $>$
$\models$	$<$, $<_c$	$<$, $<_c$	$\models$	$>_c$	$>$
$>_c$	$<$, $<_c$	$<$, $<_c$, $\models$, $>_c$, $>$	$>_c$, $>$	$>_c$, $>$	$>$
$>$	$<$, $<_c$, $\models$, $>_c$, $>$	$>_c$, $>$	$>_c$, $>$	$>_c$, $>$	$>$

Table 1: Composition table for the temporal relations

of the temporal relations introduced by [Freksa, 1992] and [Allen, 1983]: before and after ($<$, $>$), older & contemporary and younger & contemporary ($<_c$, $>_c$), and head to head ($\models$). Thus, in our case $\mathcal{IR} = \{<, <_c, \models, >_c, >\}$, $\mathcal{IR}_{older} = \{<, <_c\}$, and $\mathcal{IR}_{\models} = \{\models\}$. The motivation for these temporal relations is due to keeping complexity low and still having the relevant temporal relations for setting up prediction rules. The composition table for these temporal relations is shown in Table 1.

Definition 3.12 (Temporal Pattern) *Temporal patterns $tp_i = (cp_i, \mathcal{TR}_i, cr_i)$ are defined as a 3-tuple of a conjunctive pattern $cp_i = ap_{i,1} \wedge ap_{i,2} \wedge \ldots \wedge ap_{i,size}$, a temporal restriction $\mathcal{TR}_i$, and a class restriction cr_i.* □

After having defined dynamic scenes, their schemata, and temporal patterns, we can define how to match such patterns to a dynamic scene. Pattern matching is essential for the computation of the support of a pattern. Basically, a match can be seen as a successful query to a database [Dehaspe, 1998]. In order to match a temporal pattern all predicates in the conjunction must be true (within a defined window size), the temporal restrictions between these predicates must be satisfied, and for the variable assignment the class restriction must not be violated.

Definition 3.13 (Pattern Match) *A match of pattern $p = (cp, tr, cr)$ is a valid assignment for each atomic pattern $p_i \in cp$ in the conjunctive pattern cp with size n to a corresponding (instantiated) predicate $p_{inst_i} \in \mathcal{P}$ of the dynamic scene where both predicate definitions of the atomic pattern $p_i = (pd_i, p_{i_{arg}})$ and the assigned predicate instance $p_{inst_i} = (pd_{inst_i}, p_{inst_{objects,i}}, \langle s_i, e_i \rangle)$ are identical, i.e., $pd_i = pd_{inst_i}$ and all arguments are pairwise unifiable. Furthermore, it must hold that no predicate instance is assigned more than once, i.e.: $\forall i, j : p_{inst_i} \neq p_{inst_j}$ with $i \neq j$ and $1 \leq i, j \leq n$.*

Additionally, the match must be within the sliding window range. Let w_s be the window's start position, w be the window size, $w_e = w_s + w$ be the window's end position, and $\mathcal{P}_{match}$ be the set of all predicate instances of the match. For all assigned predicate instances $p_{inst_j} \in \mathcal{P}_{match}$ with $p_{inst_j} = (pd_{inst_j}, p_{inst_objects_j}, \langle s_j, e_j \rangle)$ it must hold that $s_j < w_e$ and $e_j \geq w_s$, i.e., that the start time of the predicate instance has already passed and that it can still be seen within the window.

Furthermore it must hold that none of the restrictions is violated. Let $\mathcal{O}_{match} = (o_1, o_2, \ldots, o_m)$ be the list of objects in the assigned predicate instances and $cr = (c_1, c_2, \ldots, c_m)$ the class restriction of the pattern. Then it must hold that $\forall i : instanceof_{trans}(o_i, c_i)$ with $1 \leq i \leq m$ where $instanceof_{trans}$ is a transitive instance-of relation utilizing the class hierarchy defined by sc in $\mathcal{DSS}$.

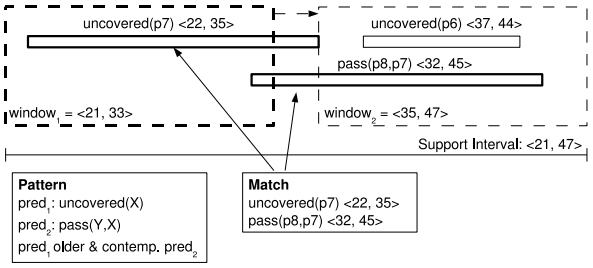


Figure 1: Pattern matching example

In order to satisfy the temporal restriction tr it must hold that $\forall r, s : ir(\langle s_r, e_r \rangle, \langle s_s, e_s \rangle) \in \mathcal{TR}[r, s]$ with $1 \leq r < n$ and $r < s \leq n$. □

As the frequency of a pattern is directly related to its support we first introduce how the support is computed in our case. In the task of frequent pattern discovery in logic, [Dehaspe, 1998] introduced an extra *key* parameter in order to determine what is counted. Entities are uniquely identified by each binding of the variables in key [Dehaspe, 1998, p. 34]. A disadvantage of this support definition is that the key parameter must be part of each pattern in order to get a support > 0. Thus, it is not possible to compare two different patterns if they do not share this key parameter.

We decided to use the observation time semantic for support computation as stated by Höppner. Here, the support is defined as "the total time in which (one or more) instances of P can be observed in the sliding window" [Höppner, 2003, p. 52]. The advantages of using observation time as support are the clear semantics and the better efficiency as not all matches have to be collected or maybe even further processed. The monotonicity property for this support definition holds and the support intervals of previous steps (i.e., of more general patterns) can be reused in order to restrict the search to parts of the temporal sequence in the subsequent levels.

Definition 3.14 (Support) *Let p be a temporal pattern, ds the dynamic scene, and $\mathcal{M}$ the set of matches. The validity interval of a single match $m_i \in \mathcal{M}$ is defined as $v_i = [s_{max_i} - w + 1, e_{min_i} + w]$ with s_{max_i} being the maximal start time and e_{min_i} the minimal end time of all predicate instances in m_i. The support of p w.r.t. ds is defined as the length of the union of all validity intervals of the matches:* $supp(p) = length\left(\bigcup_{k=1}^{|\mathcal{M}|} v_k\right)$. □

This support definition computes the length of intervals where at least one match for a pattern can be found for a given window size. The frequency is the probability to find a match of a pattern at a random window position for a given dynamic scene and window size (cf. [Höppner, 2003]).

If the support value is divided by the sequence length of the dynamic scene plus the two times the window size minus one (sliding window at the start and the end of the sequence; the window must include the start time of the first interval in order to match a pattern) we get the frequency of the pattern, i.e., $freq(p) = \frac{supp(p)}{seq_{length}+2w-1}$.

Fig. 1 illustrates the matching of a pattern and the covered support interval by this match ($\langle 21, 47 \rangle$). The pattern in this examples matches the first time at window start position 21 when `pass(p8, p7) <32,45>` is visible in the window. It still matches as long as no end time point of a predicate in the match was left behind the window.

Algorithm 1 *MiTemP-main* (Pattern Generation)

Input: $ds = (\mathcal{P}, \mathcal{O}, i, \mathcal{DSS})$, win_{size}, $size_{min}$, $size_{max}$, $minfreq$ /* dynamic scene, window size, minimal and maximal pattern size, minimal frequency */

Output: All frequent patterns $\mathcal{P}_{freq}$ with $size$, $size_{min} \leq size \leq size_{max}$

1: Init $\mathcal{P}_{freq} = \emptyset, i = 1$
2: $C_i \leftarrow create\_single\_predicate\_patterns()$ /* Create one candidate for each predicate definition */
3: **while** $C_i \neq \emptyset$ **do**
4: $support[|C_i|] \leftarrow$ MiTemP-support(ds, win_{size}, C_i)
5: $L_i = \{c_j \in C_i | \frac{support(c_j)}{max\_supp} \geq minfreq\}$
6: $\mathcal{P}_{freq} \leftarrow \mathcal{P}_{freq} \cup \{l \in L_i | size_{min} \leq size(l) \leq size_{max} \wedge complete\_temporal\_restriction(l)\}$
7: $i \leftarrow i + 1$
8: $C_i =$ *MiTemP-gen-lengthening*(L_{k-1}) $\cup$ *MiTemP-gen-temp-refinement*(L_{k-1}) $\cup$ *MiTemP-gen-unification*(L_{k-1}) $\cup$ *MiTemP-gen-class-refinement*(L_{k-1}) $\cup$ *MiTemP-gen-instantiation*(L_{k-1})
9: **end while**

Algorithm 2 *MiTemP-gen-lengthening* (Lengthening Candidate Generation)

Input: $\mathcal{L}_{i-1}$ /* Frequent patterns of the previous step */
Output: New candidate patterns C_i

1: $\mathcal{F}_{i-1} = \{l \in \mathcal{L}_{i-1} | lengthening\_allowed(l)\}$
2: **for** $(p_i \in \mathcal{F}_{i-1})$ **do**
3: **for** $(p_j \in \mathcal{F}i - 1), j \geq i$ **do**
4: **if** $p_i = (ap_{i,1} \wedge ap_{i,2} \wedge \ldots \wedge ap_{i,i-2} \wedge ap_{i,i-1}) \wedge p_j = (ap_{i,1} \wedge ap_{i,2} \wedge \ldots \wedge ap_{i,i-2} \wedge ap_{j,i-1})$ **then**
5: $p_{new1} = (ap_{i,1} \wedge \ldots \wedge ap_{i,i-2} \wedge ap_{i,i-1} \wedge ap_{j,i-1})$
6: $p_{new2} = (ap_{i,1} \wedge \ldots \wedge ap_{i,i-2} \wedge ap_{j,i-1} \wedge ap_{i,i-1})$
7: /* Add if all subsets are frequent (prune step) */
8: **if** $\forall p_{sub} \subset p_{new1} : p_{sub} \in \mathcal{L}_{i-1}$ **then**
9: $\mathcal{C}_i \leftarrow \mathcal{C}_i \cup p_{new1}$
10: **end if**
11: **if** $\forall p_{sub} \subset p_{new2} : p_{sub} \in \mathcal{L}_{i-1}$ **then**
12: $\mathcal{C}_i \leftarrow \mathcal{C}_i \cup p_{new2}$
13: **end if**
14: **end if**
15: **end for**
16: **end for**

The goal of this work is to identify all frequent temporal patterns from a dynamic scene. In order to restrict the search space we introduce upper and lower limits for the number of predicates, i.e., for the minimal and maximal size of the conjunctive pattern, and force the temporal restriction to be completely constrained, i.e., all temporal relation sets must consist of exactly one element. If a pattern is frequent and it satisfies these conditions we refer to it as a *relevant frequent pattern*. The set of these patterns forms the language $\mathcal{L}_{MiTemP} = \{tp | tp = (cp, \mathcal{TR}, cr) \wedge freq(tp) \geq minfreq \wedge size_{min} \leq |cp| \leq size_{max} \wedge \forall i, j : |\mathcal{TR}[i, j]| = 1$ with $1 \leq i < |cp|$ and $i < j \leq |cp|\} \cup \epsilon$ with $|cp| > 1$. The most general empty pattern is denoted by ϵ.

4 *MiTemP*: Mining Temporal Patterns

This section introduces the *MiTemP* (Mining Temporal Patterns) algorithms. All algorithms are shown in pseudo code while the implementation has been realized with *XSB Prolog* [Sagonas *et al.*, 2006]. The main loop for the level-wise refinement is shown in Algorithm 1 (*MiTemP-main*). As mentioned above temporal patterns consist of different components like the conjunctive pattern, temporal restrictions, variable unification, class restrictions, and instantiations. For each of these components refinement operators exist that specialize a given pattern. In order to set up an optimal refinement operator which creates every pattern only once a status about the executed refinements is affixed to each pattern as it is also done by [Lee, 2006]. The status keeps track of how many refinements of each type have been performed and which position (predicate pair for temporal refinement, variable position for unification, class restriction, or instantiation) has been processed at last. A status $status(p) = (l, t, t_{last}, u, u_{last}, c, c_{last}, i, i_{last})$ where l, t, u, c, i are the number of refinement operations of the different types, namely lengthening, temporal refinement, unification, class refinement, and instantiation; $t_{last}, u_{last}, c_{last}, i_{last}$ refer to the last position where a temporal refinement, unification, class refinement, or instantiation has been performed. Similar to [Lee, 2006] we define the following refinement operations:

- Lengthening $\rho_L(p)$: Adding an atom to the end of a conjunctive pattern
- Temporal refinement $\rho_T(p)$: Adding a temporal constraint between two predicates in the conjunctive pattern at the leftmost position after the previous temporal refinement
- Unification $\rho_U(p)$: Unifying a variable v_j with a previous one v_i $(i < j)$ in the conjunctive pattern where no variable v_k with $k > j$ has been unified before
- Class refinement $\rho_C(p)$: Specializing a class c_i in the class restriction for the variables of the conjunctive pattern where no class restriction has been performed to any c_j with $j > i$.
- Instantiation $\rho_I(p)$: Instantiating a variable v_i of the conjunctive pattern with an instance $o \in \mathcal{O}$ where no variable v_j has been instantiated with $j > i$. It must also hold, that no variable v_k with $k \neq i$ has been instantiated to o.

The refinement operator is defined as the union of these operators: $\rho(p) = \rho_L(p) \cup \rho_T(p) \cup \rho_U(p) \cup \rho_C(p) \cup \rho_I(p)$. While [Lee, 2006] also defines a "deepening" operator (for replacing a variable by a functor) which is omitted here we introduce the class refinement operator which exploits the class hierarchy of the dynamic scene schema. Another difference is that the temporal refinement here adds arbitrary temporal relatios between time intervals while the "promotion" operator of Lee replaces the *before* relation between two events by a *directly before* relation.

Certain rules for each refinement coordinate when which refinement step is allowed. The lengthening operation is just allowed as long as no other refinement type has been applied and the maximal size of the conjunctive pattern is not exceeded. In order to perform a temporal refinement the minimum pattern size must be met, and no other refinement (except lengthening) must have been applied to the pattern. As we are looking for temporally completely constrained patterns the temporal refinement is only allowed to refine the next not yet processed predicate pair in the sequence. In the refinement step itself one of the possible temporal relations $tr \in \mathcal{TR}[i, j]$ is selected. After this step the composition table (Table 1) is used to further restrict the following temporal relations. Only those patterns where all predicate pairs are restricted to one temporal relation are further processed by other refinement types.

Algorithm 3 *MiTemP-support* (Support Computation)

Input: $ds = (\mathcal{P}, \mathcal{O}, i, \mathcal{DSS})$, win_{size}, $\mathcal{PL}$ /* dynamic scene, window size, pattern list */

Output: Support values $support(p_i)$ for all patterns $p_i \in \mathcal{PL}$

1: /* s_{min} is the earliest start and e_{max} is the latest end time */
2: Init $supp\_intervals(p_i) = \emptyset$, $next\_to\_check(p_i) = -\infty$
3: Init $\mathcal{P}_{win} = \emptyset$, $w_{start} = s_{min} - win_{size} + 1$
4: **while** $w_{start} \leq e_{max}$ **do**
5: **for** $p_i \in \mathcal{PL}$ **do**
6: **if** $(potential\_match(p_i, w_{start}) \wedge next\_to\_check(p_i) \leq w_{start})$ **then**
7: $m \leftarrow pattern\_match(p_i, w_{start}, win_{size})$
8: **if** $m \neq null$ **then**
9: $supp\_intervals(p_i) \leftarrow supp\_intervals(p_i) \cup get\_support\_interval(m)$ /* Add newly covered interval */
10: $next\_to\_check(p_i) \leftarrow get\_min\_end\_time(m)$
11: **end if**
12: **end if**
13: $w_{start} \leftarrow w_{start} + 1$
14: **end for**
15: **end while**
16: **for** $p_i \in \mathcal{PL}$ **do**
17: $support(p_i) \leftarrow length(supp\_intervals(p_i))$
18: **end for**

The three remaining refinements – unification, class restriction, and instantiation– are also ordered, i.e., if a refinement has been applied it is not allowed to use any of the preceding refinement types any more. Within each of the refinements only variable positions after the the last refinement are allowed to be processed. For instance, if the third variable in a conjunctive pattern has been instantiated just the fourth or later variables can be used for further instantiation steps. Variables are only unified with one out of the set of preceeding variables. Unified variables are left out at the remaining refinement steps (as they are processed implicitely if the unified counterpart is restricted). At the class refinements the current class of a variable is specialized to one of its direct subclasses. Instantiations are only performed for variables which already refer to a leaf class in the class refinement. In all cases after a refinement as much information as possible is derived: after unification of two variables the class restriction at the corresponding positions is set to the more special class, after instantiation to instance o all variable positions in the class restriction are set to the corresponding class $i(o)$.

Due to space restrictions we can only sketch the proof for optimality of our refinement operator. For a complete proof the inverse refinement operators must be defined formally and it must be shown that no pattern is missed by creating the most special representation of a pattern after refinement. An optimal refinement operator is one that satisfies completeness and non-redundancy properties [Lee, 2006]. Analogical to Lee's proof it can be shown that these two properties hold; for more details see [Lee, 2006, p.97-99]. The inverse operations of the refinement operators $\rho^{-1}(p) = \rho_L^{-1}(p) \cup \rho_T^{-1}(p) \cup \rho_U^{-1}(p) \cup \rho_C^{-1}(p) \cup \rho_I^{-1}(p)$ are mutually exclusive as – depening on the counters in the pattern status $status(p)$ – just one of the inverse operations can be applied. Each of these operations itself leads to a single more general pattern which follows from their definitions (at $\rho_L^{-1}(p)$ just the last element can be removed, at $\rho_T^{-1}(p)$ the last restricted position is generalized to the set of all allowed temporal relations, at $\rho_U^{-1}(p)$ the mostright unified variable is split and replaced by a new variable, at $\rho_C^{-1}(p)$ the mostright restricted class is replaced by its single super class, at $\rho_I^{-1}(p)$ all occurences of the rightmost instantiated variable are replaced by a new variable). Assuming there exist two different paths (i.e., redundancy is given) from the most general empty pattern to a pattern $p \in \mathcal{L}_{MiTemP}$, $r_0 = \epsilon, r_1, \ldots, r_m = p$ and $s_0 = \epsilon, s_1, \ldots, s_n = p$ with $r_{i+1} = \rho(r_i)$ and $s_{i+1} = \rho(s_i)$. If the inverse refinement operator is applied to both r_m and s_n the resulting sequences must be identical due to the property of the inverse refinement operator with $r_n = s_n, r_{n-1} = s_{n-1}, \ldots r_1 = s_1, r_0 = s_0 = \epsilon$ and $m = n$ which contradicts the assumption of the two different paths. Thus, it follows that ρ is non-redundant.

```
directSubClassOf(team1,                holds(pass(p7,p6),27,29).
                 object).              holds(closerToGoal(p6, p7),23,31).
directSubClassOf(team2,                holds(uncovered(p6,p6),25,33).
                 object).              holds(uncovered(q9,q9),30,36).
directInstanceOf(p6, team1).           holds(closerToGoal(q8,q6),36,40).
directInstanceOf(p7, team1).           holds(pass(p9,p7),39,41).
directInstanceOf(p8, team1).           holds(closerToGoal(p7,p9),35,43).
directInstanceOf(p9, team1).           holds(uncovered(q8,q8),42,44).
directInstanceOf(q6, team2).           holds(uncovered(p7,p7),37,45).
directInstanceOf(q7, team2).           holds(pass(p8,p6),51,53).
directInstanceOf(q8, team2).           holds(closerToGoal(q7,q6),50,54).
directInstanceOf(q9, team2).           holds(closerToGoal(p6,p8),47,55).
holds(uncovered(q6,q6),12,14).         holds(uncovered(p6,p6),49,57).
holds(pass(p9,p8),15,17).              holds(pass(p8,p7),65,67).
holds(closerToGoal(p8,p9),11,19).      holds(uncovered(q6,q6),58,68).
holds(uncovered(p8,p8),13,21).         holds(closerToGoal(p7,p8),61,69).
holds(closerToGoal(q8,q9),16,26).      holds(uncovered(p7,p7),63,71).
```

Figure 2: Example input for evaluation

In order to show completeness it is necessary to proof that for each pattern $p \in \mathcal{L}_{MiTemP}$ a path $p_0 = \epsilon, p_1, \ldots, p_n = p$ exists. Here, again the inverse refinement operator and the status can be used. Let p be any pattern in $\mathcal{L}_{MiTemP}$. This pattern has the status $status(p)$ with the refinement level $n = |status(p)|$. If we get $p_{n-1} = \rho^{-1}(p_n)$ then $p_n \in \rho(p_{n-1})$ and $|status(p_{n-1})| + 1 = |status(p)|$. Referring to Lee we can find any p_i with $0 \leq i \leq n-1$ by applying $p_{i-1} = \rho^{-1}(p_i)$ and we know that $|status(p_i)| = i$ and $|status(p_0)| = 0$. By definition, the empty pattern is the only one with a refinement level of 0. Thus, we have found a sequence $p_0 = \epsilon, p_1, \ldots, p_n = p$ with $p_{i+1} \in \rho(p_i)$.

The candidate generation algorithm for lengthening differs from the other refinements as it is not applied to each pattern separately but to the set of frequent patterns of the previous step. The algorithm (Algorithm 2) is similar to *apriori-gen* [Agrawal and Srikant, 1994]. Starting from single predicate patterns in each following step patterns with the same $n-1$ prefix are combined in order to create new pattern candidates (cf. [Lee, 2006]). The difference here is that the "items" in the list are actually predicates which can appear multiple times in a conjunctive pattern. As the predicate order is also relevant for distinguishing the patterns no alphanumeric order can be used to just create one new candidate of two previous frequent patterns with identical prefix. Here, two patterns must be generated.

Algorithm 3 shows the support computation procedure. Input to the algorithm are the dynamic scene, the size of the sliding window, and the list of patterns to check. As long as the latest end time is not reached a window is moved over the sequence. At each window position just the "visible" predicates identified by the window position are taken into account for pattern matching. This has the advantage that during pattern matching many assignments do not need to be checked as they are out of range of the sliding window anyway. If a match is found for a pattern at the current window position the support interval list is extended by the support interval of the match and the next position to check

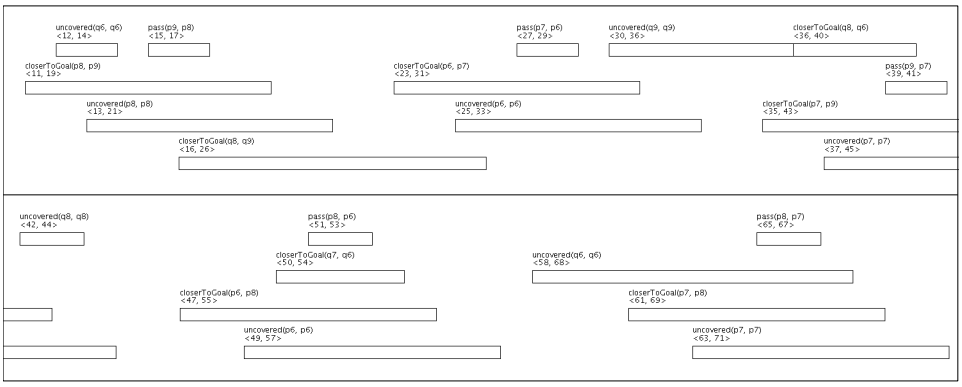


Figure 3: Test scenario

is assigned. If the match is valid beyond the window border some pattern matching steps can be omitted before the pattern has to be checked again. Finally, the window position is moved to the next position.

5 Learning Temporal Patterns with *WARMR*

As the temporal validity intervals of predicates can be seen as just another dimension of relations it should be possible to transfer the learning problem to relational association rule mining. Intuitively, it seems to be unhandy but feasible to add information about start and end time to every predicate. We developed a converter which automatically transfers a *MiTemP* input file to *ACE* input files. *ACE* is a data mining system which provides a number of different relational data mining algorithms including *WARMR* [Blockeel *et al.*, 2002; 2006]. Different problems had to be solved in order to set up *WARMR* to mine the same frequent patterns (with identical support calculation) as it is done by *MiTemP*. Due to space restrictions it is not possible to go into detail how the *WARMR* input is generated. A separate report capturing these details is currently written.

The transformation of the class hierarchy and corresponding instances is straight forward. The `directSubClassOf` and `directInstanceOf` relations can be kept and put to *ACE*'s knowledge base file. The transitive clauses for querying instances of classes and subclasses of a class can also be left unchanged and put into the background knowledge file. The `holds` predicates representing the validity intervals of relations are now represented by relations with an additional argument which stands for the time interval. The predicate instance $holds(pass(p8, p7), \langle 32, 45 \rangle)$ is converted to `pass(1, p8, p7, i(32, 45))` where the first argument is a unique predicate ID.

For setting up the learning bias in *WARMR* it is possible to define *rmode* statements. These statements define how a query can be extended during the generation of new query candidates. It is also possible to define constraints which must be satisfied in order to add an atom to the query. More details about the *rmodes* can be found, for instance, in Dehaspe's doctoral thesis and the *ACE* user's manual [Dehaspe, 1998; Blockeel *et al.*, 2006].

For each given *MiTemP* refinement as described in section 4 *rmodes* must be defined. For lengthening a *rmode* must be defined for each predicate definition. In order to avoid the same predicate instance being used more than once it must be guaranteed that the predicate ID variable differs from all other predicate ID variables of this query.

Temporal relations between intervals are represented by clauses which check if the temporal relation actually holds for the interval pair, i.e., for each temporal relation a clause exists and a *rmode* is created. In order to refine a pattern by adding a temporal constraint one of the temporal clauses is added to the query by relating two intervals of existing predicates of the query to each other.

Unification is handled by a special unification clause which unifies two existing variables in the previous query. The *rmode* declarations of *ACE* also provide means to define *rmodes* which do not introduce a new variable in the new atom but reuse an existing one. However, our intended solution should also cover the instantiation of variables (i.e., using constants). Setting up *rmodes* for all cases (unification, constants, and new variables) and their combinations in predicates with an arbitrary (potentially large) number of arguments would have lead to a huge number of *rmodes* for the predicates. Thus, if a new predicate is added to the query all arguments are new variables in the beginning. These can be unified with another variable or can be bound to a constant in further refinement steps.

For instantiation a *rmode* definition allows a variable to be unified with an instance. The set of instance candidates depends on the predicate where the variable occurs. Only those instances are taken into account which appear at least in one of the predicates at the variable's position in the dynamic scene, i.e., no "impossible" query will be generated.

Class refinement is performed by adding `instanceOf` predicates, constraining a variable to a certain class (or one of its sub classes). A constraint definition makes sure that for each variable just one `instanceOf` predicate will be added. Additional constraints ensure that a variable will be used just for instantiation or class refinement and that unified variables are not refined at all.

Setting up *WARMR* for computing the support as intended was a little bit trickier. *WARMR* needs a counting attribute which is used for support computation, i.e., the number of different values of this attribute where a query matches determines the support of the query. In our case the support is defined to be the number of temporal positions where within a sliding window a pattern holds. In order to let *WARMR* compute the intended support a predicate `currentIndex` has been introduced and used as counting attribute. For each existing temporal position a predicate is created in the knowledge base file. In combination with another predicate representing the window position (`inWindowPos`) for each temporal position it can be checked if a pattern holds.

Some tricks have made it possible to use *WARMR* as in-

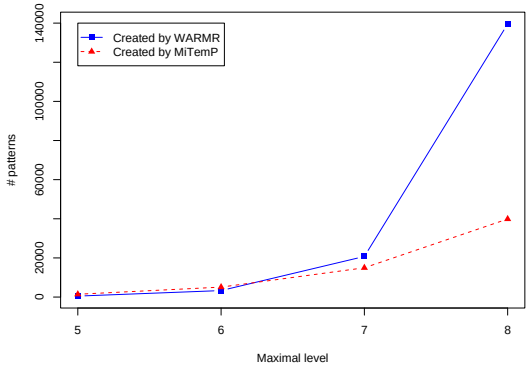


Figure 4: Number of created patterns

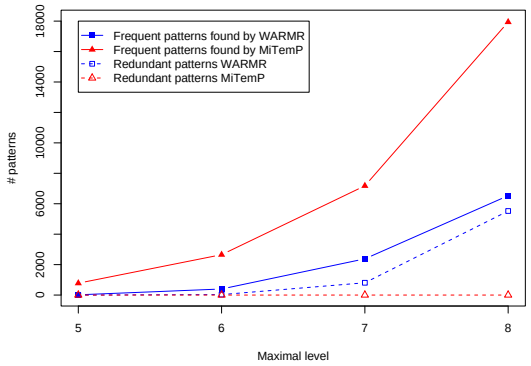


Figure 5: Number of frequent and redundant patterns

tended for mining temporal patterns. However, we had to accept some compromises in the solution. Computing the support is a little bit more inefficient as necessary as there is no way to use a real sliding window which covers an extended interval. In the current solution each time position has to be checked on its own (even if it can theoretically be known that the pattern holds at the current position due to the sliding window of an earlier position) and also predicate instances out of scope of the window might be checked.

Another problem is that redundant patterns are generated by *WARMR*. To the best of our knowledge it is not possible to avoid that for a unification two patterns are generated ($A = B$ and $B = A$). Furthermore, different representations can be generated for the same pattern if additional restrictions could be derived from the pattern (e.g., temporal relations using the composition table, class restrictions which should be specialized due to unification of variables as it is done by *MiTemP*).

In the case of *MiTemP* we required each pattern to be completely constraint w.r.t. temporal relations, i.e., that for each predicate pair exactly one temporal relation should be assigned. To the best of our knowledge it is not possible to define a constraint in *ACE* which guarantees to just create patterns which satisfy this property. This leads to the generation of some patterns which are out of scope of *MiTemP*.

Even though *WARMR* might have some drawbacks for our temporal pattern mining task it should be stated clearly that *WARMR* is not a special solution for mining temporal patterns from such an interval representation but a generic system for mining frequent queries which also can be used to mine queries representing temporal patterns.

6 Evaluation

The experiments with *WARMR* and *MiTemP* have different goals. First of all, it is a proof of concept that both approaches can be used to mine frequent temporal patterns. In order to find out if both approaches lead to the same support values the frequencies of all common patterns are compared. Furthermore, it is expected that constraining the search space at the refinement steps of the algorithm reduce the number of generated patterns a lot.

For the evaluation a simple soccer scenario has been used (Fig. 2). Different objects in the dynamic scene are objects `p6` - `p9` of class `team1` and objects `q6` - `q9` of class `team2`. Relations between these objects can be `uncovered`, `closerToGoal`, and `pass`. Fig. 3 shows the temporal validity intervals of the relations between the objects (time proceeding from left to right).

```
WARMR:
freq(8,96635,
     [currentIndex(A),getWindowPos(A,B),pass(C,D,E,F,B),
      uncovered(G,H,I,J,B),not(G=C),unif(I,H),unif(I,E),
      olderContemp(J,F),instanceOf(I,team1)],0.903614457831325).

MiTemP:
pattern(4544,
        [uncovered(_h6502661,_h6502661),
         pass(_h6502666,_h6502661)],temp(tr([olderContemp])),
         classRestr(team1,team1,object,team1),
         status(2,1,temp(1,2),2,varPos(2,2),0,varPos(-1,-1),
                1,varPos(1,1))) [Freq: 0.9036]
```

Figure 6: Example for a learned pattern

In different experiments the maximal refinement level of *WARMR* and *MiTemP* has been altered from five to eight. As *WARMR* could not create any complete pattern as defined above with a maximal refinement level below five these settings have been left out here. Table 2 summarizes the results of the test runs. Besides the number of created, redundant, and frequent patterns for both approaches it is also shown how many patterns have just been found by one approach up to this level and how many common patterns have been found by both approaches. The last two columns show the coverage, i.e., how many patterns of the other approach are covered at the current level. Fig. 4 shows a graph comparing the number of generated patterns at the different maximal refinement levels. Fig. 5 compares the number of mined frequent patterns and the number of redundant patterns of both approaches for the different levels.

All common patterns of *WARMR* and *MiTemP* get identical frequency values assigned. Fig. 6 shows example outputs of the same pattern by *WARMR* and *MiTemP*. While *WARMR* creates a number of redundant patterns (growing with an increasing maximal refinement level) the refinement operators in *MiTemP* are optimal as no redundant pattern was created (dotted lines in Fig. 5). *MiTemP* identifies much more frequent patterns at the different maximal refinement levels and creates less patterns at maximal refinement levels seven and eight– at levels five and six *MiTemP* creates more patterns. While the fraction of relevant frequent patterns to created patterns is quite low with *WARMR* (at level eight it is $\frac{6530}{139543} = 4.68\%$) almost every second pattern created by *MiTemP* is a relevant frequent one (at level eight: $\frac{17940}{39889} = 44.98\%$).

In the eighth level some patterns which have been mined by *WARMR* have not yet been found by *MiTemP* at this level. An inspection of the patterns has shown that these are patterns with many instantiations. Due to the refinement structure in *MiTemP* a variable is not instantiated before the class refinement restricts the variable to a leaf class (i.e., having no sub classes). In the *WARMR* solution in-

Max. level	#created *WARMR* patterns	#frequent *WARMR* patterns	#redundant *WARMR* patterns	#created *MiTemP* patterns	#frequent *MiTemP* patterns	#redundant *MiTemP* patterns	#common patterns *WARMR*/ *MiTemP*	#unique patterns *WARMR*	#unique patterns *MiTemP*	Coverage of *MiTemP* patterns in *WARMR*	Coverage of *WARMR* patterns in *MiTemP*
5	498	16	0	1436	779	0	16	0	763	$\frac{16}{779} = 2.05\%$	$\frac{16}{16} = 100.0\%$
6	3317	399	31	5087	2653	0	399	0	2254	$\frac{399}{2653} = 15.04\%$	$\frac{399}{399} = 100.0\%$
7	20754	2380	807	14962	7178	0	2380	0	4798	$\frac{2380}{7178} = 33.16\%$	$\frac{2380}{2380} = 100.0\%$
8	139543	6530	5523	39889	17940	0	6340	190	11600	$\frac{6340}{17940} = 35.34\%$	$\frac{6340}{6530} = 97.09\%$

Table 2: Results of the test runs with different maximal refinement levels

stantiation can be performed directly to a variable, i.e., the intermediate class refinement steps (specializing variables to `team1` or `team2`) are not needed and thus, some patterns can be created at an earlier refinement level.

7 Conclusion

In this paper we have presented an approach to temporal pattern mining which mines frequent patterns from time interval-based relational representations of dynamic scenes. An *Apriori* like algorithm has been introduced which performs a top-down search of the pattern space without multiple generation of patterns. The use of reasoning techniques creates the most specialized representation after refinement or identifies inconsistencies in patterns. This avoids the creation of "impossible" patterns which cannot be frequent as well as reduces the number of specialization steps which are implicit in the pattern already. Here, a composition table is used for identifying possible temporal refinements, and class information of variables is used to find the most special class of a variable by taking into account predicate definitions, variable unifications, and instantiations. An implementation of the well-known *WARMR* algorithm has been used to create another solution for the mining problem. The experiments have shown the advantage of avoiding the creation of impossible or redundant patterns in *MiTemP*. While less patterns had to be explored at refinement levels seven and eight much more frequent patterns have been found by *MiTemP* already. This can be particularly of importance if large sequences have to be processed, i.e., if support computation is costly.

Acknowledgment

We would like to thank Frank Höppner at the Fachhochschule Braunschweig/Wolfenbüttel for helpful discussions on support computations in temporal pattern mining. We also want to express our gratitude to the members of the DTAI research group of the KU Leuven, Belgium, for providing the *ACE* system including *WARMR* [Blockeel *et al.*, 2002]. Particularly, we would like to thank Jan Struyf for his great support with *ACE*/*WARMR*.

References

[Agrawal and Srikant, 1994] Rakesh Agrawal and Ramakrishnan Srikant. Fast algorithms for mining association rules. In *Proceedings of the 20th International Conference on Very Large Data Bases, VLDB*, pages 487–499, September 1994.

[Agrawal *et al.*, 1993] Rakesh Agrawal, Tomasz Imielinski, and Arun N. Swami. Mining association rules between sets of items in large databases. In *Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data*, pages 207–216, Washington, D.C., May 1993.

[Allen, 1983] James F. Allen. Maintaining knowledge about temporal intervals. *Communications of the ACM*, 26(11):832–843, November 1983.

[Blockeel *et al.*, 2002] Hendrik Blockeel, Luc Dehaspe, Bart Demoen, Gerda Janssens, Jan Rammon, and Henk Vandecasteele. Improving the efficiency of inductive logic programming through the use of query packs. *Journal of Artificial Intelligence Research*, 16:135–166, 2002.

[Blockeel *et al.*, 2006] Hendrik Blockeel, Luc Dehaspe, Jan Rammon, Jan Struyf, Anneleen Van Assche, Celine Vens, and Daan Fierens. *The ACE Data Mining System, User's Manual*. Katholieke Universiteit Leuven, Belgium, February 16 2006.

[Dehaspe and Toivonen, 1999] Luc Dehaspe and Hannu Toivonen. Discovery of frequent DATALOG patterns. *Data Mining and Knowledge Discovery*, 3(1):7 – 36, March 1999.

[Dehaspe, 1998] Luc Dehaspe. *Frequent Pattern Discovery in First-Order Logic*. PhD thesis, Katholieke Universiteit Leuven, Belgium, 1998.

[Freksa, 1992] Christian Freksa. Temporal reasoning based on semi-intervals. *Artificial Intelligence*, 54(1–2):199–227, 1992.

[Höppner, 2003] Frank Höppner. *Knowledge Discovery from Sequential Data*. PhD thesis, Technische Universität Braunschweig, 2003.

[Kaminka *et al.*, 2003] Gal Kaminka, Mehmet Fidanboylu, Allen Chang, and Manuela Veloso. Learning the sequential coordinated behavior of teams from observation. In Gal Kaminka, Pedro Lima, and Raul Rojas, editors, *RoboCup 2002: Robot Soccer World Cup VI, LNAI 2752*, pages 111–125, Fukuoka, Japan, 2003.

[Lattner *et al.*, 2006] Andreas D. Lattner, Andrea Miene, Ubbo Visser, and Otthein Herzog. Sequential pattern mining for situation and behavior prediction in simulated robotic soccer. In Bredenfeld et al., editor, *RoboCup-2005: Robot Soccer World Cup VIII*, pages 118–129. Springer Verlag, Berlin, 2006. LNCS 4020.

[Lee, 2006] Sau Dan Lee. *Constrained Mining of Patterns in Large Databases*. PhD thesis, Albert-Ludwigs-Universität Freiburg, 2006.

[Mannila *et al.*, 1997] Heikki Mannila, Hannu Toivonen, and A. Inkeri Verkamo. Discovery of frequent episodes in event sequences. *Data Mining and Knowledge Discovery*, 1:259–289, 1997.

[Sagonas *et al.*, 2006] Konstantinos Sagonas, Terrance Swift, David S. Warren, Juliana Freire, Prasad Rao, Baoqiu Cui, Ernie Johnson, Luis de Castro, Rui F. Marques, Steve Dawson, and Michael Kifer. *The XSB System Version 3.0 - Volume 1: Programmer's Manual, Volume 2: Libraries, Interfaces, and Packages*, 2006.

Der Vortrag auf den Seiten 322-329

Daniel A. Keim, Daniela Oelke, Royal Truman and Klaus Neuhaus
Using Visual Analysis to Explore a Set of Functionally Equivalent Proteins

ist erschienen unter dem Titel

Daniel A. Keim, Daniela Oelke, Royal Truman, Klaus Neuhaus:
Finding Correlations in Functionally Equivalent Proteins by Integrating Automated and Visual Data Exploration, Proc. of 6th IEEE Symposium on BioInformatics and BioEngineering (BIBE 2006), November-Dezember, 2006.

Sound Multi-objective Feature Space Transformation for Clustering

Ingo Mierswa and Michael Wurst
Department of Computer Science
University of Dortmund

Abstract

In this work we propose a novel, generalized framework for feature space transformation in unsupervised knowledge discovery settings. Unsupervised feature space transformation inherently is a multi-objective optimization problem. In order to facilitate data exploration, transformations should increase the quality of the result and should still preserve as much of the original data set information as possible. We exemplify this relationship on the problem of data clustering. First, we show that existing approaches to multi-objective unsupervised feature selection do not pose the optimization problem in an appropriate way. Furthermore, using feature selection only is often not sufficient for real-world knowledge discovery tasks. We propose a new, generalized framework based on the idea of information preservation. This framework enables feature selection as well as feature construction for unsupervised learning. We compare our method against existing approaches on several real world data sets.

1 Introduction

Many knowledge discovery problems cannot be solved accurately by using the original feature space. This is due to several factors as noise, redundancy, sparsity or the fact that standard learning algorithms cannot represent necessary complex feature relationships. Both supervised and unsupervised knowledge discovery therefore depend on methods that transform the feature space in an appropriate way.

For supervised learning a set of labeled data points must be given. The learning method should merely find a function which *predicts* the label for unseen data points. The feature space transformation problem can be solved by finding a minimal feature space that maximizes the expected prediction accuracy.

Unsupervised machine learning differs essentially from supervised learning. The aim is usually rather to *describe* the data set and thus to automatically find inherent, natural patterns. Feature space transformation is important for unsupervised learning as well. Noise, sparsity and redundancy can hide the natural patterns in a data set, just as they can hide the relationship of the data points to a target function in supervised learning. There are several approaches that try to identify promising feature sets for unsupervised learning with respect to a task related criterion [Roth and Lange, 2003] but not with respect to a particular clustering algorithm. In addition, feature selection is a limited form of feature space transformation. It can for example not solve the problems of sparsity and feature interaction. Methods for feature space reduction can be applied to reduce noise and sparsity before applying unsupervised learning, e. g. Kernel-PCA [Schölkopf and Smola, 2002]. However, selecting appropriate parameters for new data sets is non trivial. This is especially problematic as such methods lead to feature spaces that are hard to interprete and resulting patterns are even harder to analyze. This is of course a clear conflict to the main target of cluster analysis.

The main limitation of these approaches is, however, that they do not reflect that feature space transformation for unsupervised learning inherently is a multi-objective optimization problem. Multi-objective problems are defined by several conflicting goals leading to the notion of Pareto optimal solutions. Several multi-objective wrapper approaches for unsupervised feature selection were proposed in [Kim *et al.*, 2000; 2002; Morita *et al.*, 2003]. These approaches minimize the number of features. Simultaneously, the quality of the identified patterns should be maximized. This idea is directly transferred from supervised multi-objective feature selection [Emmanouilidis *et al.*, 2000]. Figure 1 depicts the resulting Pareto front for a supervised feature selection problem. The used data set consists of 10 features necessary for classification and 10 additional noise features. It can clearly be seen that almost the complete range of solutions is covered, ranging from solutions containing only one feature and providing a small classification accuracy to a solution consisting of nine features with the highest accuracy. Adding the last non-noise feature or even noisy features would not lead to an improvement of accuracy and would therefore not lead to further Pareto optimal solutions. Hence, the Pareto front can not only be used as a feature selection method but also as a feature ranking method for the selected features.

While the basic idea of these approaches is very promising, they are limited in two points. First, minimizing the number of features just as in supervised learning is not robust for the unsupervised setting. Under very weak assumptions the set of Pareto optimal solutions collapses into one singular point that represents a trivial solution. Second, merely selecting features is not sufficient for many data mining tasks. In order to deal with problems as sparsity, new features must be constructed. We propose a new, generalized framework that approaches both problems. The quality of the resulting patterns should be optimized while the original feature space should be transformed as little as possible. Both goals are clearly conflicting as will be discussed in this paper.

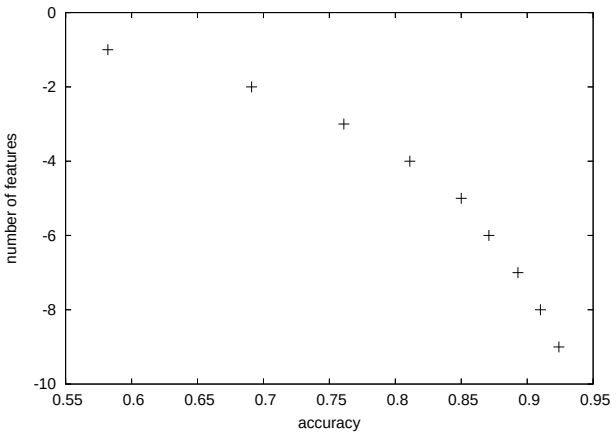


Figure 1: A typical Pareto front for supervised multi-objective feature selection.

1.1 Outline

In section 2 we will discuss existing approaches for multi-objective unsupervised feature selection for different cluster evaluation measures. Although the transfer from supervised learning is an appealing idea, we will show that these approaches do not lead to complete Pareto fronts for this type of problem. In section 3 we will discuss how simply changing the optimization direction for one of the criteria leads to a natural multi-objective optimization problem which will be solved by means of evolutionary algorithms. Finally, section 4 enriches the proposed framework by incorporating feature construction as well. Section 5 presents results on several artificial and real-world data sets and compares the discussed approaches. Section 6 concludes this paper.

2 Multi-objective feature selection for clustering

We will discuss in the next two sections why multi-objective optimization is a natural choice for selecting appropriate feature subsets for clustering problems. A straightforward approach for this type of optimization problem is to simultaneously optimize conflicting criteria by transforming the problem into a single-objective optimization problem. This leads to a set of user parameters which have to be defined in order to weight the criteria. However, in the clustering setting the user has no idea of criteria weights and, furthermore, there exist no simple decision about correct or wrong clusterings. Such a decision would totally depend on the amount of information the user can obtain from different clusterings. We try to maintain as much information as possible and aim at finding all solutions which are optimal for arbitrary criteria weight vectors. These solutions are called *Pareto-optimal*. The multi-objective search space of a maximization problem is subject to a partial order:

Definition 1 *A solution* a dominates *a solution* b *(written as* $a \succ b$*) if for the* p *criteria* r_i *the following is true:*

$$\forall i \in \{1,\ldots,p\} : r_i(a) \geq r_i(b) \quad \wedge \qquad (1)$$
$$\exists i \in \{1,\ldots,p\} : r_i(a) > r_i(b)$$

Our selection scheme needs to decide if a solution is dominated by a set B of solutions. We define:

Definition 2 *A solution* a *is* non-dominated *by a set of solutions* B *if* $\nexists b \in B : b \succ a$.

Now we are able to define what we mean with Pareto-optimal solutions:

Definition 3 *A solution* a *is* Pareto-optimal *if* a *is non-dominated by the complete solution space.*

The usual approach for multi-objective problems are evolutionary algorithms which can optimize more than one target function by introducing special selection operators [Coello Coello, 1999]. Traditional approaches in the field of mathematical programming must be applied more than once for multi-objective optimization [Yu and Zeleny, 1975]. Due to the population based approach of evolutionary algorithms a broad selection of Pareto-optimal solutions can be found during one run. The user can select one of these solutions after optimization. Additionally, multi-objective evolutionary algorithms do not strongly depend on form and continuity of the Pareto-optimal set [Coello Coello, 1999]. We will see in Section 5 that for clustering with non-normalized optimization criteria the Pareto front is neither nicely shaped nor continuous.

A basic condition to pose an multi-objective optimization problem properly is that the described criteria are actually in conflict to each other. By improving on one criterion, we cannot simultaneously improve on the other criteria. Only problems for which this condition holds are sound and can be solved by multi-objective optimization.

The current state of the art for multi-objective unsupervised feature selection is represented by the work initially described in [Kim *et al.*, 2000; 2002] and [Morita *et al.*, 2003]. In the following, we will describe both approaches and show that they are both limited in several ways. These limitations are a result of the way the multi-objective optimization problem is posed.

The corresponding methods all employ a wrapper approach. They subsequently apply a clustering scheme, e.g. k-means, to different feature subsets and evaluate the result with respect to several criteria. In [Kim *et al.*, 2002] four performance criteria for k-means clustering are used[1]. The first one is a variant of within cluster distance W that is normalized by the number of features

$$W_{norm} = \frac{1}{M} W \text{ with } W = \sum_{k=1}^{K} \sum_{x_i \in C_k} \sum_{m=1}^{M} (x_{im} - c_{km})^2 \qquad (2)$$

where c_{km} as the m-th value of the centroid of cluster C_k and x_{im} is the m-th value of the example x_i. The centroid is the point with the smallest distance to all points in C_k. A variant of between cluster distance is used as a second measure. However, this measure behaves essentially in the same way as W_{norm} (minimizing within cluster distance is equivalent to maximizing between cluster distance [Hastie *et al.*, 2001]). The third measure represents the number of clusters K which should be minimized. The last measure captures the number of features nf that should be minimized as well.

In the following theorem we show that for a given number of clusters K minimizing W_{norm} and the number of features leads to exactly one Pareto optimal point. This optimal point always selects one single feature from the dataset, in particular the one that leads to a minimal loss with respect to the used clustering performance criterion.

Theorem 1 *Minimizing* W_{norm} *and the number of features* nf *leads to one single Pareto optimal point.*

[1] In the original work all criteria are normalized by a constant. This, however, has no influence on Pareto optimality.

Proof: For W_{norm} we can denote the loss of an individual feature m as

$$a_m = \sum_{k=1}^{K} \sum_{x_i \in C_k} (x_{im} - c_{km})^2 \quad (3)$$

In order to minimize the number of features selecting only one feature is optimal. We show that always

$$W_{norm} \geq \min_{1 \leq m \leq M} \{a_m\}. \quad (4)$$

That means that the performance can only decrease by adding any feature but the one that optimizes a_m. It can easily be seen that

$$W_{norm} = \frac{1}{M} \sum_{m=1}^{M} a_m \quad (5)$$

$$\geq \frac{1}{M} \sum_{m=1}^{M} \min_{1 \leq m \leq M} \{a_m\} \quad (6)$$

$$= \min_{1 \leq m \leq M} \{a_m\} \quad (7)$$

Hence, using W_{norm} for optimization is not a well suited approach for feature selection in clustering problems as it leads to trivial solutions. Simultaneously minimizing W_{norm} and the number of features nf leads to one single Pareto optimal point. Therefore, selecting a single feature only is always the best solution for both criteria. The Pareto set collapses into a single solution. A similar proof can be given for normalized between cluster distance.

In [Morita *et al.*, 2003] a normalized variant of the DBIndex [Davies and Bouldin, 1979] is proposed as alternative performance criterion to W_{norm}, hence

$$DB_{norm} = \frac{1}{M} DB \quad (8)$$

$$\text{with } DB = \frac{1}{K} \sum_{k=1}^{K} \max_{k,l \neq k} \left\{ \frac{S_k + S_l}{d(c_{km}, c_{lm})} \right\}$$

where d is the Euclidean distance and S_k and S_l are the average within cluster distances for cluster C_k and C_l respectively which is defined as

$$S_k = \frac{1}{|C_k|} \sum_{x_i \in C_k} d(x_i, c_k). \quad (9)$$

This approach is better suited, as the DBIndex is normalized with respect to the feature space. However, this criterion is still very sensitive. If the feature set contains for example a real valued feature that takes discrete values only, then choosing this one feature is again Pareto optimal, similar to the case of W_{norm}. However, this one feature does almost certainly not represent the complete dataset in the descriptive sense mentioned in the introduction. In section 5, we will see several examples for which the Pareto set collapses into a single trivial solution even for normalized DBIndex or, at least, for which the resulting Pareto sets do not cover the complete range of possible feature subsets. The same applies for other recently proposed normalization schemes [Handl and Knowles, 2006] which basically just increase the weighting factor between the number of features and the cluster evaluation measure.

The major problem of these approaches is that they do not pose the problem correctly from the point of view of multi-objective optimization. In the next section we give an alternative problem formulation that solves the described difficulties.

3 Information preserving feature selection

In the last section, we discussed several quality measurements for clustering schemes. In the following, we assume that all criteria should be maximized during feature selection. In contrast to the existing approaches discussed in section 2 we do not *minimize* the number nf of features but *maximize* it. This change of the optimization direction directly follows from Theorem 1. Although maximizing the number of features during feature selection might sound surprising at first, this paradigm change can be motivated by the aim of unsupervised learning: the search for descriptive patterns. Maximizing the number of features prevents the algorithm from selecting trivial solutions and leads to more complete Pareto sets of diversive natural clusterings. The fitness is evaluated by performing a clustering scheme on the reduced feature sets. We use DB as quality criterion. Since there is a natural competition between maximizing the number of features nf and the cluster criterion we do not need to apply an artificial normalization factor as in DB_{norm}.

We use NSGA-II as a multi-objective feature selection wrapper [Deb *et al.*, 2002]. NSGA-II employs a selection technique which first sorts all individuals into levels of non-domination. Individuals from the first levels are added to the next generation until the desired population size is reached. Before individuals are added from the last possible level, this level is sorted with respect to the crowding distance in order to preserve diversity in the population.

Individuals are bit vectors of length M indicating if a feature should be selected or not. The population size is set to $2M$, the maximal number of generations is 1000. A bit flip mutation is performed with probability $1/M$ and uniform crossover with probability 0.9.

4 Information preserving feature aggregation

Merely selecting features is often not sufficient. First, sparse data is a severe problem in many applications areas. For text clustering, generalizing terms by adding superordinate terms can significantly improve the quality of the result [Hotho *et al.*, 2003]. The same holds for association rule mining. Adding generalized features, which combine individual items to classes, enables the algorithm to find patterns, which would not be valid in the original data space [Srikant and Agrawal, 1995]. Second, many datasets contain features produced by similar underlying processes, e.g. time series data. Popular preprocessing approaches as moving average replace neighboring values by a generalized value exploiting the assumption that neighbors are similar.

In the following, we present a general formalism for feature aggregation. This formalism is a straightforward generalization of the feature selection framework presented in the last section and should fulfill several requirements. First, the constructed feature space should be easily interpretable in order to allow for a quick inspection of the results. Second, the optimization problem should be posed in a way that it can be solved efficiently. Third, as for selection, trivial solutions must be avoided. Finally, the aggregation value should deviate as little as possible from both given feature values. This last property again is necessary in order to properly define a multi-objective feature space transformation similar to the mere selection problem discussed in the last section.

Definition 4 *Let X denote the data set and X_r, X_s, and X_t single features. A* feature aggregation function *is a function $f : X_r \times X_s \rightarrow X_t$ that maps two features to a new feature.*

Please note that the newly aggregated feature replaces the arguments. In the following, we state formal conditions that an aggregation function should fulfill to meet the points mentioned above. As point of departure, we use the concept of *t-conorms*, which naturally captures the notion of disjunctive value combinations. T-conorms are a class of theoretically and empirically established generic aggregation functions. They are a natural extension of disjunctions for continuous values. Disjunctions have proven to be essential for many data mining applications, e.g. for generalized association rules [Srikant and Agrawal, 1995].

Definition 5 *A function is a* t-conorm *if it fulfills the following constraints:*

1. *Boundary condition:*
$$f(X, 0) = X \tag{10}$$
2. *Commutativity:*
$$f(X_r, X_s) = f(X_s, X_r) \tag{11}$$
3. *Associativity:*
$$f(f(X_r, X_s), X_t) = f(X_r, f(X_s, X_t)) \tag{12}$$
4. *Monotonicity:*
$$X_r \geq X_s \Rightarrow f(X_r, X_t) \geq f(X_s, X_t) \tag{13}$$

Associativity and commutativity ensure that the feature aggregation is order independent, thus that the order in which features are aggregated does not have an influence on the result. This is of course not only desirable for disjunctive aggregations but also considerably reduces the search space and leads to results that are easier to interprete, as the system produces sets of features instead of trees. The boundary condition ensures that the aggregation follows the notion of a disjunctive merging (in contrast to $f(X, 0) = 0$, which would describe a conjunctive aggregation). Monotonicity preserves the ordinal information in the data.

Although t-conorms already can be used for disjunctive aggregations, they are, however, not sufficient to capture the notion of a minimal deviation. For example, it would still be possible that $f(x, x) \neq x$. Thus, even if both features have the same value, the resulting value could be different. This clearly violates the concept of merging two features and altering them minimally as stated above. We therefore add an additional constraint that excludes such functions:

Condition 1 *A function is an* information preserving t-conorm *if it is a t-conorm and fulfills the following minimal deviation condition:*

$$\begin{aligned} &\forall x, y \in X : \neg\exists f'(x, y) : \\ &|f'(x, y) - x| + |f'(x, y) - y| < \\ &|f(x, y) - x| + |f(x, y) - y| \end{aligned} \tag{14}$$

This condition states that the aggregation function should always yield a merged value that has a minimal deviation from both original values. In the following, we show that from the t-conorm conditions and Condition 1, two important properties can analytically be derived. The first property was discussed before and states that the aggregation result for equal values should again be the value itself. Otherwise, users would not be able to understand the meaning of aggregated features and it would not be possible to guarantee that the aggregated features are in any way similar to the original features. This property is called *idempotence* and directly follows from Condition 1:

Lemma 1 *Each information preserving t-conorm fulfills* idempotence, *i.e. $f(x, x) = x$ (proof trivial).*

Still, there might be a problem for non-equal values if the aggregated value would differ too much from the original values. In order to prevent the aggregation function to generate arbitrary values we set a last condition for aggregation functions:

Condition 2 *A function fulfills* domain preservation *iff $min(x, y) \leq f(x, y) \leq max(x, y)$.*

Thus the merged value must be in the domain spanned by the input values. We can show that Condition 2 can directly be followed from Condition 1:

Theorem 2 *A t-conorm $f(x, y)$ that fulfills Condition 1 also fulfills domain preservation.*

Proof: For $x = y$ the condition is trivially violated. We have to prove four cases and assume that $f(x, y) > max(x, y)$ and $x > y$. Then:

$$\begin{aligned} &|f(x, y) - x| + |f(x, y) - y| = \\ &(f(x, y) - x) + (f(x, y) - y) > \\ &(f(x, y) - x) + (x - y) \geq (x - y) = \\ &|max(x, y) - x| + |max(x, y) - y| \end{aligned} \tag{15}$$

Thus condition 1 is violated. The other cases can be shown analogously.

Together with Lemma 1, this theorem states that Condition 2 is a sufficient condition for information preserving t-conorms (Condition 1). Moreover, the above conditions constrain the set of possible aggregation functions to exactly a single one, the maximum function:

Corollary 1 *The maximum function is the only aggregation function fulfilling the information preserving t-conorm condition.*

Proof: It can be shown that for all t-conorms $f(x, y)$ the following holds: $max(x, y) \leq f(x, y)$ (proof trivial). On the other hand, the domain preservation conditions requires that $f(x, y) \leq max(x, y)$, hence $f(x, y) = max(x, y)$.

Given the aggregation function, we still need to extend the performance measure proposed above, to capture feature aggregation as well. We have seen that for mere feature selection the number nf of selected features is sufficient for measuring the degree of feature space preservation. One of the surprising results of this work is that this number should be maximized instead of minimized in the unsupervised setting. We want to extend the proposed framework in a way that feature selection is a special case of the more generic feature space transformation setting. We give two conditions which must be fulfilled by this generalized cost measure:

Condition 3 *Let nf_o be the number of selected original, i.e. non-aggregated, features. Let nf_a be the number of aggregated features in the transformed feature set. For an unsupervised feature space transformation measure nf the following must hold:*

Abba.	properties	N	M	noise	σ_o	σ_n	K	Results
GRID	equidistant values	3125	5	0	–	–	0	(a) and (b)
RANDOM	uniformly distributed	500	10	10	–	∞	0	(c) and (d)
GM	Gaussian mixture	1000	15	10	0.5	0.5	16	(e) and (f)
GM-L	Gaussian mixture	100000	15	10	0.5	0.5	16	(g) and (h)
IRIS	Iris without noise	150	4	0	0.8	–	3	(i) and (j)
IRIS-NN	Iris with nominal noise	150	5	1	0.8	0.01	3	(k) and (l)
IRIS-GN	Iris with Gaussian noise	150	14	10	0.8	0.8	3	(m) and (n)
WPBC	WPBC without noise	198	34	0	33.2	–	?	(o) and (p)

Table 1: The used data sets for unsupervised feature selection. The first column summarizes the used abbreviations, the second describes the data set. N is the total number of examples, M the number of features. *Noise* defines how many features of M where explicitly added noise features. The next columns define the mean standard deviation of the original (σ_o) and the noise features (σ_n). The column K indicates the number of clusters if known. The last column indicates which Pareto sets were found with both approaches.

abbr.	properties	N	M	K	Results
IRIS-M	Iris data set with divided features	150	8	3	(a)
KDDCUP	quantum physics data (KDD cup 2004)	5000	78	2	(b)
NEWS	articles from three newsgroups	3000	1052	3	(c)

Table 2: The used data sets for unsupervised feature space transformation.

1. *$nf = nf_o$ if the feature set does not contain any aggregated features.*
2. *Every aggregation must lead to a loss of $-a$ with $a > 0$.*

In the following, we will assume $a = 1$. A very simple measure fulfilling these conditions is given by $nf = nf_o + nf_a$. If no features were aggregated nf_a is 0 and $nf = nf_o$. Since all aggregation functions must replace the input features, aggregating two original base features reduces nf_o by 2 and increases nf_a by 1. Hence nf is totally increased by 1. The same applies in the case of two already aggregated features or in the case of a merge of one base feature with an already aggregated features. Hence, every aggregation leads to the same loss of -1. Just as for the mere feature selection case the number nf should be maximized in order to ensure minimal deviation and thus a set of conflicting criteria. This again leads to a proper definition of a multi-objective optimization problem even for the unsupervised feature transfer setting.

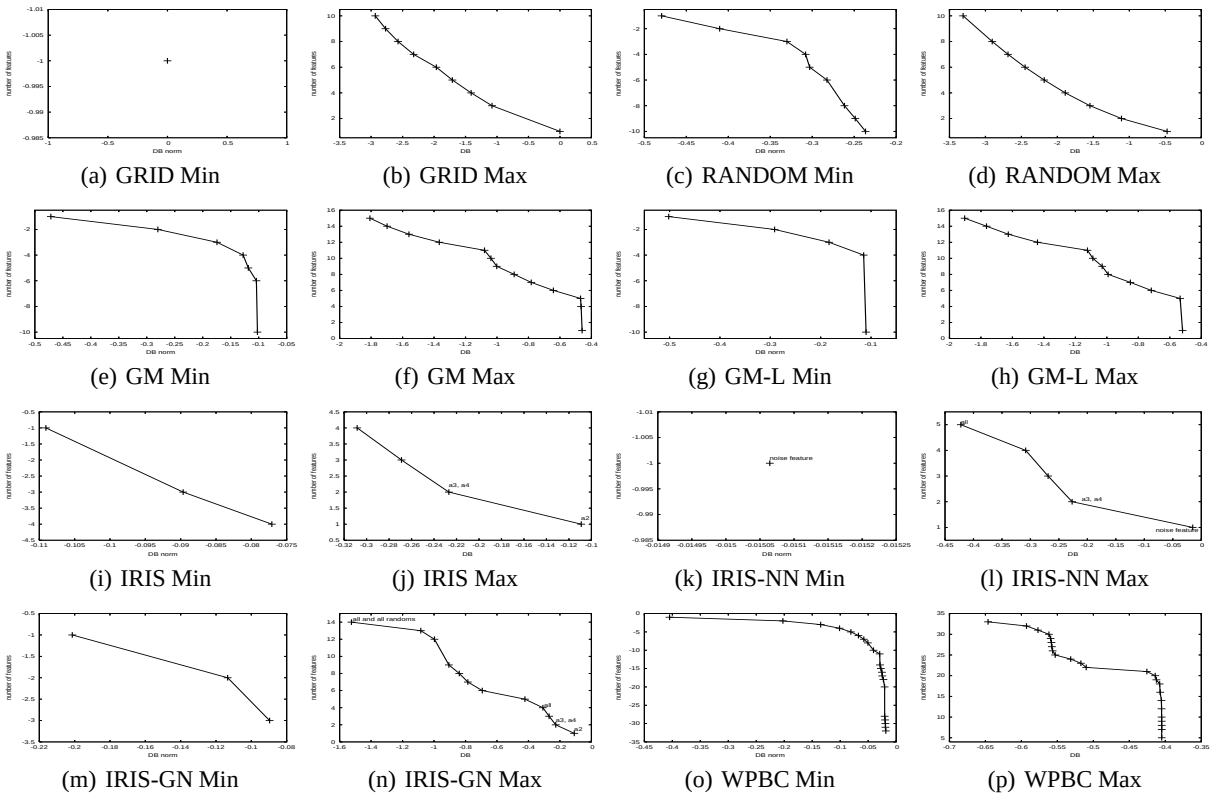
Allowing the aggregation of features induce a representation change for the individuals of the evolutionary algorithm. Individuals are still represented by vectors $\vec{v}$ of length M. In contrast to the feature selection case, each coefficient of this vector is a number $v_i \in [-1, \max(v_1, \ldots, v_M)]$. This number v_i represents the state of the i-th feature. -1 means that the feature is not selected at all. 0 means that the feature is used in its original form. Any number greater than 0 means that the feature should be aggregated with other features with the same number. This ensures that each feature is used at most once in the complete set. The mutation operator performs an uniformly distributed random change of each coefficient in the interval $[-1, max(v_1, \ldots, v_M) + 1]$. This mutation is performed with probability $1/M$ for each coefficient. The other algorithm parameters are the same as in the special case of feature selection.

One important property of our approach is that the number of features is strictly monotonically decreasing. This is important for the efficiency of the proposed method, as decreasing the number of features will decrease the runtime of the inner clustering algorithm. In contrast to other feature construction approaches the used vector representation also ensures that the amount of memory is restricted to the start individual size. Therefore, our approach can also be used for large scale unsupervised feature selection and aggregation and is feasible even for large data sets with many features.

5 Evaluation

We first compare our approach to existing approaches for multi-objective unsupervised feature selection. We then analyze the properties of our generalized feature space transformation on several synthetic and real-world data sets. The essential requirement is that the resulting Pareto sets are as broad as possible. The worst case is a Pareto front that collapses into a single point.

In order to measure the effect of the artificial normalization factor necessary for the existing feature set minimization approach, we applied the algorithms on a grid data set (GRID) and a random data set (RANDOM) containing only white noise. Another data set (GM) consisting of 16 Gaussian clusters with random standard deviations between 0.0 and 1.0 in five dimensions was created. This data set was enriched with ten additional single Gaussian noise features with average standard deviation 0.5. The same data set but with 100000 examples was created in order to check if our approach is feasible with respect to large data set sizes (GM-L). We also applied both algorithms on two clustering benchmark datasets, namely the IRIS data set [Fisher, 1936] and the WPBC (Wisconsin Prognostic Breast Cancer) data set [Wolberg *et al.*, 1995]. These data sets were also used by [Kim *et al.*, 2000; 2002; Morita *et al.*, 2003] for evaluation. The WPBC data set is especially interesting because of many redundant features.



(a) GRID Min (b) GRID Max (c) RANDOM Min (d) RANDOM Max

(e) GM Min (f) GM Max (g) GM-L Min (h) GM-L Max

(i) IRIS Min (j) IRIS Max (k) IRIS-NN Min (l) IRIS-NN Max

(m) IRIS-GN Min (n) IRIS-GN Max (o) WPBC Min (p) WPBC Max

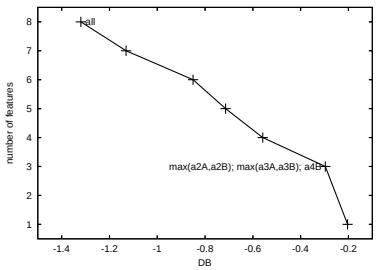
Figure 2: The Pareto fronts for all data sets. The left result for each dataset is achieved by the approach discussed in section 2 for a normalized DBIndex (nDB) against minimizing nf. It can clearly be seen that these results are not as complete and that kinks are covered by the artificial $1/x$ structure. The results on the right are achieved by our maximization approach, thus non-normalized DBIndex (DB) against maximizing nf.

This allows us to check how well both approaches are able to cope with redundancy. Table 1 summarizes the properties of all data sets.

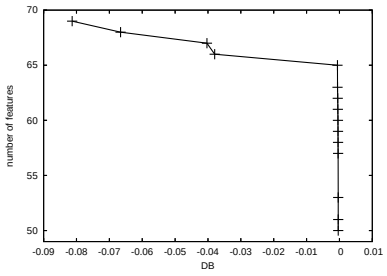
All experiments were performed with the free machine learning environment YALE [Mierswa *et al.*, 2006]. It should be noted that in most cases the population converges to the final front after less than 20 generations. The NSGA-II selection was able to sustain the found solution until the end of optimization. Figure 2 shows all Pareto sets for the simultaneous optimization of the used cluster criterion and the feature set size. The achieved performance (DB or DB_{norm}) is depicted on the x-axis, the number of features nf is depicted on the y-axis. In case of the minimization approach, the number of features is multiplied with -1 for optimization. In order to turn the problem into a full maximization problem, we also multiplied DBIndex with -1.

One might ask why the comparison plots have different scales and variables. Former experiments have shown that both approaches are able to deliver Pareto-optimal solutions independently of the used scale. However, a scale based comparison alone is not applicable in order to decide which Pareto sets are superior. Hence, other criteria like completeness or shape of the fronts must be taken into account. One of the insights of our work is that a normalization factor, as proposed by other authors, is not necessary if the number of (original) features is maximized. Since the normalization induces an additional artificial competition *and* covers inherent structures in the Pareto sets, we decided to plot the results for non-normalized DB. For normalized DB_{norm}, which would lead to the same scales for both approaches, our approach simply produces a super set of solutions. Moreover, if non-normalized DB is used for the formerly proposed minimization approach, the Pareto fronts collapse in almost all cases.

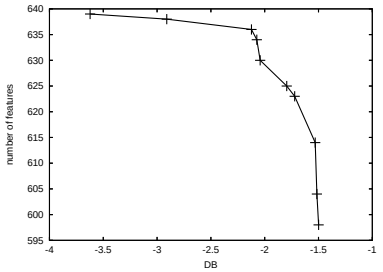
It can clearly be seen that in all cases the Pareto sets provided by our approach contain more points than the results of the normalized minimization approach. If there is only one feature with a relative small standard deviation, the Pareto set of the minimization approach will still collapse (GRID and IRIS-NN) even for normalized DBIndex. Of course, well-defined multi-objective solutions should be able to deliver the complete Pareto front including more than only this one trivial solution. Moreover, the normalization factor $1/x$ introduces a convex front although there is nothing to optimize at all. This effect can be seen for the RANDOM data set, where the minimization approach finds a convex Pareto front while the front provided by our approach is still linear. For the Gaussian mixture clusters (GM), again our approach is able to deliver a broader Pareto front including some kinks. These kinks can be used as hint for interesting regions of the Pareto front easing the final selection of solutions. The best clustering result for the minimization approach was the feature set at the right end of the Pareto front containing 10 features. The found clusterings for this feature set, however, did not correspond to the original clusterings at all. On the other hand, our approach was able to find the correct feature set consisting of 5 features providing 12 of the 16 original clusters (the first kink seen from the right end). It can also be seen that the main structure of the Pareto front remains with respect to the sample size. Applying our approach on GM-L with 100000 examples was feasible and delivers similar results.



(a) IRIS-M



(b) KDDCUP



(c) NEWS

Figure 3: The Pareto fronts delivered by the unsupervised multi-objective feature aggregation experiments. The Pareto sets still cover the complete range of possible solutions, e.g. from 1 until 8 features for the IRIS-M data set. Additionally, features were only aggregated if this combination was necessary.

For both the normal IRIS data set and the IRIS-GN data set enriched with noise features the proposed approach finds the complete Pareto set including the correct clusterings while the minimization approach was only able to find a small number of feature subsets. Correct clustering are depicted by small labels indicating the used feature set. For the IRIS data set without noise, the minimization approach was also able to deliver a feature subset of 3 features providing the correct clustering. In contrast to our approach, the minimization approach collapses (IRIS-NN) or was not able to deliver the correct clustering (IRIS-GN). Since it is not clear which clustering is "correct" beforehand, the user should be able to select from the complete Pareto front delivered by our approach. The same conclusion applies for the other real-world data set WPBC.

In addition to the comparison between our approach and the formerly proposed approach, we also applied the new unsupervised feature aggregation algorithm on one semi-synthetic and two real-world data sets. The properties of these data sets are summarized in Table 2. For the data set IRIS-M we divided the values of the four Iris features into two parts A and B resulting in a total of eight features. For each of the original feature values we randomly select one of the new features as target, the other feature value

Data set	$\|P\|$		$\|C\|$		F	
	min	max	min	max	min	max
GRID	1	**9**	0	0	?	?
RANDOM	9	9	0	0	–	–
GM	7	**13**	0	**12**	no	**yes**
GM-L	5	**12**	0	**11**	no	**yes**
IRIS	3	**4**	3	3	yes	yes
IRIS-NN	1	**5**	0	**3**	no	**yes**
IRIS-GN	3	**12**	0	**3**	no	**yes**
WPBC	21	**24**	?	?	?	?
IRIS-M	–	**7**	–	**3**	–	**yes**
KDDCUP	–	**15**	–	**2**	–	–
NEWS	–	**10**	–	**3**	–	–

Table 3: Comparison of the results for the minimization approach and the proposed non-normalized maximization approach. Better values are indicated with a bold font.

is set to a random value between 0 and the current value. This way the complete original information can only be reconstructed by aggregating the correct features. The KDDCUP data set consists of a stratified sample of 5000 examples drawn from the quantum physics data of the KDD cup 2004. For the data set NEWS, we combined three newsgroups of the well known 20-newsgroups data set which results in 3000 examples.

Figure 3 shows the results for unsupervised feature space transformation. For the IRIS-M data set, the complete range of solutions is covered by the resulting Pareto set and the necessary features $a2$ and $a3$ were reconstructed by aggregation. At this point, the known clustering of the IRIS data set was found by our approach (again depicted by a label indicating the used feature set). For KDDCUP, a clear kink can be seen indicating redundant features which are aggregated in the lower part of the vertical line. For both the KDDCUP data set and the NEWS data set two respectively three clusters were found covering large parts of the original classes. For all real-world data sets a broad range of the feature space is covered by the result which again supports our claim of robust and useful solutions.

Table 3 summarizes all results for both the mere feature selection case and the feature aggregation case. $|P|$ denotes the number of found Pareto points, $|C|$ indicates the number of found correct clusterings, and column F indicates if the correct feature set was found (if known). Better values are marked with a bold font. The hyphen indicates, that the approach can not be applied to this data set, the question mark indicates that the correct values are not known. It can clearly be seen that the new approach outperforms existing approaches in terms of Pareto front completeness, robustness, and ability to find correct clusterings.

6 Conclusion

We presented a novel multi-objective framework for feature space transformation in clustering settings which plays an important role in a wide variety of applications ranging from pattern recognition to customer relationship management and web search. Clustering is an inherently multi-objective problem. There is usually not one correct result as for supervised learning. Users rather explore the space

of results interactively to gain insight into the natural patterns within the data set.

Our work is based on previous work on multi-objective feature selection for clustering. We found, however, that existing approaches were limited in two points. First, they do not pose the optimization problem in a sound and robust way. We showed both analytically and empirically, that the corresponding sets of Pareto optimal solutions collapse to a single, trivial solution. We therefore proposed an approach that is based on the idea of information preservation. As much of the original data space should be preserved as possible, while the validity of the resulting clusters is optimized. We show that this approach yields complete and useful Pareto sets. The original feature set and clustering were found in all cases. These Pareto sets moreover show a strong inner structure which can be used to explore the set of solutions even more efficiently by inspecting only these interesting points.

In addition, merely selecting features is not sufficient in many settings. Especially the problems of sparse data and feature interactions cannot be properly solved by feature selection only. We extended our approach to allow for a limited form of feature construction as well. Aggregation is used to derive new features, that generalize over two or more features in the original data set. T-conorms are a class of theoretically and empirically established generic aggregation functions. They are a natural extension of disjunctions for continuous values, which have proven to be essential for many data mining applications, e.g. generalized association rules. A set of basic conditions limits feature aggregation to the t-conorm maximum which summarizes two features with minimal alteration. We show that even for feature aggregation our approach leads to robust Pareto sets. Our experiments supports this claim.

Also, our approach is very generic. As it essentially adopts a wrapper approach, it can be combined with a large variety of problems and algorithms. It is therefore easy to adapt to new problems and application domains. We successfully applied the proposed approach also for graph-based clusterings or density based clusterings or for other performance criteria [Handl and Knowles, 2006] without need for additional normalization.

References

[Coello Coello, 1999] C. A. Coello Coello. A comprehensive survey of evolutionary-based multiobjective optimization techniques. *Knowledge and Information Systems*, 1(3):129–156, 1999.

[Davies and Bouldin, 1979] D.L. Davies and D.W. Bouldin. A cluster separation measure. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1(2):224–227, 1979.

[Deb *et al.*, 2002] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan. A fast and elitist multi-objective genetic algorithm: Nsga-ii. Technical report, Kanpur Genetic Algorithms Laboratory, Indian Institute of Technology, 2002.

[Emmanouilidis *et al.*, 2000] C. Emmanouilidis, A. Hunter, and J. MacIntyre. A multiobjective evolutionary setting for feature selection and a commonality-based crossover operator. In *Proc. of the Congress on Evolutionary Computation (CEC)*, pages 309–316, 2000.

[Fisher, 1936] R.A. Fisher. The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, 7(2):179–188, 1936.

[Handl and Knowles, 2006] J. Handl and J. Knowles. Feature subset selection in unsupervised learning via multiobjective optimization. *International Journal on Computational Intelligence Research*, 2006.

[Hastie *et al.*, 2001] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer series in statistics. Springer, 2001.

[Hotho *et al.*, 2003] A. Hotho, S. Staab, and G. Stumme. Ontologies improve text document clustering. In *Proc. of the IEEE International Conference on Data Mining (ICDM 2003)*, 2003.

[Kim *et al.*, 2000] Y. Kim, W.N. Street, and F. Menczer. Feature selection in unsupervised learning via evolutionary search. In *Proc. of the 6th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 365–369, New York, NY, USA, 2000. ACM Press.

[Kim *et al.*, 2002] Y. Kim, W. N. Street, and F. Menczer. Evolutionary model selection in unsupervised learning. *Intelligent Data Analysis*, 6:531–556, 2002.

[Mierswa *et al.*, 2006] I. Mierswa, M. Wurst, R. Klinkenberg, M. Scholz, and T. Euler. YALE: Rapid prototyping for complex data mining tasks. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2006)*, 2006.

[Morita *et al.*, 2003] M. Morita, R. Sabourin, F. Bortolozzi, and C.Y. Suen. Unsupervised feature selection using multi-objective genetic algorithms for handwritten word recognition. In *Proc. of the 7th International Conference on Document Analysis and Recognition (ICDAR)*, 2003.

[Roth and Lange, 2003] V. Roth and T. Lange. Feature selection in clustering problems. In *Proc. of Neural Information Processing Systems (NIPS)*, 2003.

[Schölkopf and Smola, 2002] B. Schölkopf and A. J. Smola. *Learning with Kernels – Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press, 2002.

[Srikant and Agrawal, 1995] Ramakrishnan Srikant and Rakesh Agrawal. Mining generalized association rules. In Umeshwar Dayal, Peter M. D. Gray, and Shojiro Nishio, editors, *VLDB'95, Proceedings of 21th International Conference on Very Large Data Bases, September 11-15, 1995, Zurich, Switzerland*, pages 407–419. Morgan Kaufmann, 1995.

[Wolberg *et al.*, 1995] W.H. Wolberg, W.N. Street, D.M. Heisey, and O.L. Mangasarian. Computer-derived nuclear "grade" and breast cancer prognosis. *Analytical and Quantitative Cytology and Histology*, 17:257–264, 1995.

[Yu and Zeleny, 1975] P. L. Yu and M. Zeleny. The set of all nondominated solutions in linear cases and a multicriteria Simplex method. *Journal of Mathematical Analysis and Applications*, 49:430–468, 1975.

Crime Pattern Detection Using Data Mining

Shyam Varan Nath
Florida Atlantic University/ Oracle Corporation
SNath1@FAU.edu / Shyam.Nath@Oracle.com
+1(954) 609 2402

Abstract

Can crimes be modeled as data mining problems? We will try to answer this question in this paper. Crimes are a social nuisance and cost our society dearly in several ways. Any research that can help in solving crimes faster will pay for itself. Here we look at use of clustering algorithm for a data mining approach to help detect the crimes patterns and speed up the process of solving crime. We will look at k-means clustering with some enhancements to aid in the process of identification of crime patterns. We will apply these techniques to real crime data from a sheriff's office and validate our results. We also use semi-supervised learning technique here for knowledge discovery from the crime records and to help increase the predictive accuracy. We also developed a weighting scheme for attributes here to deal with limitations of various out of the box clustering tools and techniques. This easy to implement machine learning framework works with the geo-spatial plot of crime and helps to improve the productivity of the detectives and other law enforcement officers. It can also be applied for counter terrorism for homeland security.

Keywords: *Crime-patterns, clustering, data mining, k-means, law-enforcement, semi-supervised learning*

1. Introduction

Historically solving crimes has been the prerogative of the criminal justice and law enforcement specialists. With the increasing use of the computerized systems to track crimes, computer data analysts have started helping the law enforcement officers and detectives to speed up the process of solving crimes. Here we will take an inter-disciplinary approach between computer science and criminal justice to develop a data mining paradigm that can help solve crimes faster. More specifically, we will use clustering based models to help in identification of crime patterns[1].

We will discuss some terminology that is used in criminal justice and police departments and compare and contrast them relative to data mining systems. Suspect refers to the person that is believed to have committed the crime. The suspect may be identified or unidentified. The suspect is not a convict until proved guilty. The victim is the person who is the target of the crime. Most of the time the victim is identifiable and in most cases is the person reporting the crime. Additionally, the crime may have some witnesses. There are other words commonly used such as homicides that refer to manslaughter or killing someone. Within homicides there may be categories like infanticide, eldercide, killing intimates and killing law enforcement officers. For the purposes of our modeling, we will not need to get into the depths of criminal justice but will confine ourselves to the main kinds of crimes.

Cluster (of crime) has a special meaning and refers to a geographical group of crime, i.e. a lot of crimes in a given geographical region. Such clusters can be visually represented using a geo-spatial plot of the crime overlayed on the map of the police jurisdiction. The densely populated group of crime is used to visually locate the 'hot-spots' of crime. However, when we talk of clustering from a data-mining standpoint, we refer to similar kinds of crime in the given geography of interest. Such clusters are useful in identifying a crime pattern or a crime spree. Some well-known examples of crime patterns are the DC sniper, a serial-rapist or a serial killer. These crimes may involve single suspect or may be committed by a group of suspects. The below figure shows the plot of geo-spatial clusters of crime.

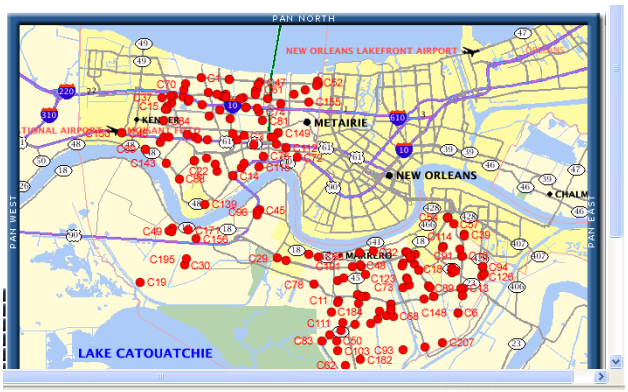


Fig 1 Geo-spatial plot of crimes, each red dot represents a crime incident.

2. Crime Reporting Systems

The data for crime often presents an interesting dilemma. While some data is kept confidential, some becomes public information. Data about the prisoners can often be viewed in the county or sheriff's sites. However, data about crimes related to narcotics or juvenile cases is usually more restricted. Similarly, the information about the sex offenders is made public to warn others in the area, but the identity of the victim is often prevented. Thus as a data miner, the analyst has to deal with all these public versus private data issues so that data mining modeling process does not infringe on these legal boundaries.

Most sheriffs' office and police departments use electronic systems for crime reporting that have replaced the traditional paper-based crime reports. These crime reports have the following kinds of information categories namely - type of crime, date/time, location etc. Then there is information about the suspect (identified or unidentified), victim and the witness. Additionally, there is the narrative or description of the crime and Modus Operandi (MO) that is usually in the text form. The police officers or detectives use free text to record most of their observations that cannot be included in checkbox kind of pre-determined questions. While the first two categories of information are usually stored in the computer databases as numeric, character or date fields of table, the last one is often stored as free text.

The challenge in data mining crime data often comes from the free text field. While free text fields can give the newspaper columnist, a great story line, converting them into data mining attributes is not always an easy job. We will look at how to arrive at the significant attributes for the data mining models.

3. Data Mining and Crime Patterns

We will look at how to convert crime information into a data-mining problem [2], such that it can help the detectives in solving crimes faster. We have seen that in crime terminology a cluster is a group of crimes in a geographical region or a hot spot of crime. Whereas, in data mining terminology a cluster is group of similar data points – a possible crime pattern. Thus appropriate clusters or a subset of the cluster will have a one-to-one correspondence to crime patterns.

Thus clustering algorithms in data mining are equivalent to the task of identifying groups of records that are similar between themselves but different from the rest of the data. In our case some of these clusters will useful for identifying a crime spree committed by one or same group of suspects. Given this information, the next challenge is to find the variables providing the best clustering. These clusters will then be presented to the detectives to drill down using their domain expertise. The automated detection of crime patterns, allows the detectives to focus on crime sprees first and solving one of these crimes results in solving the whole "spree" or in some cases if the groups of incidents are suspected to be one spree, the complete evidence can be built from the different bits of information from each of the crime incidents. For instance, one crime site reveals that suspect has black hair, the next incident/witness reveals that suspect is middle aged and third one reveals there is tattoo on left arm, all together it will give a much more complete picture than any one of those alone. Without a suspected crime pattern, the detective is less likely to build the complete picture from bits of information from different crime incidents. Today most of it is manually done with the help of multiple spreadsheet reports that the detectives usually get from the computer data analysts and their own crime logs.

We choose to use clustering technique over any supervised technique such as classification, since crimes vary in nature widely and crime database often contains several unsolved crimes. Therefore, classification technique that will rely on the existing and known solved crimes, will not give good predictive quality for future crimes. Also nature of crimes change over time, such as Internet based cyber crimes or crimes using cell-phones were uncommon not too long ago. Thus, in order to be able to detect newer and unknown patterns in future, clustering techniques work better.

4. Clustering Techniques Used

We will look at some of our contributions to this area of study. We will show a simple clustering example here. Let us take an oversimplified case of crime record. A crime data analyst or detective will use a report based on this data sorted in different orders, usually the first sort will be on the most important characteristic based on the detective's experience.

Crime Type	Suspect Race	Suspect Sex	Suspect Age gr	Victim age gr	Weapon
Robbery	B	M	Middle	Elderly	Knife
Robbery	W	M	Young	Middle	Bat
Robbery	B	M	?	Elderly	Knife
Robbery	B	F	Middle	Young	Piston

Table 1 Simple Crime Example

We look at table 1 with a simple example of crime list. The type of crime is robbery and it will be the most important attribute. The rows 1 and 3 show a simple crime pattern where the suspect description matches and victim profile is also similar. The aim here is that we can use data mining to detect much more complex patterns since in real life there are many attributes or factors for crime and often there is partial information available about the crime. In a general case it will not be easy for a computer data analyst or detective to identify these patterns by simple querying. Thus clustering technique using data mining comes in handy to deal with enormous amounts of data and dealing with noisy or missing data about the crime incidents.

We used k-means clustering technique here, as it is one of the most widely used data mining clustering technique. Next, the most important part was to prepare the data for this analysis. The real crime data was obtained from a Sherriff's office, under non-disclosure agreements from the crime reporting system. The operational data was converted into denormalised data using the extraction and transformation. Then, some checks were run to look at the quality of data such as missing data, outliers and multiple abbreviations for same word such as blank, unknown, or unk all meant the same for missing age of the person. If these are not coded as one value, clustering will create these as multiple groups for same logical value. The next task was to identify the significant attributes for the clustering. This process involved talking to domain experts such as the crime detectives, the crime data analysts and iteratively running the attribute importance algorithm to arrive at the set of attributes for the clustering the given crime types. We refer to this as the semi-supervised or expert-based paradigm of problem solving. Based on the nature of crime the different attributes become important such as the age group of victim is important for homicide, for burglary the same may not be as important since the burglar may not care about the age of the owner of the house.

To take care of the different attributes for different crimes types, we introduced the concept of weighing the attributes. This allows placing different weights on different attributes dynamically based on the crime types being clustered. This also allows us to weigh the categorical attributes unlike just the numerical attributes that can be easily scaled for weighting them. Using the integral weights, the categorical attributes can be replicated as redundant columns to increase the effective weight of that variable or feature. We have not seen the use of weights for clustering elsewhere in the literature review, as upon normalization all attributes assume equal importance in clustering algorithm. However, we have introduced this weighting technique here in light of our semi-supervised or expert based methodology. Based on our weighted clustering attributes, we cluster the dataset for crime patterns and then present the results to the detective or the domain expert along with the statistics of the important attributes.

The detective looks at the clusters, smallest clusters first and then gives the expert recommendations. This iterative process helps to determine the significant attributes and the weights for different crime types. Based on this information from the domain expert, namely the detective, future crime patterns can be detected. First the future or unsolved crimes can be clustered based on the significant attributes and the result is given to detectives for inspection. Since, this clustering exercise, groups hundreds of crimes into some small groups or related crimes, it makes the job of the detective much easier to locate the crime patterns.

The other approach is to use a small set of new crime data and score it against the existing clusters using tracers or known crime incidents injected into the new data set and then compare the new clusters relative to the tracers. This process of using tracers is analogous to use of radioactive tracers to locate something that is otherwise hard to find.

5. Results of Crime Pattern Analysis

The proposed system is used along with the geo spatial plot. The crime analyst may choose a time range and one or more types of crime from certain geography and display the result graphically. From this set, the user may select either the entire set or a region of interest. The resulting set of data becomes the input source for the data mining processing. These records are clustered based on the predetermined attributes and the weights. The resulting, clusters have the possible crime patterns. These resulting clusters are plotted on the geo-spatial plot.

We show the results in the figure below. The different clusters or the crime patterns are color-coded. For each group, the legend provides the total number of crimes incidents included in the group along with the significant attributes that characterize the group. This information is useful for the detective to look at when inspecting the predicted crime clusters.

We validated our results for the detected crime patterns by looking the court dispositions on these crime incidents as to whether the charges on the suspects were accepted or rejected. So to recap the starting point is the crime incident data (some of these crimes already had the court dispositions/ rulings available in the system), which the measured in terms of the significant attributes or features or crime variables such as the demographics of

the crime, the suspect, the victim etc. No information related to the court ruling was used in the clustering process. Next we cluster the crimes based on our weighing technique, to come up with crime groups (clusters in data mining terminology), which contain the possible crime patterns of crime sprees. The geo-spatial plot of these crime patterns along with the significant attributes to quantify these groups is presented to the detectives who now have a much easier task to identify the crime sprees than from the list of hundreds of crime incidents in unrelated orders or some predetermined sort order. In our case, we looked at the crime patterns, as shown in same colors below and looked at the court dispositions to verify that some of the data mining clusters or patterns were indeed crime spree by the same culprit(s).

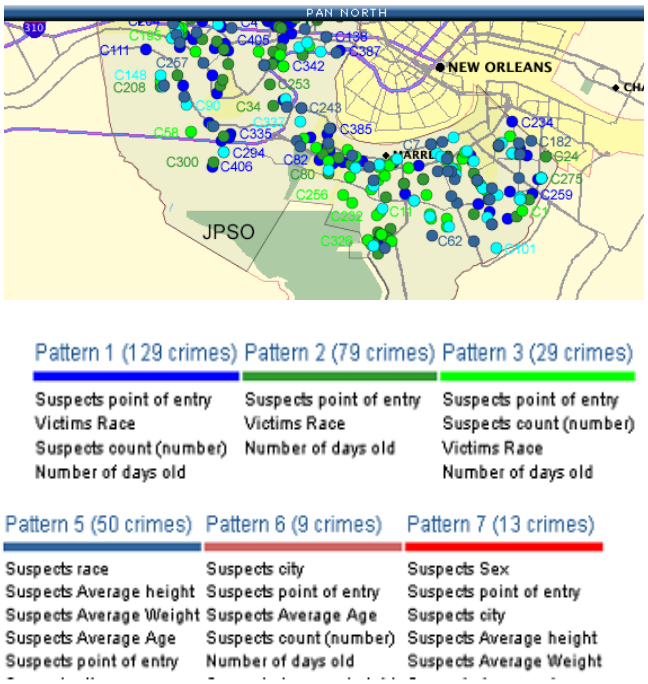


Figure 2 Plot of crime clusters with legend for significant attributes for that crime pattern

6. Conclusions and Future Direction

We looked at the use of data mining for identifying crime patterns crime pattern using the clustering techniques. Our contribution here was to formulate crime pattern detection as machine learning task and to thereby use data mining to support police detectives in solving crimes. We identified the significant attributes; using expert based semi-supervised learning method and developed the scheme for weighting the significant attributes. Our modeling technique was able to identify the crime patterns from a large number of crimes making the job for crime detectives easier.

Some of the limitations of our study includes that crime pattern analysis can only help the detective, not replace them. Also data mining is sensitive to quality of input data that may be inaccurate, have missing information, be data entry error prone etc. Also mapping real data to data mining attributes is not always an easy task and often requires skilled data miner and crime data analyst with good domain knowledge. They need to work closely with a detective in the initial phases.

As a future extension of this study we will create models for predicting the crime hot-spots [3] that will help in the deployment of police at most likely places of crime for any given window of time, to allow most effective utilization of police resources. We also plan to look into developing social link networks to link criminals, suspects, gangs and study their interrelationships. Additionally the ability to search suspect description in regional, FBI databases [4], to traffic violation databases from different states etc. to aid the crime pattern detection will also add value to this crime detection paradigm.

7. References

[1] Hsinchun Chen, Wingyan Chung, Yi Qin, Michael Chau, Jennifer Jie Xu, Gang Wang, Rong Zheng, Homa Atabakhsh, “Crime Data Mining: An Overview and Case Studies”, AI Lab, University of Arizona, proceedings National Conference on Digital Government Research, 2003, available at: http://ai.bpa.arizona.edu/

[2] Hsinchun Chen, Wingyan Chung, Yi Qin, Michael Chau, Jennifer Jie Xu, Gang Wang, Rong Zheng, Homa Atabakhsh, “Crime Data Mining: A General Framework and Some Examples”, IEEE Computer Society April 2004.

[3] C McCue, “Using Data Mining to Predict and Prevent Violent Crimes”, available at: http://www.spss.com/dirvideo/richmond.htm?source=dmpage&zone=rtsidebar

[4] Whitepaper, “Oracle’s Integration Hub For Justice And Public Safety”, Oracle Corp. 2004, available at: http://www.oracle.com/industries/government/IntegrationHub_Justice.pdf

Web Usage Mining for Adaptive and Personalized Websites

Asem Omari and Stefan Conrad
Heinrich-Heine-University Duesseldorf
Institute of Computer Science
Databases and Information Systems
Duesseldorf, Germany
{omari, conrad}@cs.uni-duesseldorf.de

Abstract

The World Wide Web is an important medium for communication, data transaction and retrieving. Data mining is the process of extracting interesting patterns from a set of data sources. Web mining is the application of data mining techniques to extract useful patterns from web data. Web Mining can be divided into three categories, web usage mining, web content mining, and web structure mining. Web usage mining or web log mining is the extraction of interesting patterns from web log server entries. Those patterns are used to study user behavior and interests, facilitate support and services introduced to the website user, improve the structure of the website, and facilitate personalization and adaptive websites. This paper aims to explore various research issues in web usage mining and its application in the field of adaptive, and personalized websites.

1 Introduction

The World Wide Web is an important medium for communication, data transaction and retrieving. Web mining is the use of data mining techniques to extract useful patterns from the web. That extracted patterns are used to improve the structure of websites, improve the availability of the information in the websites and the way that information are introduced to the website user, and to improve data retrieval and the quality of automatic search of information resources available in the web. Web mining can be divided into three major categories:

1.1 Web Usage Mining

Web usage mining describes the usage of web pages. It mines web log records to discover user access patterns of web pages. Usage data are collected from different sources such as web server side data, client side data, and proxy servers. Server side data are collected from the web server of a site that consists of various types of the logs generated by the log server [Pierrakos *et al.*, 2003]. Client side data are collected from the host that is accessing the website by using a remote agents implemented in Java or Javascript [Pierrakos *et al.*, 2003]. That agents are used to collect information directly from the client such as the time the user is accessing or leaving the website, and the user's navigation history. Proxy servers also use access log to record web page requests and responses from the server [Pierrakos *et al.*, 2003].

1.2 Web Content Mining

Web content mining is mining the data that a web page contains. The contents of most of the web pages are texts, graphics, tables, data blocks, and data records. A lot of research has been done to cover different web content mining issues for the purpose of improving the contents of the web pages, improving the way they are introduced to the website user, improving the quality of search results, and extracting interesting web page contents.
The authors in [Lin and Ho, 2002] propose the *InfoDiscoverer* system to discover informative contents from a set of web pages of a website according to HTML tag TABLE in a web page. The system partitions the web page blocks into either informative, or redundant. Informative content blocks are distinguished parts of the page, whereas redundant content blocks are common parts. This approach yields to the increase of the retrieval and extraction precision, and reduces the indexing size and extraction complexity.
A number of methods to help user find various types of unexpected information from his/her competitors' websites are proposed in [Liu *et al.*, 2001]. The work in [Morinaga *et al.*, 2002] presents a new framework for mining product reputations on the internet. It automatically collects people's opinions about target products from web pages, and it uses four different types of text mining techniques to obtain the reputation of those products. The research in [Davison, 2002] examines the accuracy of predicting a user's next action based on the analysis of the content of the pages requested recently by the user. Predictions are made using the similarity of a model of the user's interest to the text in and around the hypertext anchors of recently requested web pages. The authors in [Liu *et al.*, 2003] propose an algorithm called *MDR* (Mining Data Records in web pages) to mine contiguous and non-contiguous data records. It finds all records formed by table and form related tags, i.e., table, form, td, tr, etc. Such data records are important because they often present the essential information of their host pages.

1.3 Web Structure Mining

Links pointing to a document indicate the popularity of the document, whereas links coming out of a document indicate the richness or the variety of topics covered in the document. Web structure mining describes the organization of the content of the web where *structure* is defined by *hyperlinks between pages and HTML formatting commands within a page* [Cohen, 2003].
Understanding the relationship between contents and the structure of the website is useful to keep an overview about

websites. The authors in [Gedov *et al.*, 2004] describe an approach that allows the comparison of web page contents with the information implicitly defined by the structure of the website. In this way, it can be indicated whether a page fits in the content of its link structure, and identify topics which span over several connected web pages. Thus supporting web designers by comparing their intentions with the actual structure and content of the web page.
Other studies deal with the web page as a collection of blocks or segments. The authors in [Cai *et al.*, 2004] use an algorithm to partition the web page into blocks, by extracting the page-to-block, block-to-page relationship from link structure and page layout analysis, a semantic graph can be constructed over the World Wide Web such that each node exactly represents a single semantic topic, this graph can better describe the semantic structure of the web. The authors in [Cohen, 2003] present a survey of some of the ways in which structure within a web page can be used to help machines understand pages.
In this paper, we introduce an overview of various research issues in web usage mining and its application in the field of adaptive and personalized websites. We give an overview about web usage mining in section 2. In section 3, we discuss several data mining techniques used in web usage mining. Before those data mining techniques are applied to web log server data, several preprocessing steps should be done in order to make web log file data ready to be mined, we introduce those preprocessing steps in section 4. Then, in section 5, we discuss adaptive websites. In order to make websites more effective to website users, they should reflect their interests, knowledge, needs, and goals. This can be done through personalization which is the subject of section 6. In this section, we talk about the use of web usage mining techniques for web personalization. In section 7, we list well-known web usage mining and analysis tools. We conclude this paper, and introduce some research directions in section 8.

2 What is Web Usage Mining

Web usage mining or web log mining is the process of applying data mining techniques to web log data in order to extract useful information from user access patterns. Web usage mining tries to make sense of the data generated by the web user's sessions or behaviors [Kosala and Blockeel, 2000]. The web usage data includes data from web server access log, proxy server logs, browser logs, user profiles, registration data, cookies, and user queries [Kosala and Blockeel, 2000]. Web usage mining tries to predict user behavior while user interacts with the web and learns user navigation patterns. The learned knowledge could then be used for different applications such as website personalization, business intelligence, usage characterization and adaptive websites. The authors [Cooley, 2003] show that web usage mining is not only enhanced by web content and structure but it can't be completed without them. There are two common used approaches for web usage mining process [Borges and Levene, 1999]:

- Mapping the log data into relational tables before an adopted data mining techniques is performed.
- Using the log data directly by utilizing special preprocessing techniques.

Web usage mining process consists of three phases: data preprocessing, pattern discovery, and pattern analysis. Pattern discovery is that set of methods, algorithms, and techniques used to extract patterns from web log file. Several techniques are used for pattern discovery such as statistical analysis, clustering, classification, and sequential pattern mining (see section 3). After patterns are discovered they need to be analyzed in order to determine interesting and important patterns, besides the removal of redundant patterns. Pattern analysis has several different forms such as knowledge query mechanism, visualization techniques, and loading usage data into a data cube in order to perform Online Analytical Processing OLAP operations [Srivastava *et al.*, 2000].

3 Web Usage Mining Techniques

In this section, we discuss data mining techniques that are mostly used in web usage mining such as statistical analysis techniques, clustering, classification, association rule mining, and sequential pattern mining.

3.1 Statistical Analysis

Statistical analysis is the process of applying statistical techniques on web log file to describe sessions, and user navigation such as viewing the time and length of a navigational path [Srivastava *et al.*, 2000]. Statistical prediction can also be used to predict when some page or document would be accessed from now [Dhyani *et al.*, 2002]. The work in [Borges and Levene, 1999] makes use of the *N-grammer* model which assumes that when a user is browsing a given page, the last *N* pages browsed affect the probability of the next page to be visited.

3.2 Clustering

Clustering is the process of partitioning a given population of events or items into sets of similar elements [Han and Kamber, 2001]. In web usage mining there are two main interesting clusters to be discovered: usage clusters, and pages clusters [Srivastava *et al.*, 2000]. The authors in [Su *et al.*, 2002] present an approach to cluster web pages to have a high quality clusters of web pages and use that clusters to produce index pages, where index pages are web pages that have direct links to pages that may be of interest of some group of website navigators. In [Koutri and Daskalaki, 2003] clustering techniques are applied to web log file to discover those subsets of web pages that need to be connected, and to improve the already connected pages. [Olga. *et al.*, August 1999] uses the *Competitive Agglomeration Clustering Algorithm* to cluster the sessions extracted from web log server into typical session profiles of users. The authors in [Velásquez *et al.*, 2003] use a clustering algorithm which identifies groups of similar sessions, allowing the analysis of visitor behavior.

3.3 Classification

Classification is dividing an existing set of events or transactions into another predefined sets or classes based on some characteristics. In web usage mining, classification is used to group users into a predefined groups with respect to their navigation patterns in order to develop profiles of users belonging to a particular class or category [Srivastava *et al.*, 2000]. [Ester *et al.*, 2002] introduces several approaches for web page classification. The authors in [Fu *et al.*, 2001] propose an approach to reorganize a website based on user access patterns and the classification of web pages into two categories: index pages, and content pages.

3.4 Association Rule Mining

Association rule mining is the discovery of attribute values that occur frequently together in a given set of data [Han and Kamber, 2001]. Association rules mining techniques are used in web usage mining to find pages that are often viewed together, or to show which pages tend to be visited within the same user session [Baron and Spiliopoulou, 2003]. The work introduced in [Xue *et al.*, 2002] proposes a re-ranking method with the help of website taxonomy to mine for generalized association rules and abstract access patterns of different levels to improve the performance of site search. The authors in [Yang *et al.*, 2002] propose an approach for predicting web log accesses based on association rule mining. Association rule mining facilitates the identification of related pages or navigation patterns which can be used in web personalization [Mobasher *et al.*, 2001][Mobasher *et al.*, 2000].

3.5 Sequential Pattern Mining

In sequential pattern mining a sequence of actions or events is determined with respect to time or other sequences [Velásquez *et al.*, 2003]. In web usage mining, sequential pattern mining could be used to predict user's future visit behaviors. Some web usage mining and analysis tools use sequential pattern mining to extract interesting patterns such as *SpeedTracer* [Wu *et al.*, 1998], and *WEBMINER* [Cooley *et al.*, 1997]. The authors in[Buchner *et al.*, 1999] suggest using adaptive websites to attract customers using sequential patterns to display special offers dynamically to them.

4 Data Preprocessing

Before data mining techniques are applied to web log file data, several preprocessing steps should be done in order to make web log file data ready to be mined. Web log file contains data about requested URL, time and date of request, method used, etc. The main data preprocessing tasks are data cleaning and filtering, path completion, session identification, and session formatting.

4.1 Data Cleaning

Data cleaning is the first preprocessing task. It involves the removal or elimination of irrelevant items that are not important for any type of web log analysis. Elimination of irrelevant items can be accomplished by checking the suffix of the URL name to filter out requests for graphics, sound, and video hits in order to concentrate on data representing actual page hits [Cooley *et al.*, 1997][Zaiane *et al.*, 1998]. For example, all log entries with filename suffixes such as gif, jpeg, and jpg can be removed. Another cleaning process is removing log entries generated by web agents like web spiders, indexers, or link checkers [Zaiane *et al.*, 1998]. Filtering out failed server requests, or transforming server error code is also done. Merging logs from multiple servers and parsing the log into data fields is also considered a data cleaning step [Cooley, 2003].

4.2 Path Completion

Path completion preprocessing task fills in page references that are missing due to local browsing caching such as using the back button available in the browser to go back to previously visited page [Cooley *et al.*, 1999].

4.3 User Identification

Identifying unique users is a complex step due to the existence of local caches, corporate firewalls, and proxy servers [Cooley *et al.*, 1997]. If the agent log shows a change in browser software, or operating system, a reasonable assumption to make is that each different IP address in the log file represent a different user [Pierrakos *et al.*, 2003]. If a page is requested that is not directly reachable by a hyperlink from any of the pages visited by the user, a heuristic assume that there is another user with the same IP address. Another assumption can be made is that consecutive accesses from the same host during a certain time interval come from the same user [Eirinaki and Vazirgiannis, 2003]. In some cases it is difficult to identify users, for example, when two users use the same machine and the same browser with the same IP address and look at the same set of pages [Cooley *et al.*, 1999].

4.4 Session Identification

A user session is defined as *the set of pages visited by the same user within the duration of one particular visit to a website* [Pierrakos *et al.*, 2003]. Session identification is dividing the page accesses of each user into individual sessions. One approach to identify user sessions, is by using a timeout threshold that is if the time between pages requests exceeds a certain limit (e.g. 30 minutes), then the user is starting a new session [Cooley *et al.*, 1999][Catledge and Pitkow, 1995]. Another approach assumes that consecutive accesses within the same time period belong to the same session [Eirinaki and Vazirgiannis, 2003].

4.5 Session Formatting

A final preprocessing step could be formatting the sessions or transactions for the type of the data mining technique, or algorithm to be applied [Cooley *et al.*, 1999]. The WEBMINER in [Cooley *et al.*, 1997] formats the cleaned web server log data in order to apply either association rule mining, or sequential pattern mining.

5 Web Usage Mining for Adaptive Websites

Adaptive websites are *websites that semi-automatically improve their organization and presentation by learning from user access patterns* [Perkowitz and Etzioni, 1998]. A site ability to adapt should be enhanced with information about its content, structure, and organization. For example, to add a link to a list of links ordered alphabetically, the link should be added at a specific point in the list. In the following subsections, we categorize different approaches of adaptive websites, even though it is difficult to make borders between different adaptation approaches for example, improving website links yields consequently to improve the structure of the website.

5.1 Improving Site Usability and Organization

Improving site usability can be achieved through making changes to the organization of the pages and links of the website. The work in [Fu *et al.*, 2001] aims to build an adaptive website that will reorganize its pages so that its users can find the information they want with minimum effort, where *effort* is defined in [Perkowitz and Etzioni, 1998] as *a function of the number of links traversed and the difficulty of finding that links in website pages*. Reorganization process is done by firstly extracting access patterns from web server's log file. Secondly, the web pages

in the web sever are classified into index pages and content pages based on the characteristics and access statistics of the pages. Finally, the whole website is analyzed and a reorganization of the website is presented based on access information and page classification. The authors in [Srikant and Yang, 2001] propose an algorithm to automatically find pages in a website whose location is different from where visitors expect to find them. The expected locations are then presented to the website administrator to add a navigation link from the expected location to the target page. The authors also present another algorithm to select the set of navigation links to optimize the benefit to the website or the visitor. The authors in [Spiliopoulou and Pohle, 2001] present a model to improve the success of the web site with the help of data mining techniques. To evaluate the efficiency values of a site pages, the authors analyze the navigational behavior of the site visitors with web usage mining. The analyst may decide to perform navigation pattern discovery over the entire log or to split it into customer log, or non-customer log and performs a comparative analysis of the two. Then makes decisions depending on the discovered results. The authors in [Mikroyannidis and Theodoulidis, 2005] introduced a framework that enables adaptation of the web topology and ontology to the needs and interests of web users. The proposed adaptation process exploits the access data of the users, together with the semantic aspect of the web, in order to facilitate web browsing.

5.2 Adaptive Content

Changing the content of a website can make the website better serve the requirements of a specific user. Content may be added, removed, or rearranged [Kilfoil *et al.*, 2003]. This includes additional explanations or details which may be added or removed depending on user's background and interests in some topic, or changing the website presentation language based on the user language preference.

5.3 Adaptive Link

Making changes to the links of the website can facilitate user's navigation of the website and minimize the time required to reach the target page. There are several techniques for adaptive link such as direct guidance, link sorting, link hiding, disabling, or highlighting. Direct guidance technique provides the user with a link to the page which is predicted to be the best next step for the user [Brusilovsky, 1997]. The AVANTI project [Fink *et al.*, 1996] tries to predict user's goals and presents links leading directly to pages it thinks a user will want to see. The work in [Wexelblat., 1996] proposes an approach to suggest a path to unexperienced users if many users follow the same path in their search for information. Link sorting is done by selecting the most relevant pages based on the users interests or goals then sorting them based on their relevance and presenting them in an ordered list of links [Kilfoil *et al.*, 2003][Brusilovsky, 1997]. Hiding or disabling the links that are not relevant to the user interests and goals makes the user less confused and speeds up user's navigation [Kilfoil *et al.*, 2003][Brusilovsky, 1996]. Link highlighting can also facilitate user's navigation [Brusilovsky, 1997][Brusilovsky, 1996].

5.4 Adaptive Web Structure

Adding or removing new pages is a final decision of the website administrator. Depending on the extracted usage patterns, several changes may be done on website structure. The authors in [Perkowitz and Etzioni, 1998] investigate the creation of index pages, which are pages that contain a direct link to pages that cover a particular topic, to facilitate the user's navigation of the website. The *PageGather* cluster mining algorithm is introduced. It takes web server logs as input and finds collections (clusters) of pages that tend to co-occur in visits, and outputs the contents of candidate index pages for each cluster found. A further development to [Perkowitz and Etzioni, 1998] is found in [Perkowitz and Etzioni, 2000] by presenting the *IndexFinder* a conceptual clustering mining algorithm in which all discovered clusters have intuitive descriptions that can be expressed to human users to solve the problem that *PageGather* gives no guarantee that all objects in the discovered cluster are about the same topic. To measure the use of a set of pages [Kilfoil *et al.*, 2003] statistics about commonly viewed pages, and subsets of pages is generated. The administrator can get an idea how the structure of the web should be, and whether there are some pages need to be removed, added, or their position need to be changed, without destroying the overall structure of the website.

5.5 Adaptive E-Commerce

Web usage mining has a great effect on e-commerce. It can be used to study customer behavior in the web, and use the extracted knowledge to facilitate navigation and services introduced to the customer, and suggest some particular products to the customer based on his interests. In [Berendt and Spiliopoulou, 2000] comparisons of navigation patterns between customers and non-customers lead to rules that specify how the website should be improved. The work in [Buchner *et al.*, 1999] suggests using adaptive websites to attract customers using sequential patterns to display special offers dynamically to them, and to keep the online shopper as loyal as possible. An example of e-commerce site that uses personalization is amazon.com, in which recommendations are presented to different customers depending on the customer profile [Au, 2002].

In order to make websites more effective to website users, they should reflect their interests, knowledge, needs, and goals. This can be done through personalization which is the subject of the next section.

6 Web Usage Mining for Personalized Websites

Web personalization is the process of customizing websites to the needs of specific users taking advantage from the patterns discovered from mining web usage data and other information such as web structure, web content, and user profile data [Eirinaki and Vazirgiannis, 2003]. Web personalization begins with the collection of web data. In this stage usage data are collected from different sources such as web server side data, client side data, and proxy servers. In general, personalization techniques are divided into offline, and online techniques. Offline personalization is based on simple user profiling and manual decision rule systems. Web usage mining is an online personalization data source. By evaluating site behavior and usage, a view about the website user is gained which yields to a more effective personalization strategies. User profiles are an important source of data for data personalization. User profiles contain user preferences, characteristics, interests knowledge, skills, activities, and behavioral patterns [Koutri *et al.*, 2005]. Such information is obtained

either explicitly using online registration forms and questionnaires resulting in static user profiles, or implicitly by recording the navigational behavior and/or the preferences of each user resulting in dynamic user profiles [Eirinaki and Vazirgiannis, 2003].
There are different ways to analyze the collected data. Content based filtering methods select content items that have a high degree of similarity to the user's profile [Vassiliou *et al.*, 2002]. An alternative to content based filtering is the collaborative filtering techniques which allow users to take advantage of other users behavioral activities based on a measure of similarity between them [Vassiliou *et al.*, 2002][Kim *et al.*, 2004]. Rule based filtering allows website administrators/marketers to specify business rules based on user demographics. The rules are used to affect the content introduced to a particular user.
Pattern discovery is the next step of the personalization process. In this step, different data mining techniques, such as clustering, classification, association rule mining, and sequential pattern analysis, are used to discover interesting patterns from web usage data.
Clustering is used to group users with common browsing behavior. The authors in [Shahabi *et al.*, 1997] implement a *Profiler* system which captures client's selected links, page order, page viewing time, and cache references. That information are used to cluster users with similar interests. The work in [Mobasher *et al.*, 2000] proposes a recommendation engine which considers the association rules between different web pages, and the derivation of URL clusters based on two types of clustering techniques in conjunction with the active user session. The recommendations are then added to the last requested page as a set of links before the page is sent to the client browser.
Association rules or sequential pattern discovery methods facilitate the identification of related pages or navigation patterns which can be used subsequently to recommend new web pages to the visitors of a website. The work in [Mobasher *et al.*, 2001] provides a framework for web personalization based on association rule mining from clickstream data. [Ishikawa *et al.*, 2002] introduces the *System L-R* recommendation system which constructs user models by classifying the web access and recommends relevant pages to the users based both on the user models and the web content.
[Pierrakos *et al.*, 2001] presents a web usage mining system *KOINOTITES* which uses web usage mining techniques to identify groups of users who have similar navigation behavior. The produced information can either be used by the administrator in order to improve the structure of the website or it can be fed directly to a personalization model, (e.g., collaborative filtering). The work in [Albanese *et al.*, 2004] proposes a web mining strategy for web personalization based on a novel pattern recognition strategy which analysis and classifies users taking into account both user provided data and navigational behavior of the users. It presents the *Refferer Based Page Recommendation, RBPR,* that uses information about a visitor's browsing context (specially, the refferer URL provided by the HTTP) to suggest pages that might be relevant to the visitors underlying information need.
The authors in [Kushmerick *et al.*, 2000] introduce a different approach of personalization that requires no input or feedback from the user. The work in [Vassiliou *et al.*, 2002] suggests a set of steps that make the personalization process effective starting from data collection and management efforts, to measuring and evaluating the success of personalization.

7 Selected Web Mining and Analysis Tools

In this section we present some well-known web usage mining and analysis tools such as WUM, SpeedTracer, Weblogminer, WebMiner, WebWatcher, and WebPersonalizer.

- ***Web Utilization Miner (WUM)*:** the system discovers the navigational patterns [Spiliopoulou and Faulstich, 1998]. A human expert specify the generic structural and statistical characteristics that makes a navigation pattern interesting to improve the organization of the web documents and adapt it better to the needs of users.
- ***SpeedTracer*:** SpeedTracer [Wu *et al.*, 1998] reconstructs user traversal paths to identify user sessions. It uses association rule mining and sequential patterns to present statistics about users, the most frequently visited pages, the distribution of user session durations, the number of visited pages, the most frequently traversed paths, and the most frequently visited groups of pages.
- ***WebLogMiner*:** WebLogMiner is a general web usage mining tool [Zaiane *et al.*, 1998]. It consists of four steps. The first step is filtering web log file and creating a relational database for the filtered information, containing different attributes such as user, resource, day, etc. In the second step, a data cube is constructed using the available dimensions. Then, online analytical processing OLAP techniques are used on the web log data cube. Finally, data mining techniques such as data characterization and comparison, statistical analysis, classification, and time series analysis are put to use with the data cube to predict, classify, and discover interesting correlations.
- ***WEBMINER*:** WEBMINER [Cooley *et al.*, 1997] uses sequential pattern mining, and association rule mining techniques. Before any knowledge discovery technique takes place, the web server log data is cleaned. The resulting data is then formatted according to the data mining technique need to be used. Instead of mining for all patterns, a Query Mechanism is used to limit the search to relevant and useful patterns. The resulting patterns are used for web restructuring, and personalization.
- ***WebWatcher*:** WebWatcher is a personalization web mining tool [Joachims *et al.*, 1997]. It is a tour guide agent that provides navigational hints to the user while browsing the website such as highlighting interesting links, based on user interests, and the content of the web pages. The system learns from the earlier tours to improve the recommendation giving skills.
- ***WebPersonalizer*:** after web log data are preprocessed data mining techniques such as association rule mining, sequential pattern discovery, clustering, and classification are applied to discover interesting patterns. The results are then used to create aggregated usage profiles in order to create decision rules. After matching each user activity and those usage profiles, a list of recommended links are provided to the user [Mobasher, 2001].

8 Summary and Research Direction

In this paper, we introduced an overview about various research issues in web usage mining and its application in the field of adaptive, and personalized web sites.
In this paper, different approaches for reorganizing and improving the design of the websites based on user navigational patterns, and user profiles have been discussed. Most of that approaches are semi-automatic i.e., they need user, or website administrator interaction in order to complete the adaptation process, such as questionnaires filled by user, or when some interesting navigational patterns are discovered, the website administrator use that patterns to make decisions to adapt the website. Different approaches try to make the adaptation procedure as automatic as possible. Other approaches try to find new measures that reflect further characteristics of website usage, and improve pattern analysis by improving visualization tools to make it easier for the analyst to understand the extracted patterns. A new approach to enhance web personalization is by making some psychological studies on user profiles which yields to better personalized websites.
Another approach for website adaptation is making simultaneous adaptation of multiple websites. That websites have something common with each other or belong to the same category of websites. Another similar approach is dealing with the website as a collection of gates. Each gate represent a general subject (i.e. sport, medicine, computer, etc.), and within every gate there are some recommended links for the web navigator. Those recommendations include links to related websites that may be of interest to the web navigator. The recommendations are done depending on the log history of previous users.

References

[Albanese *et al.*, 2004] Massimiliano Albanese, Antonio Picariello, Carlo Sansone, and Lucio Sansone. A Web Personalization System Based on Web Usage Mining Techniques. In *WWW Alt. 2004*, pages 288–289. ACM Press, 2004.

[Au, 2002] SaiMing Au. A Study of Application of Web Mining for E-Commerce: Tools and Methodology. *International Journal of The Computer, The Internet and Management*, 10(3):1–14, 2002.

[Baron and Spiliopoulou, 2003] Steffan Baron and Myra Spiliopoulou. Monitoring the Evolution of Web Usage Patterns. In *EWMF*, pages 181–200, 2003.

[Berendt and Spiliopoulou, 2000] Bettina Berendt and Myra Spiliopoulou. Analysis of Navigational Behavior in Web Sites Integrating Multiple Information System. *VLDB Journal*, 9:56–75, 2000.

[Borges and Levene, 1999] Jose Borges and Mark Levene. Data Mining of User Navigation Patterns. In *WEBKDD*, pages 92–111, 1999.

[Brusilovsky, 1996] Peter Brusilovsky. Methods and Techniques of Adaptive Hypermedia. *User Modeling and User-Adapted Interaction*, 6:87–129, 1996.

[Brusilovsky, 1997] Peter Brusilovsky. Efficient Techniques for Adaptive Hypermedia. In *Intelligent Hypertext*, pages 12–30, 1997.

[Buchner *et al.*, 1999] Alex Buchner, Maurice D. Mulvenna, Sarab S. Anand, and John G. Hughes. An Internet-enabled Knowledge Discovery Process. Technical report, MINEit Software Ltd., 1999.

[Cai *et al.*, 2004] Ding Cai, Xiaofei He, Ji-Rong Wen, and Wei-Ying Ma. Block-Level Link Analysis. In *Proc. of SIGIR 2004*, pages 440–447. ACM Press, 2004.

[Catledge and Pitkow, 1995] Lara D. Catledge and James E. Pitkow. Characterizing Browsing Strategies in the World Wide Web. *Computer Networks and ISDN Systems*, 27(6):1065–1073, 1995.

[Cohen, 2003] William W. Cohen. Learning and Discovering Structure in Web Pages. *Bulletin of the IEEE Computer Society Technical Committee on Data Engineering*, 26(1):3–10, 2003.

[Cooley *et al.*, 1997] R. Cooley, J. Srivastava, and B. Mobasher. Web Mining: Information and Pattern Discovery on the World Wide Web. In *Proc. of ICTAI 1997*, page 558, Nov. 1997.

[Cooley *et al.*, 1999] Robert Cooley, Bamshad Mobasher, and Jaideep Srivastava. Data Preparation for Mining World Wide Web Browsing Patterns. *Knowledge and Information Systems*, 1(1):5–32, 1999.

[Cooley, 2003] Robert Cooley. The Use of Web Structure and Content to Identify Subjectively Interesting Web Usage Patterns. *ACM Trans. Inter. Tech.*, 3(2):93–116, 2003.

[Davison, 2002] Brian D. Davison. Predicting Web Actions from HTML Content. In *Proc. of the The 13th ACM Conf. on Hypertext and Hypermedia*, pages 159–168, College Park, MD, Jun 2002.

[Dhyani *et al.*, 2002] D. Dhyani, W. Keong, and N. Bhowmick. A Survey of Web Metrics. In *ACM Computing Surveys*, volume 34, pages 469–503, 2002.

[Eirinaki and Vazirgiannis, 2003] Magdalini Eirinaki and Michalis Vazirgiannis. Web Mining for Web Personalization. *ACM Transactions for Internet Technology*, 3(1):1–27, 2003.

[Ester *et al.*, 2002] Martin Ester, Hans-Peter Kriegel, and Matthias Schubert. Web Site Mining a New Way To Spot Competitors, Customers and Suppliers in The World Wide Web. In *Proc. 8th ACM SIGKDD KDD 2002*, pages 249–258, Edmonton, CA, 2002. ACM Press.

[Fink *et al.*, 1996] J. Fink, A. Kobsa, and A. Nill. User-oriented Adaptivity and Adaptability in the Avanti Project. In *In Proc. of the Designing for the Web Conf: Empirical Studies, Microsoft Campus Redmond, USA*, 1996.

[Fu *et al.*, 2001] Y. Fu, M. Creado, and M. Shih. Adaptive Web Sites by Web Usage Mining. In *Int. Conf. on Internet Computing (IC 2001)*, Las Vegas NA, 2001.

[Gedov *et al.*, 2004] Vassil Gedov, Carsten Stolz, Ralph Neuneier, Michal Skubacz, and Dietmar Seipel. Matching Web Site Structure and Content. In *Proc. WWW 2004*, pages 286–287. ACM Press, 2004.

[Han and Kamber, 2001] Jiawei Han and Micheline Kamber. *Data Mining Concepts and Techniques*. Morgan Kaufmann Publishers, San Francisco, 2001.

[Ishikawa *et al.*, 2002] Hiroshi Ishikawa, Manabu Ohta, Shohei Yokoyama, Junya Nakayama, and Kaoru Katayama. On the Effectiveness of Web Usage Mining for Page Recommendation and Restructuring. In *Web,*

Web-Services, and Database Systems, pages 253–267. Springer-Verlag, 2002.

[Joachims *et al.*, 1997] Thorsten Joachims, Dayne Freitag, and Tom M. Mitchell. WebWatcher: A Tour Guide for the World Wide Web. In *IJCAI (1)*, pages 770–777, 1997.

[Kilfoil *et al.*, 2003] M. Kilfoil, A. Ghorbani, W. Xing, Z. Lei, J. Lu, J. Zhang, and X. Xu. Toward an Adaptive Web: The State of the Art and Science. In *Proc. of CNSR 2003*, pages 119–130, Moncton, Canada, 2003.

[Kim *et al.*, 2004] Dong-Ho Kim, Vijayalakshmi Atluri, Michael Bieber, Nabil Adam, and Yelena Yesha. A Clickstream-based Collaborative Filtering Personalization Model: Towards a Better Performance. In *WIDM 2004*, pages 88–95. ACM Press, 2004.

[Kosala and Blockeel, 2000] Raymond Kosala and Hendrik Blockeel. Web mining research: A survey. *ACM SIGKDD Explorations*, 2(1):1–15, 2000.

[Koutri and Daskalaki, 2003] Martha Koutri and Sophia Daskalaki. Improving Web Site Usability Through a Clustering Approach. In *10th International Conference on Human-Computer Interaction HCI, Crete, Greece*, pages 11–19, 2003.

[Koutri *et al.*, 2005] M. Koutri, N. Avouris, and S. Daskalaki. A Survey on Web Usage Mining Techniques for Web-based Adaptive Hypermedia Systems. In *Adaptable and Adaptive Hypermedia Systems, IRM Press*, pages 125–149, 2005.

[Kushmerick *et al.*, 2000] Nicholas Kushmerick, James McKee, and Fergus Toolan. Towards Zero-Input Personalization: Referrer-Based Page Prediction. *Lecture Notes in Computer Science*, 1892:133–143, 2000.

[Lin and Ho, 2002] Shian-Hua Lin and Jan-Ming Ho. Discovering Informative Content Blocks from Web Documents. In *Proc. ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*, pages 588–593. ACM Press, 2002.

[Liu *et al.*, 2001] Bing Liu, Yiming Ma, and Philip S. Yu. Discovering Unexpected Information from Your Competitors' Web Sites. In *Proc. 7th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*, pages 144–153. ACM Press, 2001.

[Liu *et al.*, 2003] Bing Liu, Robert Grossman, and Yanhong Zhai. Mining Data Records in Web Pages. In *Proc. 9th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*, pages 601–606. ACM Press, 2003.

[Mikroyannidis and Theodoulidis, 2005] Alexander Mikroyannidis and Babis Theodoulidis. Web Usage Driven Adaptation of the Semantic Web. In *End User Aspects of the Semantic Web Workshop, 2nd European Semantic Web Conference (ESWC 2005)*, Heraklion, Greece, 2005.

[Mobasher *et al.*, 2000] Bamshad Mobasher, Robert Cooley, and Jaideep Srivastava. Automatic Personalization Based on Web Usage Mining. *Commun. ACM*, 43(8):142–151, 2000.

[Mobasher *et al.*, 2001] Bamshad Mobasher, Honghua Dai, Tao Luo, and Miki Nakagawa. Effective Personalization Based on Association Rule Discovery from Web Usage Data. In *WIDM 2001*, pages 9–15, NY, USA, 2001. ACM Press.

[Mobasher, 2001] Bamshad Mobasher. WebPersonalizer: A Server-Side Recommender System Based on Web Usage Mining. Technical Report TR01-010, School of Computer Science, Telecommunications and Information Systems, DePaul University, Chicago, IL, USA, 2001.

[Morinaga *et al.*, 2002] Satoshi Morinaga, Kenji Yamanishi, Kenji Tateishi, and Toshikazu Fukushima. Mining Product Reputations on the Web. In *Proc. of the 8th ACM SIGKDD int. conf. on Knowledge discovery and data mining*, pages 341–349. ACM Press, 2002.

[Olga. *et al.*, August 1999] Nasraoui Olga., H. Frigui, A. Joshi, and R. Krishnapuram. Mining Web Access Logs Using Relational Competitive Fuzzy Clustering. In *Proc. of the 8th Int. Fuzzy Systems Association Congress*, Hsinchu, Taiwan, August 1999.

[Perkowitz and Etzioni, 1998] Mike Perkowitz and Oren Etzioni. Adaptive Web Sites: Automatically Synthesizing Web Pages. In *Proc. 15th National Conference on Artificial Intelligence and 10th Innovative Applications of Artificial Intelligence Conference (AAAI 98 / IAAI 98)*, pages 727–732. AAAI Press/The MIT Press, 1998.

[Perkowitz and Etzioni, 2000] Mike Perkowitz and Oren Etzioni. Adaptive Web Sites. *Communications of the ACM*, 43(8):152–158, 2000.

[Pierrakos *et al.*, 2001] Dimitrios Pierrakos, Geogios Paliouras, Christos Papatheodouro, and Constantine D. Spyropoulos. KOINOTITES: A Web Usage Mining Tool for Personalization. In *Proc. of the Panhellenic Conf. on Human Computer Interaction*, 2001.

[Pierrakos *et al.*, 2003] Dimitrios Pierrakos, Georgios Paliouras, Christos Papatheodorou, and Constantine D. Spyropoulos. Web Usage Mining as a Tool for Personalization: A Survey. *User Modeling and User-Adapted Interaction*, 13(4):311–372, 2003.

[Shahabi *et al.*, 1997] C. Shahabi, A. M. Zarkesh, J. Adibi, and V. Shah. Knowledge Discovery from Users Web Page Navigation. In *Proc. of the 7th Int. Workshop on Research Issues in Data Engineering RIDE 1997*, page 20, Washington, DC, USA, 1997. IEEE Computer Society.

[Spiliopoulou and Faulstich, 1998] Myra Spiliopoulou and Lukas C. Faulstich. WUM: a Web Utilization Miner. In *Workshop on the Web and Data Bases (WebDB 1998)*, pages 109–115, 1998.

[Spiliopoulou and Pohle, 2001] Myra Spiliopoulou and Carsten Pohle. Data Mining for Measuring and Improving the Success of Web Sites. *Data Min. Knowl. Discov.*, 5:85–114, 2001.

[Srikant and Yang, 2001] Ramakrishnan Srikant and Yinghui Yang. Mining Web Logs to Improve Web Site Organization. In *Proc. WWW 2001*, pages 430–437, Hong Kong, 2001. ACM Press.

[Srivastava *et al.*, 2000] Jaideep Srivastava, Robert Cooley, Mukund Deshpande, and Pang-Ning Tan. Web Usage Mining: Discovery and Applications of Usage Patterns from Web Data. *ACM SIGKDD Explorations*, 1(2):12–23, 2000.

[Su *et al.*, 2002] Zhong Su, Qiang Yang, Hong Jiang Zhang, Xiaowei Xu, Yu-Hen Hu, and Shaoping Ma.

Correlation-Based Web Document Clustering for Adaptive Web Interface Design. *Knowledge and Information Systems*, 4(2):151–167, 2002.

[Vassiliou *et al.*, 2002] Charalampos Vassiliou, Dimitris Stamoulis, and D. Martakos. The Process of Personalizing Web Content: Techniques, Workflow and Evaluation. In *Proc. of the SSGRR 2002w*, 2002.

[Velásquez *et al.*, 2003] Juan Velásquez, Hiroshi Yasuda, and Terumasa Aoki. Combining the Web Content and Usage Mining to Understand the Visitor Behavior in a Web Site. In *Proc. ICDM 2003*, pages 669–672. IEEE Computer Society Press, 2003.

[Wexelblat., 1996] Alan Wexelblat. An Environment for Aiding Information-browsing Tasks. In *In Proc. of AAAI Spring Symposium on Acquisition, Learning and Demonstration: Automating Tasks for Users*, Birmingham, UK, 1996. AAAI Press.

[Wu *et al.*, 1998] K.-L. Wu, P. S. Yu, and A. Ballman. SpeedTracer: A Web Usage Mining and Analysis Tool. *IBM Systems Journal*, 37(1), 1998.

[Xue *et al.*, 2002] Gui-Rong Xue, Hua-Jun Zeng, Zheng Chen, Wei-Ying Ma, and Chao-Jun Lu. Log Mining to Improve the Performance of Site Search. In *Proc. WISE 2002 Workshops*, pages 238–245, Singapore, 2002. IEEE Computer Society Press.

[Yang *et al.*, 2002] H. Yang, S. Parthasarathy, and S. Reddy. On the Use of Constrained Associations for Web Log Mining. In *WEBKDD Workshop: Web Mining for Usage Patterns and User Profiles*, pages 100–118, 2002.

[Zaiane *et al.*, 1998] Osmar R. Zaiane, Man Xin, and Jiawei Han. Discovering Web Access Patterns and Trends by Applying OLAP and Data Mining Technology on Web Logs. In *Proc. of ADL 1998*, pages 1–9, Washington, DC, USA, 1998. IEEE Computer Society.

Pattern recognition of gene expression data on biochemical networks with simple wavelet transforms

Gunnar Schramm[1,2], Marcus Oswald[3], Hanna Seitz[3], Sebastian Sager[4], Marc Zapatka[2], Gerhard Reinelt[3], Roland Eils[1,2] and Rainer König[1,2]
[1] Department of Bioinformatics and Functional Genomics, Institute for Pharmacy and Molecular Biotechnology, University of Heidelberg; [2] Theoretical Bioinformatics, German Cancer Research Center (DKFZ); [3] Institute of Computer Science, University of Heidelberg; [4] Interdisciplinary Center for Scientific Computing, University of Heidelberg
69120 Heidelberg, Germany
g.schramm@dkfz.de, Marcus.Oswald@Informatik.Uni-Heidelberg.de, Hanna.Seitz@Informatik.Uni-Heidelberg.de, Sebastian.Sager@iwr.uni-heidelberg.de, m.zapatka@dkfz.de, Gerhard.Reinelt@Informatik.Uni-Heidelberg.de, r.eils@dkfz.de, r.koenig@dkfz.de

Abstract

Biological networks show a rather complex, scale-free topology consisting of few highly connected (hubs) and many low connected (peripheric and concatenating) nodes. Furthermore, they contain regions of rather high connectivity, as in e.g. metabolic pathways. To analyse data for an entire network consisting of several thousands of nodes and vertices is not manageable. This inspired us to divide the network into functionally coherent sub-graphs and analysing the data that correspond to each of these sub-graphs individually. We separated the network in a two-fold way: 1. clustering approach: sub-graphs were defined by higher connected regions using a clustering procedure on the network; and 2. connected edge approach: paths of concatenated edges connecting striking combinations of the data were selected and taken as sub-graphs for further analysis. As experimental data we used gene expression data of the bacterium *Escherichia coli* which was exposed to two distinctive environments: oxygen rich and oxygen deprived. We mapped the data onto the corresponding biochemical network and extracted disciminating features using Haar wavelet transforms for both strategies. In comparison to standard methods, our approaches yielded a much more consistent image of the changed regulation in the cells. In general, our concept may be transferred to network analyses on any interaction data, when data for two comparable states of the associated nodes are made available.

1 Introduction

Modern high throughput methods in biotechnology allow to profile cellular mechanisms not only for a single, focused, but for rather large numbers of aspects and compounds. This is especially given by the DNA microarray technology. It allow us to explore the expression levels for a major subset or all genes of an organism under a variety of conditions such as alternative treatments, mutants, developmental stages and time points. For example, the technique enables us to classify tumour samples [Van 'T Veer, *et al.*, 2002], to define small sets of potential marker genes to distinguish leukemias [Stephanopoulos, *et al.*, 2002], and to discover regulatory mechanisms [Gasch, *et al.*, 2000, Spellman, *et al.*, 1998]. In general, such studies use supervised methods to support diagnosis [Stephanopoulos *et al,* 2002, Van 'T Veer *et al*., , 2002] and yield gene lists that are crucial for the classifications [Thuerigen, *et al.*, 2006]. However, lists of single genes are rather tedious to analyse for yielding a general functional meaning. Therefore, methods were developed that map these gene lists onto functionally relevant reaction and signaling cascades [Manoli, *et al.*, 2006]. Furthermore, regulatory networks could be defined with co-expressed genes. E.g., without prior information, the structure and function of the network that regulates the SOS pathway in *E. coli* could be elucidated with transcription profiles [Gardner, *et al.*, 2003]. Another way to easier extract the functionality of expression patterns is to first map the data onto networks that consist of nodes bearing the expression data and vertices that link nodes with common functionality and directed or undirected interdependence. E.g. physically interacting proteins may be related functionally by being succeeding nodes in a signaling cascade. Such knowledge of protein-protein interaction from high-throughput techniques [Uetz, *et al.*, 2000] was applied to analyse gene expression data and revealed novel regulatory circuits [Ideker, *et al.*, 2002]. On a statistical basis, this data is useful for inferring changed signal transduction of e.g. diseased situations. Besides this, over the last four decades, biochemical investigations have discovered an increasingly consistent image of the cellular metabolism (see e.g. [Berg, *et al.*, 2002]). These biochemical reactions can be functionally linked together by setting a reaction r_i as the precursor of a reaction r_j if and only if one of the reaction-products of r_i is needed as a substrate for r_j. This yields a biochemical or metabolic network. Microarray expression data for each gene can then be mapped on its corresponding enzyme and the reaction the enzyme it is catalysing. Such interaction knowledge from the biochemical network has been used to support the clustering procedure for gene expression profiles of yeast [Hanisch, *et al.*, 2002, Zien, *et al.*, 2000]. Pattern analyses on such networks advantage from

the fact that such interactions are well defined and established. This is especially true for less complex organisms such as yeast or *Escherichia coli* [Karp, *et al.*, 2002]. Simple clustering of gene expression data on these metabolic networks can yield sub-graphs that are either commonly stimulated or repressed as we showed previously for tryptophan treated cells [König and Eils, 2004]. With this method we were able to find the biosynthesis pathway of tryptophan as an expression pattern in the network having a common response to the environmental changes. Note that such a clustering method discovers patterns of co-expressed genes. We now developed methods enabling the discovery of more complex patterns. As a case study, we investigated the response of the hetero-fermentative bacterium *E. coli* in response to oxygen deprivation. The regulatory machinery can react on this environmental change in different ways. One basic response changes the catabolism of glucose, switching off or down-regulating the respiratory sub-graphs such as the glyoxylate cycle and switching on the fermentation and production of acid end products (see e.g. [Neidhardt, 1996]). This is supported by several signalling concepts, e.g. by inducing inhibitors for glyoxylate cycle (TCA) genes, down-regulating glyoxylate cycle genes or activating and up-regulating genes for the fermentation processes.
Within the first approach (clustering approach), we performed a clustering of the network without gene expression data to define regions with high connectivity. Gene expression data was mapped onto these regions. We wanted to explore all possible relevant expression level combinations in these regions. For this, we used the adjacency matrix representations for these regions, mapped the data onto them and calculated simple Haar wavelet transforms applying a standard procedure for two-dimensional images (see e.g. [Theodoridis and Koutroumbas, 1998]). The extracted features were ranked due to their ability to distinguish between the two environmental conditions (oxygen rich versus oxygen deprived). In so doing, we were able to reveal interesting switches that are posted at process bifurcations, in rather good agreement to the expected anaerobic response. However, with this approach we were only able to extract interesting data patterns in highly connected sub-graphs. Therefore, we applied a second method to reveal crucial data patterns also of linearly ordered reactions. Features were generated by applying the one dimensional Haar-wavelet transform onto each pair of nodes. With this method we were able to detect expected up-regulated pathways of formate fermentation and C6 nutrients metabolism. Furthermore, our method revealed a down-regulation of the iron processing parts of the metabolic network as well as the up-regulation of the histidine biosynthesis pathway which constitutes a response for enriched acidic products during anaerobic growth.

2 Methods

The description of the clustering method will be briefly described here. For a description in detail, see [König, *et al.*, 2006, Schramm, *et al, in prep.*]. Metabolic reactions were extracted from the EcoCyc database (Version 9, [Keseler, *et al.*, 2005]). A graph was established by defining neighbours of metabolites. Two metabolites were neighbours if and only if an enzymatic reaction existed that needed one of the metabolites as input (needed substrate) and produced the other as output (product). Note, that in this representation, enzymes are edges and metabolites the nodes. This network was clustered to group enzymes into parts of the network with their major connections (the clustering algorithm is described in [König *et al.*, 2006]). The clustering algorithm produced a symmetrical sub-matrix of the cluster matrix for each cluster, whose rows and columns were the metabolites. The matrix contained a "1" entry at position (i, j) if an enzyme existed that combined metabolites of row i and column j. Otherwise a "0" entry was set.

2.1 Mapping gene expression data onto the cluster-matrices

For our case study, we collected raw intensity values of gene expression data from the work of Covert et al. [Covert, *et al.*, 2004]. We normalised them with an established variance normalisation method [Huber, *et al.*, 2002] and selected the data for 43 hybridisations of the following samples: strain K-12 MG 1655, wild-type, Δ*arcA*, Δ*appY*, Δ*fnr*, Δ*oxyR*, Δ*soxS* single mutants and the Δ*arcA*Δ*fnr* double mutant. The mutated genes are key transcriptional regulators of the oxygen response [Covert *et al*., 2004]. They effect a major portion of all genes in *E. coli* and therefore supported a variance stimulation of the respiratory and fermentative control of the investigated strain. All gene expression experiments were done in triplicate under aerobic and anaerobic conditions, respectively, except for anaerobic wild-type which was repeated four times. The gene expression data of each data-set was mapped onto the corresponding reactions of the transcribed proteins. Mean values were taken if a reaction was catalysed by a complex of proteins. The expression data of all samples was mapped onto each cluster-matrix, yielding 43 different patterns for each cluster.

2.2 Pattern discovery: defining the features with the Haar wavelet transform

We wanted to calculate a value for every possible expression pattern of neighbouring genes and groups of genes within a cluster that may show essential differences between samples of different conditions. Therefore, we performed a Haar-wavelet transform for each cluster-matrix. The wavelet transformed expression values served as features for the classifier (classification method, see next section). This allowed the identification of regions with a varying pattern between aerobic and anaerobic conditions. The wavelet-transformation is described in the following. Each cluster-matrix was divided into 2x2 pixeled disjoint sub-sections (e.g. a cluster matrix of size 8 x 8 was divided into 16 sub-sections). Clusters with non-fitting sizes (e.g. 3x3, 5x5, ...) were extended with rows and columns of zeros to yield matrices that could be divided into 2x2 pixeled sub-sections. For each sub-section, all combinations of row-wise and column-wise means and differences, respectively, were calculated. This yielded 4 combined values for each 2x2 pixeled sub-section: 1st: mean of the mean of the upper and mean of the lower row, 2nd: difference of the mean of the upper and the mean of the lower row, 3rd: mean of the difference of the upper and the difference of the lower row, and, 4th: difference of the difference of the upper and the difference of the lower row. All four combined values for each 2x2 pixeled sub-

section were stored and applied as features for the classifier. This was done for all sub-sections of the matrix. All 1st combined values (mean of means) were taken for a new matrix and were again grouped into 2x2 fractions that were combined in the same manner, yielding again 4 new features for every fraction. This procedure was repeated until no further grouping was possible. Such a "Haar" wavelet transform can be regarded as a low pass filter when calculating the mean, and a high pass filter when calculating the difference between neighbouring value pairs. The transform applied a filter in horizontal and subsequently in vertical direction. The procedure consisted of repeatedly applying high and low pass filters on the image. Therefore, either high frequency or low frequency portions of the signal were calculated and stored, until the maximal possible compositions were obtained. This procedure was carried out for all clusters of every sample and the results of the transforms were stored as the corresponding features for every sample.

2.3 Extracting essential features and their sub-graphs with the classifier

The SAM method [Tusher*, et al.*, 2001] as a modified t-test was performed to rank the features according to their p-values. Higher ranking features (low p-values) were selected focusing the classifier on the most relevant patterns (9,996 out of 70,912). For classification, we applied the Support Vector Machine implementation as provided by the R MCRestimate package [Ruschhaupt*, et al.*, 2004]. To receive a suitable feature extraction result, a 10-fold cross validation was performed and repeated 10 times with different splittings of the data, respectively. A linear kernel was applied for the feature extraction as described elsewhere [Ruschhaupt *et al*., 2004]. Parameter optimisation was performed for the regularisation term that defined the costs for false classifications (9 steps, range: 2^n, n = -4, -2, ..., 8, 10). This optimisation was realised by an internal three-fold cross validation during every iteration. To determine the most relevant features, a recursive feature elimination [Ruschhaupt *et al*., 2004] was applied during the parameter optimisation procedure. This yielded a set of discriminating features for every run. These features were ranked due to their selection frequency of all 100 runs. Note, that high-ranking features yielded the corresponding sub-graphs (cluster of the cluster matrix) of the reaction network that contained well discriminating patterns of the expression data. We defined a cut-off criterion for selecting only substantial features by comparing the selection frequency of each feature with random selections. We assumed a binomial distribution, neglecting the cases that the same feature may have been chosen twice in one run. The overall number of drawings was the sum of all selections (8,191 selections). The probability to draw the respective feature was the reciprocal value of the number of all features (1/9,996). The number of drawings for the respective feature was its selection frequency. As we calculated this for every feature, the resulting p-values where corrected for multiple testing by multiplying them with the number of all features (Bonferroni correction [Bonferroni, 1935a]).

2.4 Generating and assembling the features for the second approach

The Haar-wavelet was used to extract discriminating reaction pairs. We added and subtracted the values of neighbouring pairs of nodes yielding low pass and high pass filtered features for each pair, respectively. All generated features were ranked via a multiple t-test between aerobic and anaerobic conditions: to correct for potential influences coming from individual mutants, t-tests were performed for every constellation of samples excluding the sample of one particular mutant, respectively. The wild type sample was never excluded. From this outcome the worst (highest) p-value for each feature was selected. All p-values were corrected for multiple testing (Bonferroni [Bonferroni, 1935b]). Features were then ranked according to their p-value. Sub-graphs were put up by connecting found significant features (reaction-pairs) having a p-value ≤ 0.01. This resulted in 5 sub-graphs. To facilitate the interpretation of the found sub-graphs, nodes with equal expression behaviour (up-, down-regulation) were grouped together, and functionally described (see Results) if the group size was ≥ 5 focusing only on larger patterns. In total 10 such clusters were found. Reaction-pairs having one up- and one down-regulated node were regarded as switches. They were extracted if their p-value was ≤ 0.01 and are also functionally characterised (see Results).

3 Results

We will give a brief description of the functional meanings for the found sub-graphs focussing on integrative aspects (for details see [König *et al*., 2006, Schramm *et al*., *in prep*.]). From the clustering approach, we yielded 973 (not necessarily disjoint) clusters of sizes between 2 and 46 reactions and 160,264 features. After deleting features that consisted only of zeros, 70,912 features remained. A modified t-test was performed [Tusher et al., 2001] to reduce the remaining features and focus the classifier on the most relevant patterns. As a threshold, a false discovery rate of 2e-05 was chosen to further analyse the 9,996 most significant features. With these features, the SVM was trained and tested by a ten-time's ten-fold cross-validation. A recursive feature elimination [Ruschhaupt *et al.,* 2004] was applied for each run, yielding 100 lists of the most discriminating features. These features were ranked according to their selection frequency. 8,191 out of 9,996 features were selected at least once. To help us focusing on the most relevant features, only features with a significant selection frequency were used (p-value ≤ 0.05, in comparison to a random selection, Bonferroni corrected for multiple testing [Bonferroni, 1935a]). This yielded 181 features. Network clusters that contained these features were extracted and are further referred to as extracted clusters. They were listed in accordance to their selection frequency. Extracted clusters that contained less than six nodes were not considered to focus on larger patterns. Reactions were regarded as up-regulated (green in figures) if the corresponding genes were significantly up-regulated under anaerobic conditions (p-value ≤ 0.05 of a t-test), down-regulated if significantly down-regulated (red in figures), and not significantly differentially regulated otherwise (grey in figures, red/green frames indicate a non-significant tendency). Note that not differentially expressed nodes were discarded.

With the second approach (connected edge approach) we yielded 660 significantly discriminating reaction-pairs

(applied p-value cut-off: 0.01). Features that occurred twice due to the generation method were considered only once. All significant reaction-pairs were mapped onto the complete metabolic graph to extract sub-graphs consisting of connected pairs. In total, 5 such sub-graphs were identified consisting of 165 reactions.
Neighbouring, connected nodes, that showed identical regulation (up or down) were grouped together. Only groups of a minimum of five reactions were selected for functional interpretation focussing on major regulation patterns. We will refer to this groups as clusters in the following. Furthermore all significant switches were extracted. As switches were deemed pairs of reactions. We yielded 20 significant switches which were defined as reaction-pairs that showed opposing regulatory behaviour.

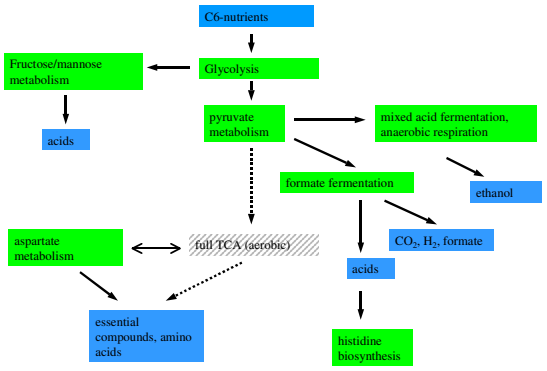


Figure 1. Carbohydrate metabolism and stress response to acids, up-regulated sub-graphs are green (filled), compounds are blue (dark). The TCA cycle (grey, fasciated) has got limited function for oxygen deprived conditions (see text).

3.1 Functional characterisation

Basically, the clustering approach found six clusters with crucial patterns in the following metabolic pathways: 1. formate fermentation, 2. Aspartate metabolism, 3. Lysine biosynthesis and C6 nutrients uptake, 4. Glycolysis and glucose storing, and 5. Glycolysis and NAD switch, 6. branched chain amino acid transporters. The connected edge approach found ten clusters: 1. formate fermentation and anaerobic electron transport chain, 2. Pyruvate metabolism, anaerobic synthesis of deoxyribonucleotides and electron transport, 3. C6 nutrients metabolism, i.e. glycolysis and fructose/mannose metabolism, 4. C6 nutrients metabolism: Glycolysis and Entner-Doudoroff pathway, 5. aerobic iron processing and transport, 6. aerobic iron processing: FE-S biogenesis, 7. histidine biosynthesis, aspartate metabolism and NAD switch, 8. processing of guanine nucleosides, 9. processing of uracil nucleosides, and 10. C1-processing changes and glutathione synthesis. Finally, we analysed the significant switches. They could be functionally combined yielding seven groups: 1. formate fermentation, 2. mixed acid fermentation, 3. C1 processing changes, 4. C6 nutrients, 5. branched chain amino acids transporters, 6. and 7. These groups consisted of two pairs each with ambiguous functionality and are not discussed here (for details, see *[*Schramm *et al, in prep*.].
The results are sketched in Figure 1. Under aerobic conditions, glycolysis and the TCA cycle are the major producers of energy. The TCA cycle depends heavily on oxygen and is therefore of limited use for anaerobic conditions. To keep the energy production going, glycolysis is up-regulated and similarily the fructose and mannose metabolism (C6 nutrients up-take pathways). The pyruvate metabolism switches the compound flow from the TCA cycle to acids and ethanol production. The formate fermentation degrades the acid formate or expells it into the outside of the cell. Additionally, the TCA normally supplies essential compounds for e.g. producing amino acids.For oxygen deprived conditions, this is taken over by the up-regulated aspartate metabolism. The cell responses to the higher concentration of acids by up-regulating histidine biosynthesis to enable an induced buffering by histidine. Under oxygen rich conditions, oxygen is causing oxidative stress. During anaerobic conditions, oxygen is reduced and therefore the oxidative stress response is down-regulated, i.e. the production of Fe-S and glutathione reduction (Figure 2). NAD switch: even though NAD may be more constitutively produced, up-regulation of quinolate synthetases which are the starting point of the NAD biosynthesis makes sense, as it could be shown that quinolate synthetases become inactive when exposed to oxygen [Ollagnier-De Choudens*, et al.*, 2005] and NAD may be primarily produced via the tryptophan biosynthesis pathway under aerobic conditions. Oxygen limitation limits energy production and therefore reduces producing energy intensive nucleosides. Reduced energy supply may also explain a switching of the branched chain amino acid transporters from ATP-dependend ABC-transporters to sodium-gradient dependend transporters. The C1 processing changes are according to a down-regulated glycin cleavage system and may be due to the fact that the reaction involved reduces NAD^+ to NADH, an oxygen costly reaction [Madigan*, et al.*, 2003]. The production of the one-carbon units, for which the glycine cleavage system is used [Stauffer, 1987], was taken over by serine hydroxymethyltransferase.

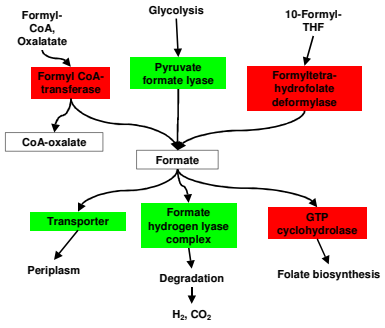


Figure 2. Formate fermentation. Green boxes indicate significant up-regulation (p-value ≤ 0.05) under anaerobic conditions. Red boxes indicate significant down-regulation. Glucose is catabolised into pyruvate. Under anaerobic conditions, pyruvate is degraded to formate which is either expelled, or further degraded into H_2 and CO_2. The reactions for these processes were up-regulated whereas the biosynthesis and degradation of costly compounds were down-regulated (folate and 10-formyl-THF, respectively).

3.2 Comparison to a standard method

To compare the findings of the method described here to a standard method for analysing gene expression data, a t-test was run on the gene expression levels for the corresponding reactions (without any network information). Extracted features were ranked due to the calculated p-value. The first 40 highest ranking reactions were: 1. formate hydrogenlyase complex, 2. FocA formate FNT transporter, 3. pyruvate formate-lyase, 4. aminomethyltransferase, 5. gcv system, 6. 3-methyl-2-oxobutanoate hydroxymethyltransferase, 7. glycine dehydrogenase (decarboxylating), 8. PFL-deactivase, 9. acetaldehyde dehydrogenase, 10. pyruvate kinase, 11. fumarate reductase, 12. enolase, 13. N-acetylmuramyl-L-alanine amidase, 14. formate dehydrogenase, 15. glutamate dehydrogenase (NADP+), 16. mannonate dehydratase, 17.+18. pyruvate formate-lyase activating enzyme (2x), 19. triose phosphate isomerase, 20. glutamyl-tRNA reductase, 21. histidine-phosphate aminotransferase, 22. 2-keto-4-hydroxyglutarate aldolase, 23. 2-keto-3-deoxy-6-phosphogluconate aldolase, 24. oxaloacetate decarboxylase, 25. putative NAD+ kinase, 26. 6-phosphofructokinase-1, 27. mannose-6-phosphate isomerase, 28. Outer Membrane Ferrichrome Transport System, 29. NADH oxidoreductase, 30. isocitrate dehydrogenase kinase, 31. isocitrate dehydrogenase phosphatase, 32. RhtB homoserine Rht Transporter, 33. histidinol-phosphatase, 34. imidazoleglycerol-phosphate dehydratase, 35. Outer Membrane Ferric Enterobactin Transport System, 36. phosphoenolpyruvate carboxylase, 37. tetrahydrodipicolinate succinylase, 38. imidazole glycerol phosphate synthase, 39. 3-hydroxy acid dehydrogenase, and 40. branched chain amino acids ABC transporter. At the top are three reactions involved in fermentation of formate that were also found with our method. Six reactions (10, 13, 15, 25, 29, 32) were not extracted by our method. Five of these reactions were not found due to the network creation method. Unspecific metabolites were deleted resulting in the deletion of reactions that catalyse such unspecific metabolites, such that pyruvate kinase, glutamate dehydrogenase (NADP+), NAD kinase, NADH oxidoreductase and RhtB homoserine Rht transporter were not included into the metabolic network. Putative reactions with not defined metabolites like N-acetyl-anhydromuramyl-L-alanine-amidase, the sixth not found reaction, were also not included into the metabolic network and could therefore not be found. With this caculated list from the standard method, we could not get any reactions for the iron processing response. Furthermore, the interesting histidine pathway was entirely found by our connected edge method. In contrast, with the standard method we found four out of ten reactions which are scattered in the list (21, 33, 34, 38) making it rather difficult to infer a combined regulation of the defined histidine pathway.

4 Conclusions

The methods described here facilitate the extraction of interesting and complex sub-graphs within a metabolic network by applying image-processing methods onto gene expression data. It suits well for less complex organisms like *E. coli*, for which the metabolic network is well established and reaction levels can be better estimated from the gene expression levels. In our case study several interesting and essential sub-graphs with differential expression patterns for *E. coli* when exposed to oxygen deprived conditions were identified like the fermentation of formate, processing of C6 nutrients, biosynthesis of histidine and iron metabolism. Thus a huge variety of anaerobic responses were discovered ranging from fermentation to energy and iron metabolism to acidic buffering. This covered not only direct regulations but also patterns originating from more complex environmental influences following the adaptation to oxygen deprivation like the response to excreted acids and thus the change in pH. Nevertheless, essential sub-graphs are not detected isolated but might interfere with related or connected pathways depending on the metabolites. The cluster containing histidine biosynthesis consisted also of parts of the aspartate and glutamine metabolism. This is due to the unspecific hub-like character of some metabolites connecting a huge variety of pathways. However, to give the found sub-graphs and reaction chains functional meaning was rather tedious and time consuming. We had to scan the appropriate literature and extract the specific information in a very detailed and long lasting procedure. We see an automated processing of this as a major task for the future.

Acknowledgments

We thank EcoCyc, Covert and his co-workers, and the ASAP team for making their data online available. The work was funded by the German National Genome Research Network (NGFN 01 GR 0450) and the Deutsche Forschungsgemeinschaft (Optimization-based control of chemical processes BO 864/10).

References

J.M. Berg, Tymoczko J.L., Stryer L.: *Biochemistry.* Fifth Edition edn. New York: W. H. Freeman; 2002.

C. E. Bonferroni: *Il Calcolo Delle Assicurazioni Su Gruppi Di Test.* In *Studi in Onore Del Professore Salvatore Ortu Carboni.* Rome, Italy; 1935a: 13-60

C. E. Bonferroni: *Il Calcolo Delle Assicurazioni Su Gruppi Di Teste.* In *Studi in Onore Del Professore Salvatore Ortu Carboni.* Rome; 1935b: 13-60

M. W. Covert, Knight E. M., Reed J. L., Herrgard M. J., Palsson B. O. *Integrating High-Throughput and Computational Data Elucidates Bacterial Networks***.** Nature, 429:92-96, 2004.

T. S. Gardner, Di Bernardo D., Lorenz D., Collins J. J. *Inferring Genetic Networks and Identifying Compound Mode of Action Via Expression Profiling***.** Science, 301:102-105, 2003.

A. P. Gasch, Spellman P. T., Kao C. M., Carmel-Harel O., Eisen M. B., Storz G., Botstein D., Brown P. O. *Genomic Expression Programs in the Response of Yeast Cells to Environmental Changes***.** Mol Biol Cell, 11:4241-4257, 2000.

D. Hanisch, Zien A., Zimmer R., Lengauer T. *Co-Clustering of Biological Networks and Gene Expression Data***.** Bioinformatics, 18 Suppl 1:S145-154, 2002.

W. Huber, Von Heydebreck A., Sultmann H., Poustka A., Vingron M. *Variance Stabilization Applied to Microarray Data Calibration and to the Quantification of Differential Expression***.** Bioinformatics, 18 Suppl 1:S96-104, 2002.

T. Ideker, Ozier O., Schwikowski B., Siegel A. F. *Discovering Regulatory and Signalling Circuits in Molecular Interaction Networks***.** Bioinformatics, 18 Suppl 1:S233-240, 2002.

P. D. Karp, Riley M., Paley S. M., Pellegrini-Toole A. *The Metacyc Database***.** Nucleic Acids Res, 30:59-61, 2002.

I. M. Keseler, Collado-Vides J., Gama-Castro S., Ingraham J., Paley S., Paulsen I. T., Peralta-Gil M., Karp P. D. *Ecocyc: A Comprehensive Database Resource for Escherichia Coli***.** Nucleic Acids Res, 33:D334-337, 2005.

R. König, Eils R. *Gene Expression Analysis on Biochemical Networks Using the Potts Spin Model***.** Bioinformatics, 20:1500-1505, 2004.

R. König, Schramm G., Oswald M., Seitz H., Sager S., Zapatka M., Reinelt G., Eils R. *Discovering Functional Gene Expression Patterns in the Metabolic Network of Escherichia Coli with Wavelets Transforms***.** BMC Bioinformatics, 7:119, 2006.

T. M. Madigan, Martinko J. M., Parker J.: *Biology of Microorganisms.* 10th edn: Prentice Hall; 2003.

T. Manoli, Gretz N., Grone H. J., Kenzelmann M., Eils R., Brors B. *Group Testing for Pathway Analysis Improves Comparability of Different Microarray Data Sets***.** Bioinformatics, 2006.

F.C. Neidhardt: *Escherichia Coli and Salmonella: Cellular and Molecular Biology.* Washington D.C.: American Society for Microbiology; 1996.

S. Ollagnier-De Choudens, Loiseau L., Sanakis Y., Barras F., Fontecave M. *Quinolinate Synthetase, an Iron-Sulfur Enzyme in Nad Biosynthesis***.** FEBS Lett, 579:3737-3743, 2005.

M. Ruschhaupt, Huber W., Poustka A., Mansmann U. *A Compendium to Ensure Computational Reproducibility in High-Dimensional Classification Tasks***.** Stat Appl Genetics Mol Biol, 3:37, 2004.

G. Schramm, Eils R., König R. *E. Coli's Crucial Switches, Pathways and Clusters of Gene Expression During Oxygen Deprivation***.** in preparation,

P. T. Spellman, Sherlock G., Zhang M. Q., Iyer V. R., Anders K., Eisen M. B., Brown P. O., Botstein D., Futcher B. *Comprehensive Identification of Cell Cycle-Regulated Genes of the Yeast Saccharomyces Cerevisiae by Microarray Hybridization***.** Mol Biol Cell, 9:3273-3297, 1998.

G. V. Stauffer: *Biosynthesis of Serine and Glycine.* In *Escherichia Coli and Salmonella Typhimurium Cellular and Molecular Biology. Volume* 1. Edited by F. C. Neidhardt: American Society for Microbiology; 1987: 412-418

G. Stephanopoulos, Hwang D., Schmitt W. A., Misra J. *Mapping Physiological States from Microarray Expression Measurements***.** Bioinformatics, 18:1054-1063, 2002.

S. Theodoridis, Koutroumbas K.: *Pattern Recognition.* London: Academic Press; 1998.

O. Thuerigen, Schneeweiss A., Toedt G., Warnat P., Hahn M., Kramer H., Brors B., Rudlowski C., Benner A., Schuetz F., Tews B., Eils R., Sinn H. P., Sohn C., Lichter P. *Gene Expression Signature Predicting Pathologic Complete Response with Gemcitabine, Epirubicin, and Docetaxel in Primary Breast Cancer***.** J Clin Oncol, 24:1839-1845, 2006.

V. G. Tusher, Tibshirani R., Chu G. *Significance Analysis of Microarrays Applied to the Ionizing Radiation Response***.** Proc Natl Acad Sci U S A, 98:5116-5121, 2001.

P. Uetz, Giot L., Cagney G., Mansfield T. A., Judson R. S., Knight J. R., Lockshon D., Narayan V., Srinivasan M., Pochart P., Qureshi-Emili A., Li Y., Godwin B., Conover D., Kalbfleisch T., Vijayadamodar G., Yang M., Johnston M., Fields S., Rothberg J. M. *A Comprehensive Analysis of Protein-Protein Interactions in Saccharomyces Cerevisiae***.** Nature, 403:623-627, 2000.

L. J. Van 'T Veer, Dai H., Van De Vijver M. J., He Y. D., Hart A. A., Mao M., Peterse H. L., Van Der Kooy K., Marton M. J., Witteveen A. T., Schreiber G. J., Kerkhoven R. M., Roberts C., Linsley P. S., Bernards R., Friend S. H. *Gene Expression Profiling Predicts Clinical Outcome of Breast Cancer***.** Nature, 415:530-536, 2002.

A. Zien, Kuffner R., Zimmer R., Lengauer T. *Analysis of Gene Expression Data with Pathway Scores***.** Proc Int Conf Intell Syst Mol Biol, 8:407-417, 2000.

Pairwise Naive Bayes Classifier

Jan-Nikolas Sulzmann
Technische Universität Darmstadt
D-64289, Darmstadt, Germany
sulzmann@ke.informatik.tu-darmstadt.de

Abstract

Class binarizations are effective methods that break multi-class problem down into several 2-class or binary problems to improve weak learners. This paper analyzes which effects these methods have if we choose a Naive Bayes learner for the base classifier. We consider the known unordered and pairwise class binarizations and propose an alternative approach for a pairwise calculation of a modified Naive Bayes classifier.

1 Introduction

The Naive Bayes classifier (NB) is a Bayesian learner which outperforms more sophisticated learning methods like Neural Networks, Nearest Neighbour, or Decision Tree Learning in many fields of application. NB is widely deployed because of its simplicity, versatility, and efficiency. We seek to increase its performance by combining it with class binarization methods which are a way to enhance a weak learner. To this end we consider on the one hand the well known unordered and pairwise classbinarization and on the other hand alternative methods for a pairwise calculation of NB.

We developed a couple of alternative methods which all are based on the same probabilistic approach. This approach uses not only the probabilities of classes but also the probabilities of pairs of classes which can be computed in two different ways. The first one which we call regular estimates the probabilities much like NB, the second one estimates not only the probabilities of classes but also the probability of pairs of classes.

This paper summarizes the results of [Sulzmann, 2006] and is assembled as follows. In the section *"Fundamentals & Notations"* we introduce the required notations and give a short survey of the basics of a Naive Bayes classifier. After this we describe the used class binarizations and several decoding methods.

The second section *"Pairwise Naive Bayes Classifier"* consist of three subsections. In the first two subsections we draw some conclusions about class binarizations with a Naive Bayes classifier. We show that contrary to expectations the unordered class binarizations is not equal to a Naive Bayes classifer. Afterwards we prove that the pairwise class binarization with decoding methods we described in the previous sections is equivalent to a Naive Bayes classifier. In the third subsection we introduce an alternative approach for a pairwise calculation of a Naive Bayes classifier. After this we describe four methods that implement this approach differently. All of them can be computed in two ways which we introduce thereafter.

In the third section *Experiments* we describe the experiments we have made. They consist of a comparison of our own methods, class binarizations and the Naive Bayes classifier which we test with several options (e.g. discretization and computation techniques). After this we analyse the results of our experiments which are summarized in two tables in the appendix.

In the last section *Conclusion* we resume the conclusions we have drawn about class binarizations with a Naive Bayes classifier and about our own methods.

2 Fundamentals & Notations

This section prides a short survey of the fundamentals and notations which are relevant for this paper.

2.1 Notation

A: an attribute, a set of attribute values

A_i: the ith attribute, $i \in \{1, ..., n\}$

a_i: an abitrary value of attribute A_i

v_i: the quantity of different attribute values of A_i

$D = (a_1, ..., a_n)$: an example described by attribute values

$c_i \in \{c_1, c_2, ..., c_m\}$: a class (a probabilistic event)

$c_{ij}, i \neq j$: a class pair, (the probabilistic event $c_{ij} = c_i \cup c_j$)

Pr: a probability

$\widehat{\Pr}$: a estimated probability

$\Pr(c_i|D)$: the probability that the example is of class c_i

$\Pr(c_i|D, c_{ij})$: the probability that the example is of class c_i under observation of class pair c_{ij}

t: the quantity of training examples

t_{c_i}: the quantity of training examples who belong to class c_i

$t_{c_j}^{a_i}$: the quantity of training examples who belong to class c_j and whose j^{th} attribute value matches a_i

2.2 Naive Bayes Classifier

The Naive Bayes classifier is a Bayesian learning method and therefore based on the theorem of Bayes:

$$\Pr(A|B) = \frac{\Pr(B|A) \cdot \Pr(A)}{\Pr(B)}$$

The goal of NB is to predict from a training set of classified examples the class of an example $D = (a_1, , a_n)$, where a_i is the value of the i^{th} attribute. The error can be minimized by selecting $argmax_{c_i} \Pr(c_i|D)$, where

$c_i, ..., c_m$ are the m classes. Therefor we need estimates $\widehat{\Pr}(c_i|D)$ of $\Pr(c_i|D)$,$\forall i$. One can adapt the theorem of Bayes to solve this problem:

$$\Pr(c_i|D) = \frac{\Pr(D|c_i) \cdot \Pr(c_i)}{\Pr(D)},$$

where $\Pr(D) = \sum_j (\Pr(D|c_j) \cdot \Pr(c_j))$

With this approach we gain the following basic version of a Bayesian learner:

$$\begin{aligned} c_B &= \arg\max_{c_i} \frac{\Pr(D|c_i) \cdot \Pr(c_i)}{\Pr(D)} \\ &= \arg\max_{c_i} \Pr(D|c_i) \cdot \Pr(c_i) \\ &= \arg\max_{c_i} \Pr(a_1, a_2, ..., a_n|c_i) \cdot \Pr(c_i) \end{aligned}$$

If we make the naive assumptions that the attributes are independent the classifier is called *naive* and the probability $\Pr(D|c_i)$ can be calculated as follows:

$$\begin{aligned} \Pr(D|c_i) &= \Pr(a_1, a_2, ..., a_n|c_i) \\ &= \prod_{j=1}^{n} \Pr(a_j|c_i) \end{aligned}$$

With this formula we obtain the basic version of NB:

$$c_{NB} = \arg\max_{c_i} \Pr(c_i) \cdot \prod_{j=n}^{n} \Pr(a_j|c_i)$$

The probabilities $\Pr(c_i)$ and $\Pr(a_j|c_i)$ can be estimated with the relative frequencies of training examples of class c_i in the training set and accordingly with the relative frequencies of trainining examples of this class whose attribute value of the corresponding attribute has the value a_j.

If one of the latter relative frequencies equals zero for one of the classes, then the total predicted probability of this class equals also zero. This problem can be solved by assuming that the relative frequencies have prior distributions. A well known representative of this approach is the Laplace estimate. It presumes that each attribute value occurs one more time than it appears in the training set.

The probabilities $\Pr(a_j|c_i)$ and $\Pr(c_i)$ can be estimated (with the Laplace estimate) as follows:

$$\widehat{\Pr}(a_i|c_j) = \frac{t_{c_j}^{a_i} + 1}{t_{c_j} + v_i} \qquad \widehat{\Pr}(c_i) = \frac{t_{c_i}}{t}$$

2.3 Class Binarization

Class binarization techniques [Fürnkranz, 2002; 2003] solve multi-class problems by turning them into a set of binary problems. This enables machine learning methods which are inherently designed for binary problems (e.g. perceptrons, support vector machines (SVM) etc.) to solve multi-class problems.

Definition 2.1 (class binarization, decoding, base learner) *A* class binarization *is a mapping of a multi-class learning problem to several two-class learning problems in a way that allows a sensible* decoding *of the prediction, i.e., it allows the derivation of a prediction for the multi-class problem from the predictions of the set of two-class classifiers. The learning algorithm used for solving the two-class problems is called the* base learner .

The most popular class binarization technique is the *unordered or one-against-all class binarization* (abbr.: 1vsAll), where one takes each class in turn and learns binary concepts that discriminate this class from all other classes. It has been independently proposed for rule learning, neural networks, and SVM.

Definition 2.2 (unordered/one-against-all class binarization) *The* unordered class binarization *transforms a m-class problem into m binary problems. These are constructed by using the examples of class i as the positive examples and the examples of classes j ($j = 1, ..., c, j \neq i$) as the negative examples.*

A more complex binarization technique is the *pairwise or round robin class binarization*. The basic idea is quite simple, namely to learn one classifier for each pair of classes.

Definition 2.3 (round robin/pairwise class binarization) *The round robin or pairwise class binarization transforms a m-class problem into $m(m-1)/2$ two-class problems $< i, j >$, one for each set of classes $\{i, j\}$,$i = 1, ..., m-1$,$j = i+1, ..., m$ The binary classifier for problem $< i, j >$ is trained with examples of classes i and j, whereas examples of classes $k \neq i, j$ are ignored for this problem.*

A crucial point of this technique is how to decode the predictions of the pairwise classifiers to a final prediction. We use Voting, Weighted Voting and methods which are based on the Bradley-Terry-modell for decoding the predictions [Wu *et al.*, 2004].

Voting (abbr.: V) is a simple technique. When we classify a new example, each of the learned base classifiers determines to which of its two classes the example is more likely to belong to. The winner is assigned a point, and in the end, the algorithm will predict the class that has accumulated the most points.

$$\arg\max_{c_i} \sum_{j=i} [\Pr(c_i|D, c_{ij})], \quad [x] = \begin{cases} 1, \text{ if x } \geq 0.5 \\ 0, \text{ else.} \end{cases}$$

For *Weighted Voting* (abbr.: WV) we need additionally the confidence of the base classifiers in its predictions. In contrast to Voting we do not add up points but weighted votes which correspond to the confidence measures.

$$\arg\max_{c_i} \sum_{j=i} \Pr(c_i|D, c_{ij})$$

The last two methods are based upon the *Bradley-Terry modell* which consists of the assumption that the following holds:

$$\Pr(c_i|D, c_{ij}) = \frac{\Pr(c_i|D)}{\Pr(c_i|D) + \Pr(c_j|D)}$$

The methods that we will introduce try to estimate with this assumption and the estimates of $\Pr(c_i|D, c_{ij})$ the probability $\Pr(c_i|D)$ for each class c_i. They attempt to minimize the distance between the estimation $\widehat{\Pr}(c_i|D, c_{ij})$ and $\overline{\Pr}(c_i|D, c_{ij})$ which can be calculated as follows:

$$\overline{\Pr}(c_i|D, c_{ij}) = \frac{\widehat{\Pr}(c_i|D)}{\widehat{\Pr}(c_i|D) + \widehat{\Pr}(c_j|D)}$$

The methods differ in their approach of minimizing the distance betweens the estimates and calculations. The first

Algorithm 2.1 Method of Hastie & Tibshirani

Input: $\widehat{Pr}(c_i|D,c_{ij})$,$t_{c_{ij}}$, $i,j \in \{1,...,m\}$, $j \neq i$
Output: $\widehat{Pr}(c_i|D)$, $i = 1,...,m$

1: Start with some initial $\widehat{Pr}(c_i|D)$,$\forall i$ and corresponding $\widehat{Pr}(c_i|D,c_{ij})$
2: **repeat** $\{i = 1,...,m,1,...\}$
3: $\quad \alpha = \frac{\sum_{j\neq i} t_{c_{ij}} \cdot \widehat{Pr}(c_i|D,c_{ij})}{\sum_{j\neq i} t_{c_{ij}} \cdot \overline{Pr}(c_i|D,c_{ij})}$
4: $\quad \overline{Pr}(c_i|D,c_{ij}) \leftarrow \frac{\alpha \cdot \overline{Pr}(c_i|D,c_{ij})}{\alpha \cdot \overline{Pr}(c_i|D,c_{ij}) + \widehat{Pr}(c_i|D,c_{ij})}$
5: $\quad \overline{Pr}(c_j|D,c_{ij}) = 1 - \overline{Pr}(c_i|D,c_{ij})$
6: $\quad \widehat{Pr}(c_i|D) = \alpha \cdot \widehat{Pr}(c_i|D)$
7: $\quad \widehat{Pr}(c_i|D) = \frac{\widehat{Pr}(c_i|D)}{\sum_j \widehat{Pr}(c_j|D)}$ (optional)
8: **until** m consecutive α are all close to ones
9: **return** $\widehat{Pr}(c_i|D) = \frac{\widehat{Pr}(c_i|D)}{\sum_j \widehat{Pr}(c_j|D)}$

method which was proposed by [Price *et al.*, 1994] (referred as PKPD) is based on a calculation formula. The second one that was suggested by [Hastie and Tibshirani, 1997] (referred a HT) specifies an algorithm which solves this minimization problem.

[Price *et al.*, 1994] consider that

$$\left(\sum_{j\neq i} \Pr\left(c_{ij}|D\right)\right) - (m-2)\cdot \Pr\left(c_i|D\right) = \sum_{j=1}^{m} \Pr\left(c_j|D\right)$$

holds. If we adapt this equation and

$$\Pr\left(c_{ij}|D\right) = \Pr\left(c_i|D\right) + \Pr\left(c_j|D\right)$$

to the estimated probabilities we obtain the following calculation formula:

$$\widehat{\Pr}\left(c_i|D\right)_{PKPD} = \frac{1}{\sum_{j\neq i} \frac{1}{\widehat{\Pr}(c_i|D,c_{ij})} - m + 2}$$

The approach of [Hastie and Tibshirani, 1997] tries to minimize the Kullback-Leibler distance $l\left(p\right)$ between $\widehat{\Pr}\left(c_i|D,c_{ij}\right)$ and $\overline{\Pr}\left(c_i|D,c_{ij}\right)$.

$$\begin{aligned} l\left(p\right) &= \sum_{i<j} t_{c_{ij}} \cdot \left(\widehat{\Pr}\left(c_i|D,c_{ij}\right) \cdot \log \frac{\widehat{\Pr}\left(c_i|D,c_{ij}\right)}{\overline{\Pr}\left(c_i|D,c_{ij}\right)} \right. \\ &+ \left. \widehat{\Pr}\left(c_j|D,c_{ij}\right) \cdot \log \frac{\widehat{\Pr}\left(c_i|D,c_{ij}\right)}{\overline{\Pr}\left(c_i|D,c_{ij}\right)} \right), \end{aligned}$$

where $t_{c_{ij}} = t_{c_i} + t_{c_j}$ is the sum of the training examples of the classes c_i + c_j.

To this end [Hastie and Tibshirani, 1997] propose to find estimated probabilities $\widehat{\Pr}(c_i|D)$ for each class which satisfy the following conditions:

$$\sum_{j\neq i} t_{c_{ij}} \cdot \widehat{\Pr}\left(c_i|D,c_{ij}\right) = \sum_{j\neq i} t_{c_{ij}} \cdot \overline{\Pr}\left(c_i|D,c_{ij}\right)$$

$$\sum_{i=1}^{m} \widehat{\Pr}\left(c_i|D\right) = 1$$

$$\widehat{\Pr}\left(c_i|D\right) > 0, i = 1,...,m$$

This problem can be solved by algorithm 2.1

3 Pairwise Naive Bayes Classifier

As aforementioned we seek to improve the performance of NB by combining it with class binarization methods. Therefore we consider the unordered and pairwise class binarization and alternative methods.

3.1 Unordered Class Binarization

The structure of the unordered class binarization with NB as its base classifier is very similar the structure to NB. So one might think they compute the same predictions for a given example. Contrary to expectations these two methods calculate different probabilitiy estimates and if applicable different predictions.

The unordered class binarization splits a m-class problem in m binary problems that consists of discriminating one class from all other. For class c_i the other classes are handled as one class $\overline{c_i}$ and their training example are thrown together. This approach does not change any relative frequencies of class c_i. Therefore the probabilities $\Pr(D|c_i)$ and $\Pr(c_i)$ remain unchanged.

The absolute frequencies of $\overline{c_i}$ have to be calculated:

$$t^a_{\overline{c_i}} = \sum_{j\neq i} t^a_{c_j} \qquad t_{\overline{c_i}} = \sum_{j\neq i} t_{c_j}$$

With the aid of this quantities and the Laplace estimate the required probabilities for $\overline{c_i}$ can be estimated as follows:

$$\widehat{\Pr}(a_k|c_i) = \frac{t^{a_k}_{\overline{c_i}} + 1}{t_{\overline{c_i}} + v_k} = \frac{\left(\sum_{j\neq i} t^{a_k}_{c_j}\right) + 1}{\left(\sum_{j\neq i} t_{c_j}\right) + v_k}$$

and

$$\widehat{\Pr}(\overline{c_i}) = \frac{t_{\overline{c_i}}}{t} = \frac{\sum_{j\neq i} t_{c_j}}{t} = \sum_{j\neq i} \widehat{\Pr}(c_j) = 1 - \widehat{\Pr}(c_i)$$

Now we can estimate $\Pr(D|\overline{c_i})$ as follows:

$$\widehat{\Pr}(D|\overline{c_i})_{UK} = \prod_{k=1}^{n} \widehat{\Pr}(a_k|\overline{c_i})_{UK} = \prod_{k=1}^{n} \frac{\left(\sum_{j\neq i} t^{a_k}_{c_j}\right) + 1}{\left(\sum_{j\neq i} t_{c_j}\right) + v_k}$$

If we consider the abovementioned estimation we can clearly see that the following holds:

$$\widehat{\Pr}(D|\overline{c_i})_{UK} \cdot \widehat{\Pr}(\overline{c_i}) \neq \sum_{j\neq i} \widehat{\Pr}(D|c_j)_{NB} \cdot \widehat{\Pr}(c_j)$$

Therefore NB and the unordered class binarization compute different estimations for $\Pr(c_i|D)$:

$$\begin{aligned} & \widehat{\Pr}(c_i|D)_{UK} \\ = & \frac{\widehat{\Pr}(D|c_i) \cdot \widehat{\Pr}(c_i)}{\widehat{\Pr}(D|c_i) \cdot \widehat{\Pr}(c_i) + \widehat{\Pr}(D|\overline{c_i})_{UK} \cdot \widehat{\Pr}(\overline{c_i})} \\ \neq & \frac{\widehat{\Pr}(D|c_i) \cdot \widehat{\Pr}(c_i)}{\widehat{\Pr}(D|c_i) \cdot \widehat{\Pr}(c_i) + \sum_{j\neq i} \widehat{\Pr}(D|c_j)_{NB} \cdot \widehat{\Pr}(c_j)} \\ = & \widehat{\Pr}(c_i|D)_{NB} \end{aligned}$$

3.2 Pairwise Class Binarization

Contrary to the expectations the pairwise class binarization with NB as its base classifier is equivalent to NB. That means they predict always the same class for an example. Different predictions can be traced back to imprecise implementations of the class binarization shemes.

Before we can give a proof of the aforementioned, we have to show some relations between the probabilities of both classifiers. The base classifier of class pair c_{ij} estimates the probabilities $\Pr(c_i|D, c_{ij})$ and $\Pr(c_j|D, c_{ij})$ that can be calculated as follows:

$$\Pr(c_i|D, c_{ij}) = \frac{\Pr(D|c_i) \cdot \Pr(c_i)}{\Pr(D|c_i) \cdot \Pr(c_i) + \Pr(D|c_j) \cdot \Pr(c_i)}$$
$$\Pr(c_j|D, c_{ij}) = 1 - \Pr(c_i|D, c_{ij})$$

As a reminder the probability $\Pr(c_j|D)$ can be calculated as follows:

$$\Pr(c_i|D) = \frac{\Pr(D|c_i) \cdot \Pr(c_i)}{\sum_j (\Pr(D|c_j) \cdot \Pr(c_j))}$$

If we compare both calculations, we are able to see that the probabilities differ only in their normalization factors. This leads to the following lemma:

Lemma 3.1 *For any two mutual different classes c_i and c_j holds*

$$\frac{\Pr(c_i|D, c_{ij})}{\Pr(c_j|D, c_{ij})} = \frac{\Pr(c_i|D)}{\Pr(c_j|D)}$$

PROOF.

$$\frac{\Pr(c_i|D, c_{ij})}{\Pr(c_j|D, c_{ij})} = \frac{\frac{\Pr(c_i|D)}{\Pr(c_i|D)+\Pr(c_j|D)}}{\frac{\Pr(c_j|D)}{\Pr(c_i|D)+\Pr(c_j|D)}} = \frac{\Pr(c_i|D)}{\Pr(c_j|D)} \quad \square$$

This relation gives a hint to some essential transitive correlation between the probabilities of class pairs.

Lemma 3.2 *For any mutual different classes c_i, c_j and c_k holds:*

(a) The following inequalities are equivalent

$$\begin{aligned} & \Pr(c_i|D) < \Pr(c_j|D) && (1)\\ \Leftrightarrow\quad & \Pr(c_i|D, c_{ij}) < \Pr(c_j|D, c_{ij}) && (2)\\ \Leftrightarrow\quad & \Pr(c_i|D, c_{ik}) < \Pr(c_j|D, c_{jk}) && (3) \end{aligned}$$

(b)

$$\begin{aligned} & \Pr(c_i|D, c_{ik}) < \Pr(c_k|D, c_{ik})\\ \wedge\quad & \Pr(c_k|D, c_{jk}) < \Pr(c_j|D, c_{jk})\\ \Rightarrow\quad & \Pr(c_i|D, c_{ij}) < \Pr(c_j|D, c_{ij}) \end{aligned}$$

PROOF. (a)

(1) $\Leftrightarrow$ (2)

$$\begin{aligned} & \Pr(c_i|D) < \Pr(c_j|D)\\ \Leftrightarrow\quad & \tfrac{\Pr(c_i|D)}{\Pr(c_i|D)+\Pr(c_j|D)} < \tfrac{\Pr(c_i|D)}{\Pr(c_i|D)+\Pr(c_j|D)}\\ \Leftrightarrow\quad & \Pr(c_i|D, c_{ij}) < \Pr(c_j|D, c_{ij}) \end{aligned}$$

(1) $\Leftrightarrow$ (3)

$$\begin{aligned} & \Pr(c_i|D) < \Pr(c_j|D)\\ \Leftrightarrow\quad & \Pr(c_i|D) + \Pr(c_k|D) < \Pr(c_j|D) + \Pr(c_k|D)\\ \Leftrightarrow\quad & \tfrac{\Pr(c_k|D)}{\Pr(c_i|D)+\Pr(c_k|D)} > \tfrac{\Pr(c_k|D)}{\Pr(c_j|D)+\Pr(c_k|D)}\\ \Leftrightarrow\quad & \tfrac{\Pr(c_i|D)}{\Pr(c_i|D)+\Pr(c_k|D)} < \tfrac{\Pr(c_j|D)}{\Pr(c_j|D)+\Pr(c_k|D)}\\ \Leftrightarrow\quad & \Pr(c_i|D, c_{ik}) < \Pr(c_j|D, c_{jk}) \end{aligned}$$

(b)

$$\begin{aligned} & \begin{pmatrix} \Pr(c_i|D, c_{ik}) < \Pr(c_k|D, c_{ik}) \\ \wedge \Pr(c_k|D, c_{jk}) < \Pr(c_j|D, c_{jk}) \end{pmatrix}\\ \Leftrightarrow\quad & \begin{pmatrix} \Pr(c_i|D) < \Pr(c_k|D) \\ \wedge \Pr(c_k) < \Pr(c_j|D) \end{pmatrix}\\ \Leftrightarrow\quad & \Pr(c_i|D) < \Pr(c_j|D) \end{aligned} \quad \square$$

These transitive relations hold analogous for the equality of probabilities.

Corollary 3.3 *For any mutual different classes c_i, c_j and c_k holds:*

(a) The following inequalities are equivalent

$$\begin{aligned} & \Pr(c_i|D) = \Pr(c_j|D) && (1)\\ \Leftrightarrow\quad & \Pr(c_i|D, c_{ij}) = \Pr(c_j|D, c_{ij}) && (2)\\ \Leftrightarrow\quad & \Pr(c_i|D, c_{ik}) = \Pr(c_j|D, c_{jk}) && (3) \end{aligned}$$

(b)

$$\begin{aligned} & \Pr(c_i|D, c_{ik}) = \Pr(c_k|D, c_{ik})\\ \wedge\quad & \Pr(c_k|D, c_{jk}) = \Pr(c_j|D, c_{jk})\\ \Rightarrow\quad & \Pr(c_i|D, c_{ij}) = \Pr(c_j|D, c_{ij}) \end{aligned}$$

PROOF. Follows directly from Lemma (3.2) or can be analogous proven. $\square$

Now we have all utilities that we need for our latter equivalency proofs. Let us have a look at the basic structure of the Round Robin class binarization whose class pairs are evaluated by voting methods.

$$\arg\max_{c_i} \sum_{j \neq i} vote(c_i, c_j), \qquad (1)$$

where vote is a function which determines depending on the voting method how class c_i is rated under the class pair c_{ij}. The voting methods Voting and Weighted Voting can be written as follows:

$$vote_V(c_i, c_j) = [\Pr(c_i|D, c_{ij})]$$
$$vote_{WV}(c_i, c_j) = \Pr(c_i|D, c_{ij})$$

Comparing these functions with each other and NB we draw two conclusions. First both functions are equivalent in respect to the voting result. That means the predictions of the Round Robin class binarization with one of these functions are the same. Second the Round Robin class binarization with these voting methods is equivalent to NB. Before we proof this conclusion we have to introduce some minor facts.

Lemma 3.4 *For any mutual different classes c_i and c_k holds:*

(a)

$$\begin{aligned} & \Pr(c_i|D, c_{ij}) < \Pr(c_j|D, c_{ij})\\ \Leftrightarrow\quad & [\Pr(c_i|D, c_{ij})] < [\Pr(c_j|D, c_{ij})] \end{aligned}$$

(b)

$$\begin{aligned} & \Pr(c_i|D, c_{ij}) < \Pr(c_j|D, c_{ij})\\ \Rightarrow\quad & [\Pr(c_i|D, c_{ik})] \leq [\Pr(c_j|D, c_{jk})] \end{aligned}$$

PROOF. (a)

$$\begin{aligned} \Leftrightarrow\quad & \Pr(c_i|D, c_{ij}) < \frac{1}{2} < \Pr(c_j|D, c_{ij})\\ \Leftrightarrow\quad & [\Pr(c_i|D, c_{ij})] < [\Pr(c_j|D, c_{ij})] \end{aligned}$$

(b)

$$\begin{aligned} & \Pr(c_i|D, c_{ij}) < \Pr(c_j|D, c_{ij})\\ \Leftrightarrow\quad & \Pr(c_i|D, c_{ik}) < \Pr(c_j|D, c_{jk})\\ \Rightarrow\quad & [\Pr(c_i|D, c_{ik})] < [\Pr(c_j|D, c_{jk})] \end{aligned} \quad \square$$

This leads directly to the following corollary:

Corollary 3.5 *For any mutual different classes c_i and c_j holds:*

$$\begin{aligned} & \Pr(c_i|D,c_{ij}) = \Pr(c_j|D,c_{ij}) \\ \Leftrightarrow \quad & [\Pr(c_i|D,c_{ij})] = [\Pr(c_j|D,c_{ij})] \\ \Leftrightarrow \quad & [\Pr(c_i|D,c_{ik})] = [\Pr(c_j|D,c_{jk})] \end{aligned}$$

Hence we draw the conclusion that the rankings of NB and of the Round Robin class binarization with voting methods are related as follows:

Lemma 3.6 *For any mutual different classes c_i and c_j holds:*

(a)

$$\begin{aligned} & \Pr(c_i|D) \leq \Pr(c_j|D) \\ \Leftrightarrow \quad & \sum_{k\neq i} \Pr(c_i|D,c_{ik}) \leq \sum_{k\neq j} \Pr(c_j|D,c_{jk}) \end{aligned}$$

(b)

$$\begin{aligned} & \Pr(c_i|D) \leq \Pr(c_j|D) \\ \Leftrightarrow \quad & \sum_{k\neq i} [\Pr(c_i|D,c_{ik})] \leq \sum_{k\neq j} [\Pr(c_j|D,c_{jk})] \end{aligned}$$

PROOF.

(a) "⇒":

For any class c_k with $c_i \neq c_k \neq c_j$ holds

$$\begin{aligned} & \Pr(c_i|D,c_{ij}) < \Pr(c_j|D,c_{ij}) \\ \Rightarrow \quad & \Pr(c_i|D,c_{ik}) < \Pr(c_j|D,c_{jk}) \end{aligned}$$

due to Lemma 3.2 and

$$\begin{aligned} & \Pr(c_i|D,c_{ij}) = \Pr(c_j|D,c_{ij}) \\ \Rightarrow \quad & \Pr(c_i|D,c_{ik}) = \Pr(c_j|D,c_{jk}). \end{aligned}$$

due to Lemma Korollar 3.3.

Recapitulating we obtain the following implication.

$$\begin{aligned} & \Pr(c_i|D) \leq \Pr(c_j|D) \\ \Rightarrow \quad & \sum_{k\neq i} \Pr(c_i|D,c_{ik}) \leq \sum_{k\neq j} \Pr(c_j|D,c_{jk}) \end{aligned}$$

"⇐": Analogous to "⇒" holds

$$\begin{aligned} & \Pr(c_i|D,c_{ij}) > \Pr(c_j|D,c_{ij}) \\ \Rightarrow \quad & \sum_{k\neq i} \Pr(c_i|D,c_{ik}) > \sum_{k\neq j} \Pr(c_j|D,c_{jk}) \end{aligned}$$

due to Lemma 3.2. Via contraposition we obtain the following:

$$\begin{aligned} & \sum_{k\neq i} \Pr(c_i|D,c_{ik}) \leq \sum_{k\neq j} \Pr(c_j|D,c_{jk}) \\ \Leftrightarrow \quad & \neg \left(\sum_{k\neq i} \Pr(c_i|D,c_{ik}) > \sum_{k\neq j} \Pr(c_j|D,c_{jk}) \right) \\ \Rightarrow \quad & \neg \left(\Pr(c_i|D,c_{ij}) > \Pr(c_j|D,c_{ij}) \right) \\ \Leftrightarrow \quad & \Pr(c_i|D,c_{ij}) \leq \Pr(c_j|D,c_{ij}) \end{aligned}$$

□

(b) Can analagous be proven with the aid of Lemma 3.4 and Korollar 3.5.

According to Lemma 3.6 the rankings of NB and the Round Robin class binarization with Voting or Weighted Voting are equivalent. Hence we can draw the following conclusion:

Theorem 3.7 *The Round Robin class binarization with NB as its base classifier and the voting methods Voting or Weighted Voting predicts the same ranking and classification as a Naive Bayes classifier:*

$$\begin{aligned} & \arg\max_{c_i} \Pr(c_i|D) \\ = \quad & \arg\max_{c_i} \sum_{j\neq i} \Pr(c_i|D,c_{ij}) \\ = \quad & \arg\max_{c_i} \sum_{j\neq i} [\Pr(c_i|D,c_{ij})] \end{aligned}$$

PROOF. According to Lemma 3.6 the rankings of these methods are equivalent. Therefore their predictions are also equal. □

This theoretical equivalency exists also in practice if the estimated probability $\widehat{\Pr}(c_i|D)$ is equal for any class c_i and all of its class pairs. This is the case if the estimations $\widehat{\Pr}(a|c_i)$ and $\widehat{\Pr}(c_i)$ have the same value for all class pairs. These estimated probabilities depend only on the relative frequency of training examples which belong to class c_i and if applicable whose attribute values match a. These frequencies are not affected by class binarization techniques. Therefore the estimated probabilities $\widehat{\Pr}(D|c_i)$ and $\widehat{\Pr}(c_i)$ are also the same for every class and all of its class pairs. Thus the same holds for $\widehat{\Pr}(c_i|D)$ and the theoretical equivalency of the pairwise class binarization and NB occurs also in practice.

Differences between the pairwise class binarization and NB can be traced back to imprecise implementations of these methods. For example if the continuous attributes should be discretized the discretization can be applied on all training example (referred as global, abbr.: G) or on the training examples of each class pair (referred as binary, abbr.: B). The former does not change the abovementioned relative frequencies but the latter clearly does.

Considering the methods that are based on the Bradley-Terry modell we draw the conclusion that the pairwise class binarization with these methods is also equivalent to NB. These methods try to minimize the distance between $\widehat{\Pr}(c_i|D,c_{ij})$ and $\overline{\Pr}(c_i|D,c_{ij})$. As we have seen the estimated probability $\widehat{\Pr}(c_i|D)$ is the same for each class c_i and any of its class pairs. Therefore the following holds:

$$\widehat{\Pr}(c_i|D,c_{ij}) = \frac{\widehat{\Pr}(c_i|D)}{\widehat{\Pr}(c_i|D) + \widehat{\Pr}(c_i|D)} = \overline{\Pr}(c_i|D,c_{ij})$$

Thus the distance of the estimated probabilities is zero and the results of methods HT and PKPD equal those of NB.

3.3 Alternative Method

In the previous subsection we have seen that the pairwise class binarization does not improve the performance of NB. Thus we considered an alternative probabilistic approach for a pairwise estimation of NB. Its basic idea is to estimate $\Pr(c_i|D)$ by the pairwise probabilities $\Pr(D|c_{ij})$ which can be computed in two different ways. Before we

explain these ways we want to introduce our approach.

$$\begin{aligned}(m-1)\Pr(c_i|D) &= \sum_{j\neq i}\Pr(c_i|D)\\ &= \sum_{j\neq i}\Pr(c_i|D,c_{ij})\cdot\Pr(c_{ij}|D)\end{aligned}$$

$$\Leftrightarrow \Pr(c_i|D) = \frac{1}{m-1}\sum_{j\neq i}\Pr(c_i|D,c_{ij})\cdot\Pr(c_{ij}|D)$$

This approach leads to the following basic classifier:

$$\begin{aligned}&\arg\max_{c_i}\frac{1}{m-1}\sum_{j\neq i}\Pr(c_i|D,c_{ij})\cdot\Pr(c_{ij}|D)\\ =\ &\arg\max_{c_i}\sum_{j\neq i}\Pr(c_i|D,c_{ij})\cdot\Pr(c_{ij}|D)\end{aligned}$$

The two term of this classifier will be referred as

$$v_{ij} = \Pr(c_i|D,c_{ij})$$

and

$$w_{ij} = w_{ji} = \Pr(c_{ij}|D).$$

We compute these terms as follows:

$$v_{ij} = \frac{\Pr(D|c_i)\cdot\Pr(c_i)}{\Pr(D|c_i)\cdot\Pr(c_i)+\Pr(D|c_j)\cdot\Pr(c_j)}$$

$$w_{ij} = \frac{\Pr(D|c_{ij})\cdot\Pr(c_{ij})}{\Pr(D)}$$

As abovementioned $\Pr(D|c_{ij})$ can be computed in two different ways. The first one (referred as R) calculates it with the estimations of $\Pr(D|c_i)$ and $\Pr(c_i)$ a regular NB.

$$\Pr(D|c_{ij})_R = \frac{\Pr(D|c_i)\Pr(c_i)+\Pr(D|c_j)\Pr(c_j)}{\Pr(c_i)+\Pr(c_j)}$$

The second one (referred as P) calculates it by merging the training examples of class pair c_{ij}. Hence not only the quantities of classes are used for the prediction but also those of class pairs. We get:

$$t_{c_{ij}} = t_{c_i} + t_{c_j} \qquad t^{a_k}_{c_{ij}} = t^{a_k}_{c_i} + t^{a_k}_{c_j}$$

and

$$\Pr(c_{ij}) = \frac{t_{c_i}+t_{c_j}}{t} \quad \Pr(a_k|D,c_{ij}) = \frac{t^{a_k}_{c_i}+t^{a_k}_{c_j}+1}{t_{c_i}+t_{c_j}+v_k}$$

$\Pr(D|c_{ij})$ can be computed as follows:

$$\Pr(D|c_{ij})_{PW} = \prod_{k=1}^{n}\Pr(a_k|c_{ij})$$

These methods have different computational complexities on training time. The first one has the same as NB, $O(tn)$. The second one considers each training example of a given class c_i not only once but one time for each class pair c_{ij}. This increases the training time by the factor m-1. Hence the second one has a computational complexity of $O(tnm)$.

Consequentially we cannot both use this basic classifier and calculate w_{ij} regularly because this results in the same prediction as NB. Therefore we have to consider modifications of the basic classifier. We will introduce two pairs of method which are related by their modifications. The first pair uses v_{ij} for voting and w_{ij} as the weight of the votes. The second pair uses the basic classifier but estimates v_{ij} by $\Pr(c_i|c_{ij})$.

The classifier of the first pair will be referred as PNB1 and PNB2 and the classifier of the second pair accordingly PNB3 and PNB4.

$$c_{\text{PNB1}} = \arg\max_{c_i}\sum_{\substack{j\neq i\\ v_{ij}\geq v_{ji}}} w_{ij}$$

$$c_{\text{PNB2}} = \arg\max_{c_i}\sum_{j\neq i} v_{ij}\cdot w_{ij}$$

$$c_{\text{PNB3}} = \arg\max_{c_i}\sum_{\substack{j\neq i\\ \Pr(c_i)\geq\Pr(c_j)}} w_{ij}$$

$$c_{\text{PNB4}} = \arg\max_{c_i}\sum_{j\neq i}\Pr(c_i|c_{ij})\cdot w_{ij}$$

As abovementioned PNB2 is equal to the basic classifier if we calculate it regularly. This holds also for PNB1. Therefore both will be calculated only pairwise. PNB3 and PNB4 can be computed regularly and pairwise.

4 Experiments

4.1 Experimental Setup

In this section we compare the methods which we introduced in the former section with NB. To this end we used the learning environment WEKA (short for The Waikato Environment for Knowledge Analysis) of the university of Waikato, New Zealand [Witten and Frank, 2005]. We extended WEKA by two new classifier.

The first one adds new utilities to the MulticlassClassifier which deals with multiclass problems by class binarization or ECOCs. The MulticlassClassifier has been augmented by the decoding method Weighted Voting, HT and PKPD and the option of choosing the discretization type. Now it is possible to discretize on all training examples or on the trainining examples of each class pair.

The second one implements our own methods PNB1 to PNB4 which can be regularly or pairwise computed.

Our experiments consist of two test series. We use data sets of the UCI repository that represent multiclass problems. Before we apply the classifiers the data set will be the one way or another discretized. We use the discretization method of [Fayyad and Irani, 1993] which is already implemented in WEKA.

The first test series compares our own PNB methods with NB. PNB1 and PNB2 will be calculated pairwise. The other methods use both computation techniques. In any case the discretization will be applied on all training examples.

The second test series compares unordered and pairwise class binarizations with NB. The discretization will be applied on the class pairs. In the unordered class binarizations it will also be applied on the whole data.

We compare the methods with a sign test. If the null hypothesis is discarded with niveau 0.95 or 0.99 we call the methods significant (abbr.: S) or accordingly highly significant (abbr.: HS) different. Else we call them equivalent (abbr.: E).

The results of the experiments are summarized in the table in the appendix. The tables contain the error rates of each method on the data sets, the quantities (how many times the method won, lost or was equal to NB) Win, Loss and Tie and the result of the sign tests. The error rates were obtained by stratified 10x10 cross validation and rounded

to fit in the table. The quantities Win, Loss and Tie were computed on the error rates before rounding was applied.

4.2 Results

In the first test series the PNB methods showed higher error rates than NB. Though PNB1, PNB2 and the regularly computed PNB 4 are equivalent to NB. PNB3 and the pairwise computed PNB4 are significant worse than NB. Comparing the regular and the pairwise computation the pairwise one is slightly worse than the regular. The application of PNB method is not advisable because of their bad results and if applicable their higher computational costs.

In the second test series we draw two conclusions. First the unordered class binarization is equivalent to NB, irrespective of choosing the binary or global discretization. Second the pairwise class binarization with the binary discretization is equivalent to NB for each decoding method we introduced but produces lower error rates in most of the cases.

5 Conclusion

In this paper we have drawn several conclusion about class binarizations with a Naive Bayes classifier. First we have shown that the unordered class binarization is not equal to a regular Naive Bayes classifier. Second we have proven that the pairwise class binarization is equivalent to NB for common decoding methods like Voting, Weighted Voting and the proposals of [Hastie and Tibshirani, 1997; Price *et al.*, 1994].

We suggested some alternative methods for a pairwise calculation of a modified Naive Bayes classifier. Our experiments showed that these methods did not improve the performance of a Naive Bayes classifier. Additionally the experiments exhibited that class binarizations can increase the performance of a Naive Bayes classifier if we apply the discretization on the training examples of each binary problem.

For further readings we suggest [Sulzmann, 2006] that gives a more detailed description of the proofs and experiments and considers some additional pairwise approaches.

References

[Fayyad and Irani, 1993] Usama M. Fayyad and Keki B. Irani. Multi-interval discretization of continuous-valued attributes for classification learning. In *Proceedings of the 13th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1022–1029, 1993.

[Fürnkranz, 2002] Johannes Fürnkranz. Round robin classification. *Journal of Machine Learning Research (JMLR)*, 2:721–747, 2002.

[Fürnkranz, 2003] Johannes Fürnkranz. Round robin ensembles. *Intelligent Data Analysis*, 7(5):385–403, 2003.

[Hastie and Tibshirani, 1997] Trevor Hastie and Robert Tibshirani. Classification by pairwise coupling. In Michael I. Jordan, Michael J. Kearns, and Sara A. Solla, editors, *Advances in Neural Information Processing Systems (NIPS)*. The MIT Press, 1997.

[Price *et al.*, 1994] David Price, Stefan Knerr, Léon Personnaz, and Gérard Dreyfus. Pairwise neural network classifiers with probabilistic outputs. In Gerald Tesauro, David S. Touretzky, and Todd K. Leen, editors, *Advances in Neural Information Processing Systems (NIPS)*, pages 1109–1116. MIT Press, 1994.

[Sulzmann, 2006] Jan-Nikolas Sulzmann. Paarweiser Naive Bayes Klassifizierer. Diplomarbeit, Technische Universität Darmstadt, D-64289, Darmstadt, Germany, July 2006.

[Witten and Frank, 2005] Ian H. Witten and Eibe Frank. *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann, 2 edition, 2005.

[Wu *et al.*, 2004] Ting-Fan Wu, Chih-Jen Lin, and Ruby C. Weng. Probability estimates for multi-class classification by pairwise coupling. *Journal of Machine Learning Research (JMLR)*, 5:975–1005, 2004.

data set	NB	$PNB1_P$	$PNB2_P$	$PNB3_P$	$PNB3_R$	$PNB4_P$	$PNB4_R$
anneal	4,00	4,17	3,91	3,51	3,49	3,44	3,42
audiology	26,95	26,72	26,89	31,13	27,79	29,13	27,80
autos	34,97	35,00	34,34	33,78	34,05	33,61	34,79
balancescale	27,93	27,93	27,97	35,59	38,25	27,82	27,88
glass	28,70	29,04	29,06	31,18	31,05	31,41	29,48
hypothyroid	1,74	1,70	1,84	3,19	4,77	3,11	2,86
iris	6,69	6,69	6,69	6,65	6,69	6,65	6,69
letter	25,95	26,80	27,20	47,38	31,27	28,99	25,95
lymph	14,83	14,83	14,70	19,06	17,76	16,12	17,73
primary-tumor	49,66	49,54	49,95	50,97	50,49	50,03	49,95
segment	8,93	9,27	9,36	14,63	8,93	14,63	8,93
soybean	7,14	7,86	8,05	15,05	11,54	10,39	7,28
splice	4,63	4,78	4,93	33,88	48,12	20,00	6,45
vehicle	39,32	39,33	39,04	57,68	57,83	41,65	39,11
vowel	41,27	41,51	41,44	42,77	41,27	42,77	41,27
waveform-5000	20,03	20,03	20,03	54,68	66,16	26,34	19,91
yeast	42,68	42,72	42,74	46,19	43,81	43,79	42,94
zoo	7,22	7,22	6,18	11,18	11,01	10,25	5,95
Mittel	21,81	21,95	21,91	29,92	29,68	24,45	22,13
compared to NB							
Win		4	6	3	2	4	6
Loss		10	11	15	13	14	9
Tie		4	1	0	3	0	3
method statistically equal to NB?							
E/S/HS		E	E	HS	HS	S	E

Table 1: Results of the first test series: PNB1-PNB4: error rates in percent

	Binary discretization		Global discretization				
data set	NB	$1vsAll_B$	$1vsAll_G$	V	WV	HT	PKPD
anneal	4,00	2,28	2,40	3,10	3,07	3,31	3,09
audiology	26,95	27,75	27,75	26,95	26,95	26,82	26,95
autos	34,97	32,80	32,05	32,70	31,89	31,84	32,78
balancescale	27,93	27,97	26,87	26,20	26,20	26,20	26,20
glass	28,70	28,35	31,90	32,08	30,21	30,76	31,38
hypothyroid	1,74	2,13	2,25	1,73	1,68	1,77	1,69
iris	6,69	6,69	6,97	6,56	6,56	6,56	6,56
letter	25,95	26,10	27,38	26,48	26,31	26,28	26,37
lymph	14,83	14,44	14,72	14,80	14,97	14,72	14,73
primary-tumor	49,66	48,62	48,62	49,66	49,66	49,42	49,66
segment	8,93	10,88	9,07	8,54	8,51	22,25	8,53
soybean	7,14	7,63	7,63	7,14	7,14	7,13	7,14
splice	4,63	5,11	5,11	4,63	4,63	4,63	4,63
vehicle	39,32	38,96	37,79	37,66	37,49	38,26	37,55
vowel	41,27	41,03	35,17	36,15	33,41	33,34	34,16
waveform-5000	20,03	21,22	21,32	20,08	20,08	23,47	20,08
yeast	42,68	42,63	41,51	42,80	41,57	41,83	42,26
zoo	7,22	5,35	3,96	7,49	6,12	6,42	6,64
Mittel	21,81	21,25	21,66	21,37	20,91	21,94	21,13
compared to NB							
Win		9	9	9	10	12	11
Loss		8	9	5	4	6	3
Tie		1	0	4	4	0	4
method statistically equal to NB?							
E/S/HS		E	E	E	E	E	E

Table 2: Results of the second test series: unordered and pairwise class binarizations with Voting, Weighted Voting, HT and PKPD: error rates in percent

Autorenindex