A Pool of Topics: Interactive Relational Topic Visualization for Information Discovery

Inessa Seifert and Michael Kruppa

German Research Center for Artificial Intelligence (DFKI GmbH),
Alt-Moabit 91c, Berlin, 10559, Germany
inessa.seifert@dfki.de,michael.kruppa@dfki.de
http://www.dfki.de

Abstract. In this paper, we present a novel relational visualization that supports people at information discovery tasks in digital libraries. This visualization displays search query results structured into topics and highlights the intersections between them. The proposed visual representation introduces interactive drag-and-drop operations for manipulation of the generated topics. These operations mirror the human online searching strategies that involve boolean AND, OR, and NOT operators. In doing so, the information seeker can refine (or relax) a search query in an interactive way during a focusing or a defocusing phase. The intersections of topics are made explicitly visible to enable the information seeker to avoid frustrating "no hits" situations.

Key words: information visualization, boolean operators, information discovery, online search strategies

1 Introduction

Modern digital libraries provide a seamless access to a vast amount of scientific literature. The amount of information available on the Internet has tremendously increased over the past years. Retrieving an article of a known title (or an author) is sufficiently fast and easy today. However, finding appropriate literature on a topic the information seeker is not familiar with is a time consuming task. During information seeking and discovery tasks, the lack of domain specific knowledge leads to underdetermined and unclear search goals that are reflected in the definition of vague search queries. Such search queries contribute to a huge number of resulting hits. Examining a great amount of scientific literature is a time consuming endeavor. Therefore, such vaguely defined queries are usually followed by a more focused formulation of different search terms combined with boolean AND, OR, and NOT operators. Yet, too many, or too specific search terms often deliver no results[1]. In such situations, people try to broaden their search goals by returning to less specified queries. The information seeking process encompasses

[1] up to 30-50% of queries containing boolean operators together with precisely defined search terms deliver no results [1,2]

series of *focusing* phases that are followed by *defocusing* phases, in which people examine the retrieved hits, learn more about the topic, and continuously change their attitude toward the search goals.

In this paper, we present a novel relational visualization approach that displays query results structured into topics and highlights the intersections between them. The proposed visual representation introduces interactive drag-and-drop operations for manipulation of the generated topics. These operations mirror the logical AND, OR, and NOT operators that enable the information seeker to refine (or relax) the search queries during a search session. The intersections of topics are made explicitly visible to enable the information seeker to avoid frustrating "no hits" situations.

2 Information Seeking

Information seeking is a complex and cognitively demanding task that has a close relation to learning and problem solving [3].

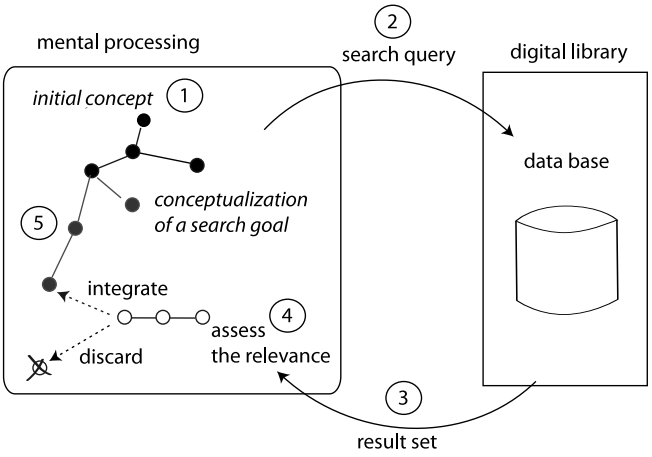


Fig. 1. Information seeking process

The information seeker starts with an initial concept of the search goal (Fig. 1, step 1) that is derived from the prior knowledge about the problem domain and defines an initial search query (Fig. 1, step 2). Based on the new knowledge acquired from the analysis of the query results (Fig. 1, step 3), people think of new concepts, revise their search goals, and formulate new queries. The search goals as well as criteria for assessing the relevance of articles from the query results (Fig. 1, steps 4 and 5) evolve during the information seeking process and cannot be specified in advance [4]. Having no specific well-defined goal and also no specific criteria for determining the solution quality, information

seeking is regarded as an ill-structured problem [5]. The information seeking process terminates as soon as a user finds appropriate articles or decides to interrupt the search.

Looking at the information seeking process from the problem solving point of view, we have to identify the dimensions of the problem domain, possible actions that people can perform during the search to reach a new problem solving state, as well as search strategies that help people to reduce the amount of information they have to process.

2.1 Dimensions

A scientific article is characterized by the meta-data that makes it unique: *title*, *author(s)*, *editor(s)*, *year*, publishing *source* (e.g., the name of the journal, or conference proceedings, volume, issuer), *abstract*, and its *text*. Some digital libraries (e.g., ACM[2], CiteSeer[3]) provide further information such as *keywords*, *categories*, and links to the *referenced articles*.

In the following, we will discuss possible search paths resulting from the information choices accessible to the information seeker.

Each article is distinguished by a title, a short description (i.e., abstract), and its text. These attributes can contain specific words that trigger the formulation of refined search queries. Author names are usually augmented with contact information, such as e-mail and author's affiliation, i.e., name and address of an institution. Familiarity with the work of a specific author as well as the reputation of a scientific institution are important factors that can guide the information seeker to the publications of a particular author, groups of scientists, or research institutions [6,7]. A year is derived from the publishing date of an article in a scientific journal or a conference proceeding that represents a source. Examining articles that belong to a particular conference proceeding or a journal is another path that can be taken by the information seeker to continue the search. Referenced articles can provide even more hints about where further information about specific topics can be found.

The variety of the illustrated possibilities to follow different search paths leads to a vast growth of the search space. To reduce the amount of information to be processed and to facilitate the search, publications contained in a digital library are structured into different categories and augmented with key words.

Categories are traditionally maintained by librarians who are responsible for the creation and preparation of literature catalogs. Modern data mining and clustering methods automatically structure query results into different clusters and accordingly label them with frequently occurring terms, i.e., *topics* [8].

Key words are usually assigned either by authors of the corresponding articles or by librarians[4].

[2] http://portal.acm.org/
[3] http://citeseerx.ist.psu.edu/
[4] for example, the primary and secondary index of the ACM digital library

2.2 Online searching strategies

To deal with a huge amount of data that has to be processed during information seeking tasks, people developed sophisticated online searching strategies. Marchionini [9] defines a strategy as "the approach an information seeker takes to a problem. Strategies are sets of ordered tactics that are consciously selected, applied and monitored to solve an information problem.[...] Tactics are discrete intellectual choices or prompts manifested as behavioral actions during an information-seeking session." He distinguishes four general strategies commonly employed during information seeking and information seeking tasks that represent these behavioral actions.

A first online searching strategy is called the "building blocks" approach [10]. During problem definition, the information seeker identifies the main facets and concept groups associated with the problem. Concepts are specific terms that frame the content of a search topic. For example, a user is interested in methods for improving the interaction with digital libraries. He/she can specify search terms that resemble this concept such as "usability for digital libraries," "user interaction," or "queries in digital libraries." (see Fig. 2)

Facets reflect meta-data, for example, a year of publication, research institutions, or specific properties of a document, for example, its format [11]. Concepts are employed for query formulations that include search terms combined with logical operators. Facets are used to filter the retrieved query results. In other words, concepts help to dice and facets help to slice the data space into observable portions. If a combination of topics delivers no results, the user can use synonyms or replace one of them with another topic.

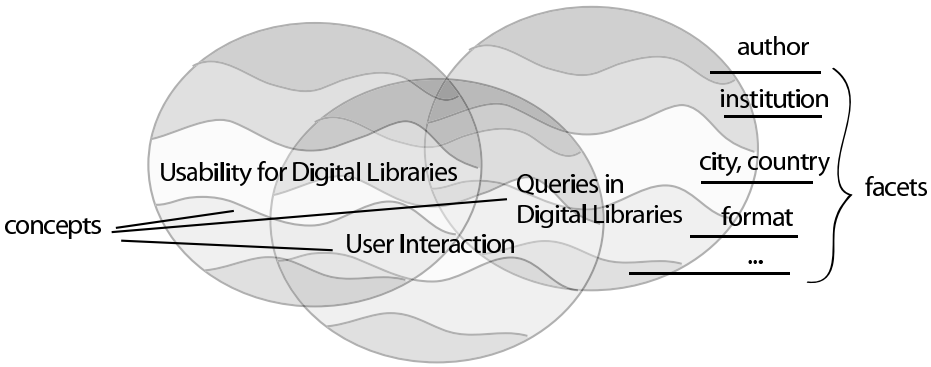


Fig. 2. "Building blocks" strategy: slicing and dicing the data space into facets and concepts

A second strategy that is widely used is the "successive fractions" approach [12]. This approach works well if a user has a vague or broad conceptualization of the search goals. Like the first strategy, it is based on the commonly known

"divide and conquer" problem solving principle. Using this strategy, the information seeker successively refines a large subset of data retrieved from a information system by introducing search terms that become more and more specific in each problem solving step. For example, the user can start with a general term "digital library", and make it more specific by constructing a search query "digital library AND user interaction".

The "building blocks" strategy as well as "successive fractions" facilitate the search process by breaking it into a sequence of systematic and discrete steps.

A third general strategy is the "pearl growing" approach [12]. The user starts with a specific document or document set that is relevant (a pearl) and uses the characteristics of that document to successively retrieve further relevant documents. The information seeker uses assigned key words, title or text words, names, citations, a year of publication, or other features to construct queries in order to find similar documents.

A fourth general strategy was described by Hawkins & Wagers [13] as "interactive scanning". This strategy requires a more intensive interaction with the content of the documents from a user. The information seeker starts with a selection of documents relevant to the problem area sorted out from the initial result set. He/she scans these documents in order to identify key features (e.g., authors, terminology, methods) that trigger the formulation of successive queries.

Another less popular but still effective strategy proposed by Vigil [14] is called "closed-loop relevance clustering." This strategy uses the NOT operator to successively remove redundant documents from sets formed as a result of query modifications and combinations.

In this contribution, we will present an approach for visualization and manipulation of information that implements a combination of the "building blocks," "successive refinement" and provides support for the "close-loop relevance clustering" strategy.

2.3 Visual Representations

In this section, we will outline suitable visualization approaches for representing the hierarchical structure of digital libraries that were pursued so far. A more detailed overview about different visualization techniques can be found in [15]. We will discuss how these approaches communicate the structure of the data space and which interactive operations are provided to the user to perform an information seeking task.

The basic structural elements of relational visualizations (also termed graphs) encompass *nodes* and *edges* between them.

Spatially inspired *concept spaces* display different concepts that involve central terms retrieved from a data base using, for example, an automated thesaurus generation algorithm [16]. Spatial distance between the concepts conveys similarity relations between them.

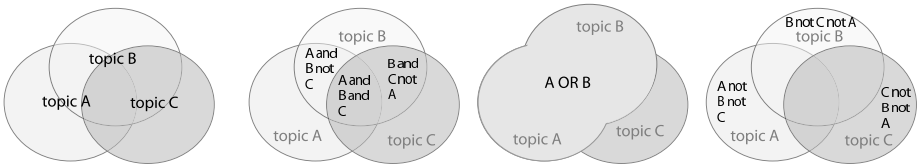
Topic maps provided by HighWire digital library consist of tree-based structures that include hierarchically structured topics [5]. Edges are used to visualize the hierarchical relations between topics and subtopics. The interactive operations allow for expanding nodes in order to change the level of granularity.

3D-visualizations present the content of a digital library as cone trees [17,18]. Cones stand for different topics and subtopics that contain documents represented as leaves of a tree. The user can interactively rotate the cones to examine the titles of the documents.

The outlined representations support browsing operations such that the user can interactively select, expand, or collapse different nodes in order to examine the data space at different levels of granularity. These operations, however, do not allow for applying the logical AND, OR, and NOT operators that are essential for the online searching strategies addressed in section 2.2.

3 Interactive Relational Topic Visualization

Cutting et al. [8] proposed an efficient online clustering technique for structuring query results into topics. This approach enabled the information seeker to use the extracted topics for the formulation of successive queries that included logical AND, OR, and NOT operators. Yet, inappropriate combinations can often deliver no results. Our vision is to support the selection of logical operators by making the critical combinations of topics that can deliver no results visually accessible before a user states a new query.



(a) Topics A, B, and C (b) intersection of topics (c) unification of topics A OR B (d) complement of topics

Fig. 3. Some variations of possible relations between topics A, B, and C

Usually, such relations are visualized using Venn diagrams [19] that depict topics as overlapping circles (see Fig. 3). The results of the logical AND operator belong to the intersection of the circles (Fig. 3(b)), the OR operator unifies the topics (Fig. 3(c)), and the NOT operator includes only the part of a topic that does not overlap with the second topic (Fig. 3(d)). It seems that Venn diagrams are very well suited as a graphical representation to support people at construction of search queries. Yet, determining all possible variations in advance

[5] http://highwire.stanford.edu/help/hbt/

is a computationally demanding task ($n!$ combinations). The same applies for the perceptual inspection of Venn diagrams. Intersections of two or three topics are visually tractable. Examining various intersections of a considerable amount of topics is difficult. To keep the response times of our system short, we refrain from determining all possible variations. To provide a better visual access to the relations between different topics, we focus on critical relations that can lead to "no hits" situations. If an intersection of two topics is empty, the NOT operation will be useless and the AND operation will deliver an empty result set. To prevent this, we retrieve in addition to topics pairwise intersections between them. The main idea behind this approach is to give the user an immediate impression of the relationships between generated topics.

3.1 A pool of topics

Figure 4 shows the user interface of the **Di**gital **Li**brary **A**ssistant (**DILIA**). The screen is separated into two parts. The left side illustrates an example pool of topics generated from the search query *digital library*. In the top left corner of the screen is a query panel. Here, a user can formulate a search query, for example *digital library*. On the right side of the screen, the user can examine the resulting hits.

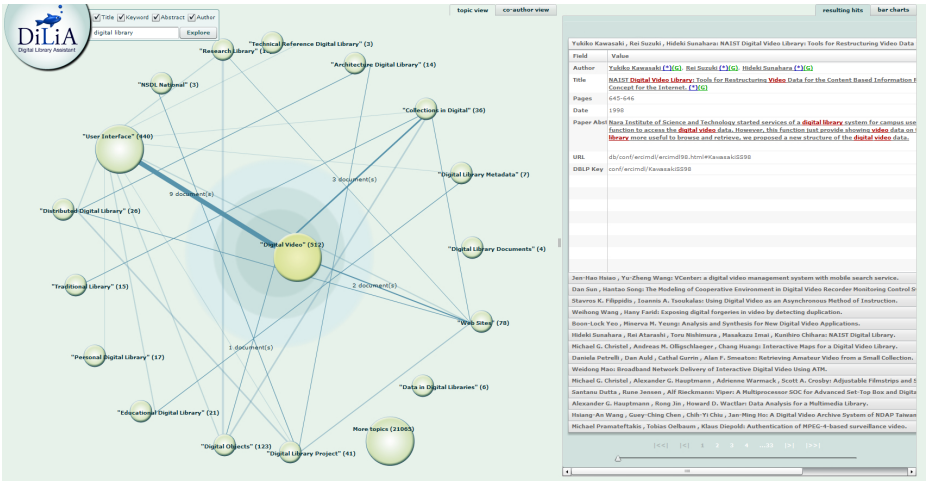


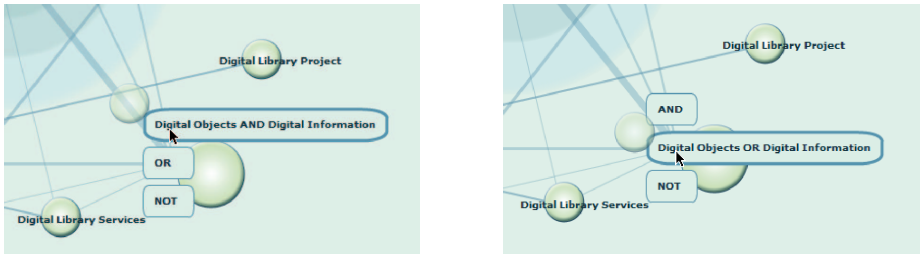
Fig. 4. Topics generated from the example search query *digital library*

In the first step, DILIA processes the search query and generates a set of topics visualized as round blobs. The size of a blob stands for the number of documents that belong to a topic. The blobs are labeled with retrieved topic names together with the number of the corresponding documents. In the second step, it determines how many documents are shared between each pair of topics

(with $O(n^2)$ run-time complexity, where n is the number of topics). Finally, the number of shared documents is visualized as an edge between topics. The edge thickness depends on the number of documents shared between the topics. By rolling over a topic, DILIA highlights the corresponding edges together with the number of shared documents.

3.2 Interactive drag-and-drop manipulation of topics

The user can drag and drop the blobs representing topics. This operation invokes an event that shows possible logical AND, OR, and NOT operations on a combination of topics that leads to further results. In doing so, DILIA enables the user to apply the "building blocks" strategy. If the intersection between



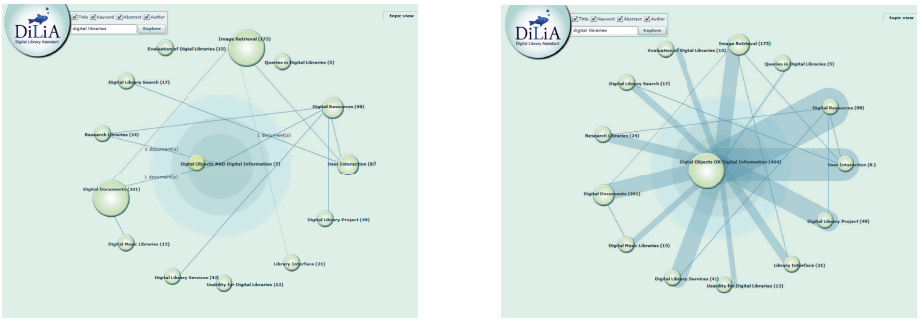
(a) "Digital Objects" AND "Digital Information"

(b) "Digital Objects" OR "Digital Information"

Fig. 5. Drag-and-drop manipulations of topics

the topics is empty, the system proposes only OR operation. In figure 5(a) the logical AND operation between topics "Digital Objects" AND "Digital Information" is automatically generated from the example search query *digital library*. Figure 5(b) shows another alternative combination of the involved topics, i.e., "Digital Objects" OR "Digital Information." Figure 6(a) shows the result of the AND combination of the topics "Digital Objects" AND "Digital Information." Figure 6(b) shows the corresponding OR combination of the topics. The edges of the AND combination become thin depicting the precision of the constructed sub-query. The edges of the OR combination are more prominent, since they include more documents as compared to the original pool of topics. In the same way, the user can employ NOT operators to remove documents contained in the dragged topic from the target topic. The NOT operation gives the user the opportunity to apply the "close-loop relevance clustering" strategy.

After a user defines a new logical combination by dropping a topic on a target topic, DILIA determines the intersections between the remaining topics and the new logical combination of the two topics. To avoid an increasing number of topics in each drag-and-drop interaction, we decided to remove the modified



(a) "Digital Objects" AND "Digital Information"

(b) "Digital Objects" OR "Digital Information"

Fig. 6. Example query results performed using drag and drop operations

topics from the screen. This combination of topics is than displayed in the center of the pool of topics.

A single click on a topic puts the selected topic in the center of the pool and filters the lists of articles on the right side of the screen according to the selected topic. This allows for a better inspection of intersections among the revolving topics and corresponding articles that belong to the topic in the center.

A double click on a topic invokes an event that extends the current search query included in the query panel with the clicked topic label. As a response, the system processes the new search query and generates new topics and corresponding topic intersections. This procedure corresponds to the "successive fractions" approach described in section 2.2.

4 Realization

For our prototype implementation, we make use of a web-based client server architecture. On the client side, we have developed a Rich Internet Application (RIA) realized in Adobe Flex[6]. This application follows the model-view-controller (MVC) concept. We have build the flex prototype based on the Cairngorm[7] MVC implementation which allowed us to ensure a consequent MVC realization. The client utilizes server side PHP[8] classes to query the digital library database which is realized as a Lucene[9] index. In order to call Lucene methods from PHP we utilize the PHP Javabridge [10]. Finally, the communication between Flex (which is compiled into a Flash Movie) and the server side PHP classes is re-

[6] http://www.adobe.com/products/flex/
[7] http://opensource.adobe.com/wiki/display/cairngorm/
[8] http://www.php.net
[9] http://lucene.apache.org/
[10] http://php-java-bridge.sourceforge.net

alized using Weborb[11]. Weborb handles the serialization/deserialization of data and the interfacing of methods between PHP and Flex. To determine the topic labels, we use the Carrot clustering engine[12].

The server side environment is based on the Apache HTTP Server[13] and Apache Tomcat[14]. The information flow between server and client is visualized in figure 7.

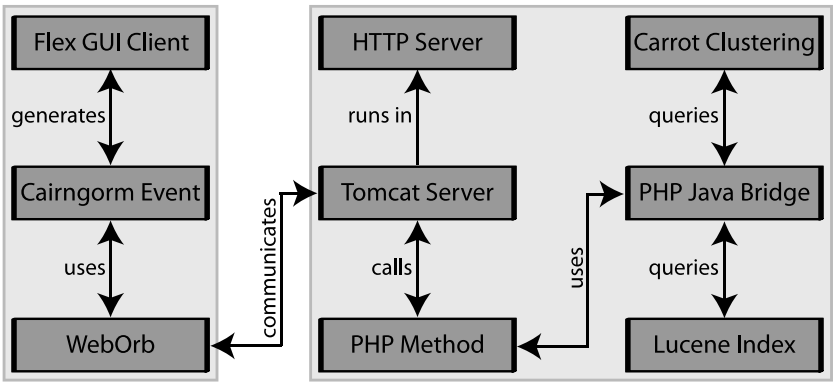


Fig. 7. Communication flow between client (left box) and server (right box)

The computational effort for deriving content clusters is quite high due to the complexity and mere size of the underlying data structures (more than one million data sets). Since the envisioned prototype should be able to do these calculations in real time, we decided to take a different approach. Similar to [8], instead of clustering the complete query result set, we used a clustering algorithm only on a subset of the result list. By clustering only the first 300 hits, we can efficiently retrieve topics with corresponding labels.

Since the Lucene Index can efficiently process query requests that can return more than 200.000 hits, we interpreted the extracted topics as sub-queries. To arrive at a subset of documents that correspond to each extracted topic, we simply form a new Lucene query combining the original query with each sub-query (using the logical operator AND). The following example illustrates the idea:

```
Initial Query: digital library

Extracted topics: "visual interfaces", "usability"...
```

[11] http://www.themidnightcoders.com/products/weborb-for-php/
[12] http://project.carrot2.org/
[13] http://httpd.apache.org/
[14] http://tomcat.apache.org/

```
1. topic sub-query: (digital library) AND "visual interfaces"
2. topic sub-query: (digital library) AND "usability"
```

With this strategy, we can not guarantee that the resulting subsets of documents contain all hits that correspond to the original search query. We defined a special topic labeled with "More topics" that gives the user the opportunity to retrieve additional topics from the next 300 hits belonging to the result set.

In a next step, we retrieve the intersections for each topic. The following section explains the basic procedure.

4.1 Determining topic intersections

The algorithm for retrieving the topic intersections can be summarized as follows.

Retrieved topic labels	"Digital Information" (267)	"Evaluation of Digital..." (15)	"Digital Library Project" (49)	"Digital Objects" (145)	"Digital Library Services" (42)	"User Interaction" (81)	"Digital Resources" (98)	"Queries in Digital..." (5)	"Digital Library Search" (17)	"Usability for Digital..." (13)	"Digital Music Libraries" (15)	"Digital Documents" (201)	"Research Libraries" (24)	"Library Interface" (21)	"Image Retrieval" (173)
"Digital Information"	267	-	-	7	1	2	5	-	-	-	-	5	2	-	-
"Evaluation of Digital..."	-	15	-	-	-	-	-	-	-	-	-	-	-	-	-
"Digital Library Project"	-	-	49	-	-	-	1	-	-	-	-	-	-	-	-
"Digital Objects"	-	-	-	145	1	1	1	-	-	-	-	1	1	-	-
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
"Image Retrieval"	-	-	-	-	-	-	-	-	-	-	-	-	-	-	173

Fig. 8. Topic intersections matrix resulting from query: "digital library"

1. We take the query submitted by the user and derive a limited number of topics.
2. Based on this set of topics and corresponding topic labels, we then perform a number of Lucene sub-queries.
3. Each sub-query is a combination of the original query and one of the determined topic labels. In this way, we can find out the number of documents included in each topic.
4. Based on these results, we then construct a n x n matrix (where n is the number of topics derived from the current query). The matrix contains the

number of documents that belong to the intersections between the topics (except for the special label "More topics") (see Fig. 8).
5. In order to fill the cells of the matrix, we again perform concatenated Lucene queries for each combination of topics and the original query.

After this final operation, we get a matrix representing the topics derived from the initial query, the number of documents contained in each of these topics, and the number of documents contained in each pairwise intersection of the topics.

5 Outlook and future work

In this contribution, we proposed an interactive Flex-based visualization capable of displaying topics extracted from query results and intersections between them. This visualization provides the user with interactive drag-and-drop operations that allow for employing boolean AND, OR, and NOT operators for combining the generated topics in order to obtain further results. Such combinations of topics enable the user to employ commonly used online searching strategies such as the "building blocks," "successive fractions," and "closed-loop relevance clustering" approach. In order to support the "pearl growing" approach, we need to modify the currently used clustering procedure. Therefore, we consider the support for the "pearl growing" approach by DILIA (Digital Library Assistant) as matter of future work. Our next steps in developing DILIA encompass usability studies with human users in order to evaluate the proposed interaction with topics and their visualization.

In the current implementation, the user can manipulate only those topics that are proposed by the information system. To support the "interactive scanning" approach and allow for more flexibility, we plan to extend the topic view with a possibility to specify additional user-defined topics. The system can determine intersections between the retrieved and the user-defined topics and highlight them, for example, with a distinct color.

The Rich Internet Application (RIA) technology employed for the prototype implementation of DILIA demonstrates new perspectives for the interaction design and user interface development to support the information discovery tasks. In doing so, the proposed relational topic visualization makes the structures concealed in a digital library visually accessible to information seekers and enables the exploration of unfamiliar problem domains in an efficient and aesthetically pleasing way.

6 Acknowledgments

The research project DILIA (Digital Library Assistant) is co-funded by the European Regional Development Fund (EFRE) under grant number 10140159. We gratefully acknowledge this support.

References

1. Borgman, C.L.: Why are online catalogs hard to use? lessons learned from information-retrieval studies. Journal of the American Society for Information Science **37**(6) (1986) 387–400
2. Bates, M.: Subject access in online catalogs: A design model. Journal of the American Society for Information Science **37**(6) (1986) 357–376
3. Vakkari, P.: Task complexity, problem structure and information actions - integrating studies on information seeking and retrieval. Information Processing and Management **35**(6) (1999) 819–837
4. Bates, M.: The design of browsing and berrypicking techniques for the online search interface. Online Review **13**(5) (1989) 407–424
5. Simon, H.A.: The structure of ill-structured problems. Artificial Intelligence **4** (1973) 181–201
6. Barry, C.L.: User-defined relevance criteria: An exploratory study. Journal of the American Society for Information Science **45**(3) (1994) 149–159
7. Anderson, T.D.: Studying human judgments of relevance: interactions in context. In Ruthven, I., ed.: Proceedings of the 1st international conference on Information interaction in context, ACM (2006) 6–14
8. Cutting, D.R., Karger, D.R., Pedersen, J.O., Tukey, J.W.: Scatter/gather: a cluster-based approach to browsing large document collections. In: SIGIR '92: Proceedings of the 15th annual international ACM SIGIR conference on Research and development in information retrieval, New York, NY, USA, ACM Press (1992) 318–329
9. Marchionini, G.: Information seeking in electronic environments. Cambridge University Press, New York, NY, USA (1995)
10. Harter, S.: Online information retrieval: Concepts, principles, and techniques. Academic Press, Orlando, FL (1986)
11. Hearst, M.A.: Uis for faceted navigation: Recent advances and remaining open problems. In: in the Workshop on Computer Interaction and Information Retrieval, HCIR 2008. (2008)
12. Meadow, C., Cochrane, P.: Basics of online searching. John Wiley and Sons, New York (1981)
13. Hawkins, D.T., Wagers, R.: Online bibliographic search strategy development. (1989) 88–95
14. Vigil, P.J.: The psychology of online searching. Journal of the American Society for Information Science **34**(4) (1983) 281–287
15. McKiernan, G.: New age navigation: Innovative information interfaces for electronic journals. The Serials Librarian **45**(2) (2003) 87–123
16. Zhang, J., Mostafa, J., Tripathy, H.: Information retrieval by semantic analysis and visualization of the concept space of d-lib magazine. D-Lib Magazine **8**(10) (October 2002)
17. Robertson, G.G., Mackinlay, J.D., Card, S.K.: Cone trees: animated 3d visualizations of hierarchical information. In: CHI '91: Proceedings of the SIGCHI conference on Human factors in computing systems, New York, NY, USA, ACM (1991) 189–194
18. Mizukoshi, D., Hori, Y., Gotho, T.: Extension models of cone tree visualizations to large scale knowledge base with semantic relations. In: The 14th International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision'2006. (2006)

19. Venn, J.: On the diagrammatic and mechanical representation of propositions and reasonings. Philosophical Magazine and Journal of Science **9**(50) (1880)