Domain Adaptive Relation Extraction for Semantic Web

Feiyu Xu, Hans Uszkoreit, Hong Li, Peter Adolphs and Xiwen Cheng

Abstract In the THESEUS Alexandria use case (Alexandria), information extraction (IE) has been intensively applied to extract facts automatically from unstructured documents such as Wikipedia and online news, in order to construct ontology-based knowledge databases for advanced information access. In addition, IE is also utilized for analyzing natural language queries for the Alexandria question answering system. The DARE system, a minimally supervised machine learning system for relation extraction developed at DFKI LT-Lab, has been adapted and extended to the IE tasks for Alexandria. DARE is domain adaptive and has been used to learn relation extraction rules automatically for the Alexandria-relevant relations and events. Furthermore, DARE is also applied to the Alexandria opinion mining task for detecting opinion sources, targets and polarities in online news. The DARE system and its learned rules have been integrated into the Alexandria IE pipeline.

1 Introduction

Information extraction (IE) has been acknowledged as a core information technology for the constantly growing digitalized world, in particular, for the World Wide Web (WWW). The goal of IE systems is to find and link pieces of the relevant information from natural language texts and store these information pieces in a database format. As an alternative to storing the extracted information pieces in a database, these pieces could also be appropriately annotated in a markup language and thus be made available for indexing and database retrieval [4, 8]. Tim Berners-Lee defines Semantic Web as ”a web of data that can be processed directly and indirectly by

Feiyu Xu, LT-Lab, DFKI, Germany, e-mail: feiyu@dfki.de
Hans Uszkoreit, LT-Lab, DFKI, Germany, e-mail: hansu@dfki.de
Hong Li, LT-Lab, DFKI, Germany, e-mail: lihong@dfki.de
Peter Adolphs, LT-Lab, DFKI, Germany, e-mail: peter.adolphs@dfki.de
Xiwen Cheng, LT-Lab, DFKI, Germany, e-mail: xiwen.cheng@dfki.de

machines." However, we know that the current web is still far away from the structured setup of the Semantic Web because the WWW was originally designed for the publication and consumption of content by humans, and not by machines. Thus, the IE technologies are important for closing the gap between the current Web and the Semantic Web with two major goals:

- allowing information providers to analyze and extract structured data from free Web texts and identify and link the data
- providing end users more natural and advanced search options to the Web content such as semantic search and question answering systems.

Therefore, IE is central for the Alexandria search platform, whose task is to provide intelligent semantic information access to the online information available on the Web. The central IE tasks include finding references to relevant concepts or objects such as names of people, companies and locations, as well as detecting relationships among them, e.g., the birth place of a Nobel Prize winner. Let us look at the following text (1) about the Nobel Prize award event provided by [19]:

Example 1. The *Physics prize*, also *$978,000*, will be shared by *Dr. Robert Laughlin* of *Stanford University*, *48*, *Dr. Horst Stoermer*, *49*, a German-born professor who works both at *Columbia University* in *New York* and at *Bell Laboratories* in *Murray Hill*, *N.J.*, and *Dr. Daniel Tsui*, *59*, a Chinese-born professor at *Princeton University*.

If we want to extract events of prize winning, the relevant concepts to be extracted from the above texts are entities such as *prize area*, *monetary amount*, *person name* and *organization* (see examples in Table 1). Award relevant relations include the relation between *person* and *organization* and the relation among *person*, *prize area* and *monetary amount* (Table 2).

concept	**extracted entities**
prize area	*physics*
person name	*Dr. Robert Laughlin*, *Dr. Horst Stoermer*, *Dr. Daniel Tsui*
monetary amount	*$978,000*
organization	*Stanford University*, *Columbia University*, *Princeton University*

Table 1: An example of concept entities

One pillar of the Alexandria platform is its knowledge acquisition pipeline. The pipeline builds upon raw text sources. This can be either a relatively static document collection such as Wikipedia or a dynamic daily news collection filled by a news aggregator, which gathers current issues of German newspapers and magazines on a daily basis and cleanses them for further processing. Once the content-bearing text content is identified and extracted, it is linguistically analyzed and everything is

relation	**extracted relation instances**
person, affiliation	
	⟨ *Dr. Robert Laughlin*, *Stanford University* ⟩ ⟨ *Dr. Horst Stoermer*, *Columbia University* ⟩ ⟨ *Dr. Daniel Tsui*, *Princeton University* ⟩
person, prizeArea, monentaryAmount	
person prize area monentary amount	⟨ {*Dr. Robert Laughlin*, *Dr. Horst Stoermer*, *Dr. Daniel Tsui* }, *physics*, *$978,000* ⟩

Table 2: An example of relation instances

passed to the IE modules, which try to squeeze out every piece of information that is relevant with respect to the modeled target domains.

In Alexandria, IE has been intensively applied to enrich the Alexandria knowledge base – a rich, ontology-driven repository of world knowledge, covering famous people, organizations, locations, cultural artifacts, events, and the like. Furthermore, it is also utilized for analyzing natural language queries for the Alexandria question answering system. Moreover, one of the central topics in the IE research, namely, domain-adaptive relation extraction, has been addressed in Alexandria. The key feature of "domain-adaptive" systems is that they can be applied to new domains and applications with little effort. We have further developed the domain-adaptive relation extraction system DARE of DFKI LT-Lab for the Alexandria IE tasks.

DARE is a minimally supervised machine learning system for relation extraction [19, 21]. "Minimally supervised" means that only very little human intervention is needed to establish the system for a new domain. The DARE system can learn rules for any relations or events automatically, given linguistically annotated documents and some initial examples of the relations. Furthermore, DARE can utilize the learned rules for extracting relations or events from free text documents. The DARE system has been applied to the event detection and opinion mining tasks in Alexandria. The biographic information is a relevant part of the Alexandria knowledge database. Thus, we have conducted a systematic analysis of the extraction performance of DARE on biographic information from various social domains of Wikipedia documents, namely, politicians, business people and entertainers[10]. Furthermore, we have adapted DARE to be able to deal with various parsers [1] and have proposed a novel strategy to adapt a deep parser to the domain-specific relation extraction task [20]. A deep parser delivers grammatical relations within a sentence. The DARE relation extraction machinery and its learned rules have been integrated into the Alexandria IE pipeline for the event detection and opinion mining tasks.

The remainder of the paper is structured as follows: section 2 gives a brief introduction of the DARE framework; section 3 to 4 give some brief descriptions of

experiments of DARE with various social domains and its interaction with parsers; section 5 describes the integration of DARE into the Alexandria IE pipeline and its applications to the event extraction and opinion mining tasks; section 6 draws some conclusions and presents some ideas for future research.

2 DARE Framework

DARE is a minimally supervised machine learning system for RE on free texts [19, 21]. It consists of two parts: 1) rule learning and 2) relation extraction (RE). Rule learning and RE feed each other in a bootstrapping framework. The bootstrapping starts from so-called "semantic seeds", which are small sets of instances of the target relation. The rules are extracted from sentences annotated with semantic entity types and linguistic parsing results, e.g., dependency structures, which match with the seeds. RE applies the acquired rules to texts in order to discover more relation instances, which in turn are employed as seeds for further iterations. The core system architecture of DARE is depicted in Fig. 1. The entire bootstrapping stops when no new rules or new instances can be detected. Relying entirely on semantic seeds as domain knowledge, DARE can accommodate new relation types and domains with minimal effort.

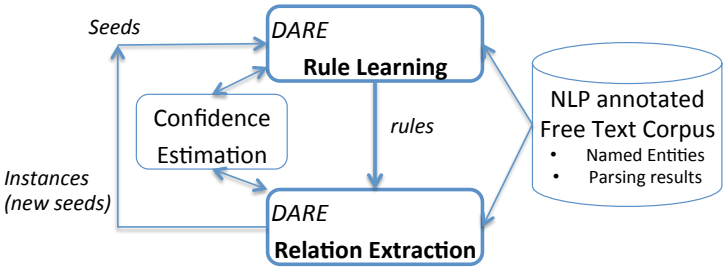

Fig. 1: DARE core architecture

The confidence values of the newly acquired rules and instances are calculated in the spirit of the "Duality principle" ([3], [5] and [22]), i.e., the confidence values of the rules are dependent on the truth value of their extracted instances and on the seed instances from which they stem. The confidence value of an extracted instance makes use of the confidence value of its ancestor seed instances.

DARE can handle target relations of varying arity through a compositional and recursive rule representation and a bottom-up rule discovery strategy. A DARE rule for an n-ary relation can be composed of rules for its projections, namely, rules that extract a subset of the n arguments. Furthermore, it defines explicitly the semantic roles of linguistic arguments for the target relation. The following examples illustrate the DARE rule and its extraction strategy. *Example 2.* is a relation instance of the target relation from [19] concerning Prize awarding event, which contains

four arguments: *Winner*, *Prize_Name*, *Prize_Area* and *Year*. *Example 2.* refers to an event mentioned in *Example 3.*

Example 2. <Mohamed ElBaradei, Nobel, Peace, 2005>.

Example 3. Mohamed ElBaradei, won the *2005 Nobel* Prize for *Peace* on Friday.

Given *Example 2.* as a seed, *Example 2.* matches with the sentence in *Example 3.* and DARE assigns the semantic roles known in the seed to the matched linguistic arguments in *Example 3*. Fig. 2. is a simplified dependency tree of *Example 3.* with named entity annotations and corresponding semantic role labelling after the match with the seed. DARE utilizes a bottom-up rule discovery strategy to extract rules from such semantic role labelled dependency trees.

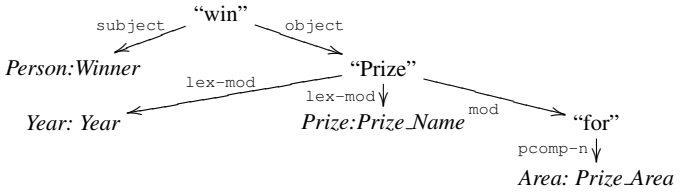

Fig. 2: Dependency tree of *Example 3.* matched with the seed

Rule name :: winner_prize_area_year_1
Rule body ::

$$\left[\begin{array}{ll} \text{head} & \left[\begin{array}{ll} \text{pos} & \text{verb} \\ \text{mode} & \text{active} \\ \text{lex-form} & \text{“win”} \end{array}\right] \\ \text{daughters} & < \left[\text{subject} \quad \left[\text{head} \quad \boxed{1}\textit{Person}\right]\right], \\ & \left[\text{object} \quad \left[\begin{array}{ll} \text{rule} & \text{year\_prize\_area\_1} :: \\ & < \boxed{4}\textit{Year}, \boxed{2}\textit{Prize\_Name}, \\ & \boxed{3}\textit{Prize\_Area} > \end{array}\right]\right] > \end{array}\right]$$

Output :: $< \boxed{1}\textit{Winner}, \boxed{2}\textit{Prize\_Name}, \boxed{3}\textit{Prize\_Area}, \boxed{4}\textit{Year} >$

Fig. 3: DARE extraction rule.

From the tree in Fig. 2, DARE learns three rules in a bottom-up manner, each step with a one tree depth. The first rule is extracted from the subtree dominated by the preposition "for", extracting the argument *Prize_Area* (*Area*), while the second rule makes use of the subtree dominated by the noun "Prize", extracting the arguments *Year* (*Year*) and *Prize_Name* (*Prize*), and calling the first rule for the argument *Prize_Area* (*Area*). The third rule "winner_prize_area_year_1" is depicted in Fig. 3. The value of *Rule body* is extracted from the dependency tree. In "winner_prize_area_year_1", the subject value *Person* fills the semantic role *Winner*. The object value calls internally the second rule called "year_prize_area_1", which handles the other arguments *Year* (*Year*), *Prize_Name* (*Prize*) and *Prize_Area* (*Area*).

Domain	Entertainer	Politician	Business Person
Number of documents	300	300	300
Size (MB)	4.8	6.8	1.6
Number of person occurrences	61450	63015	9441
Number of person entities	9054	6537	1652
Sentences containing person-person-relations	9876	11111	1174

Table 3: Data Properties of the three Domain Corpora

In [15], a more detailed analysis of the DARE framework has been conducted. The experiments show that data properties and the seed configuration play important roles for the learning and extraction performance. Furthermore, the majority of errors are caused by parsers.

3 Extraction of Biographic Information from Various Social Domains

In [10], a systematic data analysis has been conducted for three social domains from Wikipedia documents for the extraction of biographic information. The three domains are politicians, entertainers and business people. We extract for each domain 300 documents. For the entertainer domain, we choose pages about actors or actresses of the Oscar academy awards and grammy winners. Pages about the US presidents and other political leaders are selected for the politician domain. American chief executives covered by Wikipedia are candidates for the business people corpus. In Table 3, we show the distribution of persons, their occurrences and sentences referring to two persons. We immediately observe that the business people texts mention much fewer persons or relationships between persons than the texts on politicians. Most mentions of persons and relationships can be found in the entertainer texts so that we can expect to find more extraction rules there than in the other domains.

Table 4 presents all figures for the precision and number of correctly extracted instances for each domain and merged domains after the application of DARE to the collected documents. For each domain, the ten most prominent persons for each domain are selected. The prominence is defined by the length of their Wikipedia page. The average precision of the business person domain is the highest, while the entertainer domain extracts the most correct instances but with the lowest precision. The politician domain has neither good precision nor good extraction gain. We cannot provide the recall number, since there is no gold-standard corpus available for our experiments.

In order to improve the precision performance, we decided to select also negative seed examples to exclude rules which extract wrong instances. For the negative seed construction, we developed a new approach. For our target relation, a negative seed contains person pairs who do not stand in a marriage relation, but who are

Single domain	1 positive seed (each)	
	Precision	Extracted Correct Instances
Entertainer	5.9%	206
Politician	16.19%	159
Business Person	70.45%	31
Multiple domains	3 positive seed (merged)	
	Precision	Correct instances
merged corpus	8.91%	499

Table 4: Average values of 10 runs for each domain and 1 run for the merged corpus with best seeds

extracted by the top 20 ranked rules produced from positive seed. The learning of the negative rules works just like the learning of the positive ones, but without any iterations. Once we have obtained rules from negative examples, we only use them for subtracting any identical rules from the rule set learned from positive seed [10].

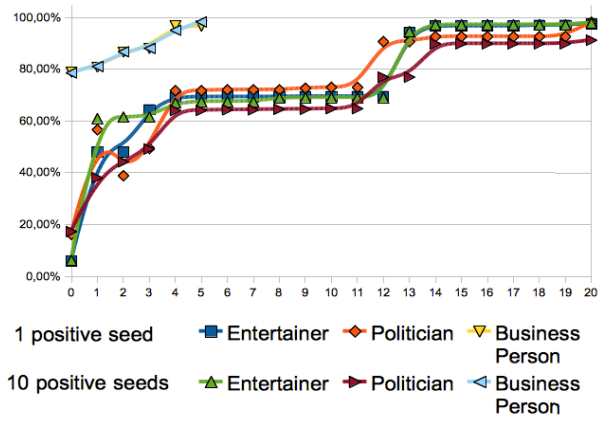

Fig. 4: Average precision of experiments in 3 domains with 1 or 10 positive seeds and 1 to 20 negative seeds: *x axis* for negative seed, *y axis* for precision

Figure 4 shows the improvement of precision after the utilization of negative seeds for 1 positive and 10 positive seed situations, while Figure 5 depicts the development of the extracted corrected instances. It appears that the number of positive seeds does not make a significant difference of the performance development. For the business people domain, only a few negative seeds suffice for getting 100% precision. For both entertainment and politician domains, the negative seeds considerably improve precision. There are several jumps in the curves. In the entertainment domain, the first negative seed removes the strongest bad rule. As a side-effect some good rules move upwards so that both precision and recall increase significantly and at the same time some other bad rules move downwards which are connected to subsequent negative seeds. Therefore, the second negative seed does not lead to big

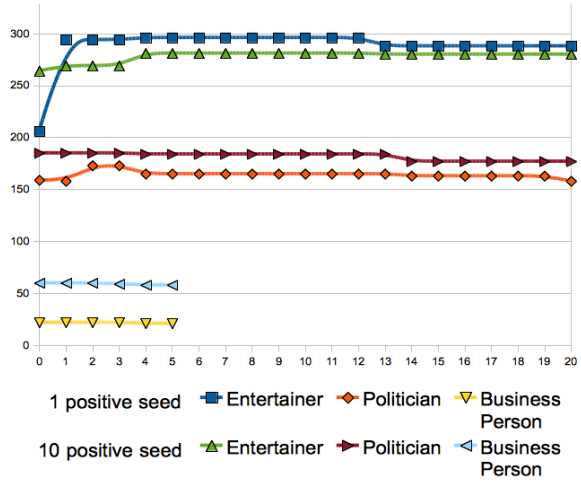

Fig. 5: Correct instances of experiments in 3 domains with 1 or 10 positive seeds and 1 to 20 negative seeds: *x axis* for negative seed, *y axis* for number of extracted correct instances

jump in the performance. Similar phenomena can be observed by analysing other flat portions of the curve.

The research results above give us the following insight: exploitation of data from a related but different domain for a domain that does not possess suitable learning data is very important in the real-world application, e.g., the entertainer domain is more suitable for learning rules about biographic information than the business people domain.

4 Interaction with Parsers

In the Alexandria IE pipeline, we have conducted experiments with various parsers. Since the parsing results play an essential role of the relation extraction performance, we have extended DARE so that it can deal with different parsing outputs. Furthermore, we have developed a parse re-ranking strategy which enables parsers to adapt to the domain-specific application tasks.

4.1 Dependency Graph as Linguistic Interface

The DARE framework as described in section 2 builds upon linguistic analyses in the form of dependency trees. However, not all linguistic structures can be faithfully described with trees. Grammars aiming at a deeper, more semantic level of analysis of a sentence – such as the Head-Driven Phrase Structure Grammar (HPSG) [13] –

rather use a general graph model as their formal foundation. Graph-based semantic representations are for instance used to encode the fact that the same expression can have several roles within a sentence. To give an example, a word can be both the object of the verb in the main clause and the subject of the verb in a relative clause, and graphs can be used to model that directly. In order to incorporate results delivered by deeper parsers, we therefore extended the original tree-based interface to a more general graph-based representation [1]. This allowed us to adapt our system to the output representations of various parsers, in particular to those containing more semantics.

A DARE graph rule has three components, as depicted in Fig. 3.

1. *rule name*: r_i
2. *output*: a set A containing n arguments of the n-ary relation, labelled with their argument roles.
3. *rule body*: a graph $G = (N,E)$ where N is a set of nodes with – possibly underspecified – features to be matched, and E is a set of – possibly labelled – edges connecting these nodes. The elements of A are coindexed with the reference feature of the corresponding argument nodes in N.

As before, the rule learning happens in two steps. Matching subgraphs are first extracted and then generalized to form extraction rules by underspecifying the nodes and introducing place-holders labelled with the role for the argument nodes. The pattern subgraphs are extracted from the dependency graph by the following procedure:

1. For a given n-ary seed $S = (s_1, \ldots, s_n)$ and a given dependency graph G, collect the set T of all terminal nodes from G that are instantiated with seed arguments in S.
2. For each acceptable combination of seed argument terminal nodes $C = \{t_1, \ldots, t_m\}$ ($m \geq 2$), find a shortest path S_i between t_i and t_{i+1} for $0 < i < m$.
3. For each combination of seed argument terminal nodes C and the corresponding set of shortest paths $S_C = \{S_1, \ldots, S_m\}$, extract the corresponding pattern subgraph P_C from G, where the set of nodes is the union of the nodes of S_i and the set of edges is the union of the edges of S_i ($0 < i < m$).

Three parsers have been evaluated with the new interface. They are 1) **MINIPAR** [11]: a broad-coverage parser for English whose parsing results are available in a dependency tree format; 2) the **Stanford Parser** [12], which provides dependency tree structures labelled with grammatical functions; 3) the **PET** parser [6] together with a broad-coverage grammar for English called ERG [7], written in the HPSG framework.

Our experiments confirm that the graph-based interface is expressive enough to allow learning extraction rules exhaustively from semantic analysis provided by various parsers [1]. As expected, switching to a graph representation for the parsers outputting dependency trees does not have any impact on the RE results. But using the graph-based representation for the extraction with deep HPSG analyses improves both recall and f-score of the RE (more than 1 point f-score improvement) and the arity of the extracted instances is higher, therefore, more information can be detected.

4.2 Parse Re-ranking

In our research, we observe that there is a strong connection between the RE task and the parser via the learned extraction rules, because these rules are derived from the parse readings. The confidence values of the extraction rules imply the domain appropriateness of the parse readings. Therefore, the confidence values can be utilized as feedback for the parser to help it re-ranking its readings. In [20], the generic HPSG parser PET with its grammar ERG has been adapted to the relation extraction task via parse re-ranking.

Figure 6 depicts the overall architecture of our experimental system. We utilize the HPSG to parse our experimental corpus and keep the first n readings of each sentence (e.g., 256) delivered by the parser. During bootstrapping, DARE tries to learn extraction rules from all readings of sentences containing a seed instance or newly detected instances. At each iteration, the extracted rules are applied to all readings of all sentences. When bootstrapping has terminated, the obtained rules are assigned confidence values based on the DARE ranking method described in section 2.

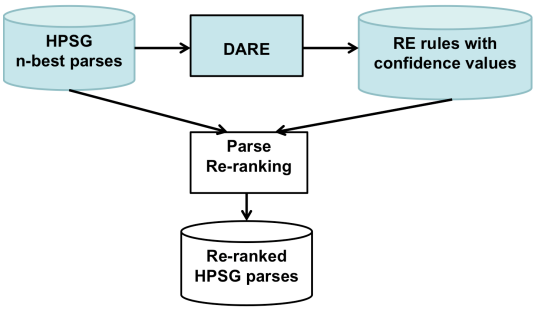

Fig. 6: DARE and Parse Re-ranking

The parse re-ranking component scores the alternative parses of each sentence based on the confidence values of the rules matching these parses, i.e., all rules that could have been extracted from a parse or successfully applied to it.

For each reading from the HPSG parser, the re-ranking model assigns a numeric score by the following formula:

$$S(t) = \begin{cases} \sum_{r \in R(t)} (\textbf{confidence}(r) - \phi\textbf{confidence}) & \text{if } R(t) \neq \phi \\ 0 & \text{if } R(t) = \phi \end{cases} \quad (1)$$

$R(t)$ is the set of RE rules matching parse reading t, and ϕ**confidence** is the average confidence score among all rules. The score of the reading will be increased if the matching rule has an above-average confidence score. And the matching of low-confidence rules will decrease the reading's re-ranking score. If a reading has no matching DARE rule, it will be assigned the lowest score 0, because no potential relation can be extracted from that reading. After the calculation, the top-n readings

are sorted in descending order. In case two or more readings received the same re-ranking score (e.g. by matching the same set of DARE rules), the original maximum entropy-based disambiguation scores are used as a tie-breaker.

In our experiments described in detailed in [20], we could show that for one generic parser/grammar, recall and f-measure could be considerably improved and hope that this effect can also be obtained for other generic parsers.

5 Integration of DARE in Alexandria IE Platform

5.1 Overview

DARE is employed in the Alexandria IE pipeline as an IE module with deep analytic capabilities. This computational power is required by the semantic relations we want to recognize – mentioned events, together with their characterizing properties, and expressed opinions. Both targeted relation types strongly benefit from a more detailed analysis of the expressed meaning, in order to capture subtle meaning differences that cannot be covered by mainstream statistical bag-of-words approaches. Both topics will be detailed in subsection 5.2 and 5.3, respectively.

The technical integration of DARE follows the architectural design decisions of the Alexandria platform. The IE pipeline uses Apache's open source UIMA framework[1] as a middleware for integrating the input and processing results of all NLP components. UIMA provides a general framework i) for describing, representing and storing rich stand-off annotations of data, ii) for implementing analytical components as well as for modelling and running pipelines on top of these components, and iii) graphical tools for running components and pipelines and inspecting results. For each task, we create a specific UIMA module, which wraps the DARE system and provides the required input and output interfaces. Each module is accompanied by a formal type system, describing the kind of annotations the module is consuming and producing.

5.2 Event Identification and Extraction

In order to systematically detect common types of events, we analyzed a newspaper corpus of 50,000 documents prepared by Neofonie. Every document in this corpus belongs to one of the following news sections: *Front Page, Politics, Economy, Health, Internet, Culture, Science, Sports, Automobiles & Technology, Local, Media, Travel, and Miscellaneous*. Using the tf-idf weight, we determined for each section the relevant terms. From the terms with the highest tf-idf frequency in each domain, we manually identified the 9 event types shown in Table 5. The relevant

[1] `http://uima.apache.org/`, as accessed on 5th March 2012

terms with high frequency for the corresponding event types are stored in a typed gazetteer, which serves as a resource for a finding cues for mentioned events in the documents.

event type	gazetteer size	examples
election	24	Direktwahl,Präsidentschaftswahl, Parlamentswahl
conference	83	Euro-Gipfel, Bundespressekonferenz
scandal	67	Abhörskandal,Bestechungsskandal,Abhöraffäre
crisis	77	Schuldenkrise, EU-Krise
social movement	8	Studentenunruhen, Aufstandsbewegung
disaster	23	Atomkatastrophe, Atom-GAU, Taifun, Bombenexplosion
festival	51	Filmfestival, Musikfestival
terror	8	Bombenanschlag, Terrorattacke
others	4	Wiedervereinigung, Party

Table 5: event types and the gazetteer examples

The recognition of event features (e.g. date, place, participants) is realized by semi-supervised learning of relation extraction rules using DARE. First we extracted event entities with the typed gazetteer. Then we constructed start seeds for DARE with a general event name, e.g., "parliamentary elections" (DE: "Bundestagswahl") and some possible named entities of different types (e.g. Person, Location, Organization) to learn extraction rules in DARE. With all the seeds created in this way, we used DARE to learn all sorts of rules between named entities and event names. We used further semi-automatic control methods to filter all learned rules according to their relevance and plausibility, and determined the precise roles of the (person or organization) participants in the relation. For example, from the sentence

> Die neue IWF-Chefin Christine Lagarde wird am Donnerstag in Brüssel an einem Sondergipfel zur Schuldenkrise in der Euro-Zone teilnehmen.

we can learn the following DARE rule (here shown in a shortened format):

```
"werden" { SB( PERSON:participant ),
           OC("teilnehmen") { MO ("an") {PNK( EVENT: event )} } }
```

, where the EVENT and PERSON are NE types. With this rule, we can extract the participants of an event. The learned rules can be used to identify new event terms and their participants, which are not defined in the gazetteer. For instance, we can update the rule above by replacing the condition $type = EVENT$ with $POS = Noun$

```
"werden" { SB( PERSON:participant ),
           OC("teilnehmen") { MO ("an") {PNK( Noun: event )} } }
```

5.3 Opinion Mining

An important application in Alexandria is the automatic monitoring of news for the identification of published opinions about politicians. Reported opinions are not only recognized but the sentiment is also classified as positive, negative or neutral. Using this technology, the public image of a person, organization or brand can be tracked over time and general trends can be recognized. Opinion and Trend Mining are therefore crucial instruments for journalistic exploration, business intelligence, or reputation management.

We have applied DARE to learn relation extraction rules that identify the source of the opinion, the target of the opinion and the polarity of the sentiment. The rules are learned from an annotated newspaper corpus described in [2]. Concerning the expression level annotation, the most relevant and well-known work is the Multi-Perspective Question Answering (MPQA) corpus [16, 17, 18]. This work presents a schema for annotating expressions of opinions, sentiments, and other private states in newspaper articles. To our knowledge, there is only one German corpus available that is annotated with opinion relevant information, which is in the domain of user-generated reviews [14]. In comparison to product reviews, opinions in newspaper articles are in general expressed with more subtle means.

Example 4a) and b) are the headlines of two different news articles that tell the same event that German chancellor Merkel has obtained the medal of freedom by the president Obama. The two authors used two different verbs: 4a) with the verb *ehren* (engl. honor) and 4b) with the verb *überreicht* (engl. hands over). The first one is strongly positive, while the second one is more or less neutral.

Example 4.
a) US-Präsident Obama wird die Kanzlerin mit einer Auszeichnung **ehren**.
(Engl.: The US-president Obama will honor the chancellor an award.)

b) US-Präsident Barack Obama **überreicht** *der Kanzlerin die "Medal of Freedom".*
(Engl.: US-president Barack Obama hands over the "Medal of Freedom" to the chancellor.)

Our annotation schema is inspired by MPQA and its major frame is describe in the following table.

With the help of the annotated corpus, we are able to learn opinion extraction rules, e.g.:

Example 5.
a) verb (kritisieren, negative): subject(source), object(target)
(Engl.: verb (criticize, negative): subject(source), object(target))

b) verb (angreifen, negative): subject(source), object(target)
(Engl.: verb (attack, negative): subject(source), object(target))

Element	**Property**
Target	
Source	
Text anchor	isaIdiom, isaPhrase, isaWord isaCompoundNoun
Polarity	positive, negative, neutral
Auxiliary	isaNegation, isaIntensifier, isaDiminisher

Table 6: Opinion Frame elements

In comparison to English texts, we are faced with the challenges of German particle verbs and complex noun compounds. Furthermore, idioms and collocations are very popular by German newspapers. We utilize the linguistic annotation delivered by the Alexandria IE pipeline. The annotation result is a valuable resource for the opinion mining research for German language, in particular, the German political news. The distinction between context-dependent and context-independent frames is important for the estimation of the need of world and domain knowledge for a running system.

6 Conclusion and Future Ideas

DARE is utilized for the THESEUS use case Alexandria for the information extraction task, both for automatic rule learning and extraction of facts and opinions from German news texts. The realistic tasks and data of Alexandria provided a challenge to the IE framework that is representative of many future applications for the exploitation and transformation of unstructured Web content. The experiments with various social domains give us very interesting insights and let us better understand the challenges and opportunities behind the typical coverage of domain information in the news. The application of the DARE framework to opinion detection yielded truly promising results. The experiments with various parsers show that robust semi-deep parsers such as the widely used Stanford Parser are still the analysis instrument of choice if the main performance criterion for the relation extraction is recall or f-measure. Only in cases in which precision needs to be guaranteed, a deeper and less robust parser working with intellectually created grammars can outperform the shallower data-driven analysis. But the performance of deep parsers for relation extraction can be considerably improved through domain-guided re-ranking, another form of domain adaption. A side result of this work was the insight that a better parse ranking for the purpose of relation extraction does not necessarily correspond to a better parse ranking for other purposes or for generic parsing. This should not be surprising since relation extraction in contrast to text understanding does not need the entire and correct syntactic structure for the detection of relation instances. The ease and consistency of rule extraction and rule application counts more than the

linguistically correct analysis. With the evolution of parsing technology the contribution of deep parsers may further improve.

The work on adapting the relation extraction system to the domains and tasks of Alexandria has also served as a source of ideas for future research. The insights gained by the investigation of the web-based data suggested to train the system directly on the web without any bootstrapping in cases where large amounts of seed data can be obtained. Indeed, for several Alexandria-relevant domains including the domain of celebrities and other people, sufficient resources of massive seeds could be identified. After the successful conclusion of the Alexandria work, subsequent research on direct training of relation extraction by finding the patterns for rules in web content has already been started [9].

Acknowledgements The extensions of the DARE framework have been conducted in cooperations between the THESEUS Alexandria use case and the following other research projects: the German DFG Cluster of Excellence on Multimodal Computing and Interaction (M2CI) and the project TAKE (funded by the German Federal Ministry of Education and Research, contract 01IW08003). Many thanks go to Yi Zhang for his joint work for the parse re-ranking approach.

References

1. Adolphs, P., Xu, F., Li, H., Uszkoreit, H.: Dependency graphs as a generic interface between parsers and relation extraction rule learning. In: KI 2011: Advances in Artificial Intelligence, 34th Annual German Conference on AI, Berlin, Germany, October 4-7,2011. Proceedings, *Lecture Notes in Computer Science*, vol. 7006, pp. 50–62. Springer (2011)
2. Adson, K., Li, H., Kirshboim, T., Cheng, X., Xu, F.: Annotating opinions in german political news. In: Proceedings of LREC 2012 (accepted) (2012)
3. Agichtein, E., Gravano, L.: Snowball: Extracting relations from large plain-text collections. In: Proceedings of the 5th ACM International Conference DL'00. San Antonio, TX (2000)
4. Appelt, D., Israel, D.: Introduction to information extraction technology. IJCAI-99 Tutorial (1999)
5. Brin, S.: Extracting patterns and relations from the world wide web. In: WebDB Workshop at EDBT 98 (1998). URL `citeseer.nj.nec.com/brin98extracting.html`
6. Callmeier, U.: Preprocessing and encoding techniques in pet. In: S. Oepen, D. Flickinger, J. ichi Tsujii, H. Uszkoreit (eds.) Collaborative Language Engineering. A Case Study in Efficient Grammar-based Processing, pp. –. CSLI Publications (2002)
7. Flickinger, D.: On building a more efficient grammar by exploiting types. Natural Language Engineering **6**(1), 15–28 (2000)
8. Grishman, R., Sundheim, B.: Message understanding conference - 6: A brief history. In: Proceedings of the 16th International Conference on Computational Linguistics. Copenhagen (1996)
9. Krause, S., Li, H., Uszkoreit, H., Xu, F.: Large-scale learning of relation-extraction rules with distant supervision from the web. In: Proceedings of the 11th International Semantic Web Conference. Springer (2012)
10. Li, H., Xu, F., Uszkoreit, H.: Minimally supervised rule learning for the extraction of biographic information from various social domains. In: Proceedings of the International Conference Recent Advances in Natural Language Processing 2011, pp. 17–24. RANLP 2011 Organising Committee, Hissar, Bulgaria (2011). URL `http://aclweb.org/anthology/R11-1003`

11. Lin, D.: Dependency-based evaluation of MINIPAR. In: A. Abeillé (ed.) Treebanks - Building and Using Parsed Corpora. Kluwer Academic Publishers (2003)
12. de Marneffe, M., Maccartney, B., Manning, C.: Generating typed dependency parses from phrase structure parses. In: Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC 2006). Genoa, Italy (2006)
13. Pollard, C.J., Sag, I.A.: Head-Driven Phrase Structure Grammar. University of Chicago Press, Chicago, USA (1994)
14. Schulz, J.M., Womser-Hacker, C., Mandl, T.: Multilingual corpus development for opinion mining. In: N.C.C. Chair), K. Choukri, B. Maegaard, J. Mariani, J. Odijk, S. Piperidis, M. Rosner, D. Tapias (eds.) Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'10). European Language Resources Association (ELRA), Valletta, Malta (2010)
15. Uszkoreit, H.: Learning relation extraction grammars with minimal human intervention: strategy, results, insights and plans. In: Proceedings of the 12th International Conference on Intelligent Text Processing and Computational Linguistics, *Lecture Notes in Computer Science*, vol. 6609, pp. 106–126. Springer (2011)
16. Wiebe, J., Wilson, T., Cardie, C.: Annotating expressions of opinions and emotions in language. Language Resources and Evaluation **39(2/3)**, 165–210 (2005)
17. Wilson, T., Wiebe, J.: Annotating attributions and private states. In: Proceedings of the Workshop on Frontiers in Corpus Annotations II: Pie in the Sky, pp. 53–60. Ann Arbor, Michigan (2005)
18. Wilson, T.A.: Fine-grained subjectivity and sentiment analysis: Recognizing the intensity, polarity, and attitudes of private states. In: PhD Thesis. University of Pittsburgh (2008)
19. Xu, F.: Bootstrapping relation extraction from semantic seeds. Phd-thesis, Saarland University (2007)
20. Xu, F., Li, H., Zhang, Y., Uszkoreit, H., Krause, S.: Minimally supervised domain-adaptive parse reranking for relation extraction. In: Proceedings of the 12th International Conference on Parsing Technologies, pp. 118–128. Association for Computational Linguistics, Dublin, Ireland (2011). URL `http://www.aclweb.org/anthology/W11-2915`
21. Xu, F., Uszkoreit, H., Li, H.: A seed-driven bottom-up machine learning framework for extracting relations of various complexity. In: Proceedings of ACL 2007. Prague, Czech Republic (2007)
22. Yangarber, R.: Scenarion customization for information extraction. Dissertation, Department of Computer Science, New York University, New York, USA (2001)