Semantic Content Access using Domain-Independent NLP Ontologies

René Witte[1] and Ralf Krestel[2]

[1] Semantic Software Lab, Concordia University, Montréal, Canada
[2] L3S Research Center, University of Hannover, Germany

Abstract. We present a lightweight, user-centred approach for document navigation and analysis that is based on an ontology of text mining results. This allows us to bring the result of existing text mining pipelines directly to end users. Our approach is domain-independent and relies on existing NLP analysis tasks such as automatic multi-document summarization, clustering, question-answering, and opinion mining. Users can interactively trigger semantic processing services for tasks such as analyzing product reviews, daily news, or other document sets.

1 Introduction

Despite recent advances in semantic processing, users continue to be overloaded with information during everyday activities: Browsing product reviews on popular e-commerce websites, for example, can result in hundreds (or thousands) of user-generated reviews, which take considerable time to read and analyse for their relevance, informativeness, and opinion. Thanks to the success of the Web 2.0, the combined length of the user-generated reviews of a single, popular book can now exceed the length of the book itself—which defies the goal of reviews to save time when deciding whether to read the book itself or not. While some condensed views are often available (like a "star rating"), this in turn is usually too compressed to be used by itself.

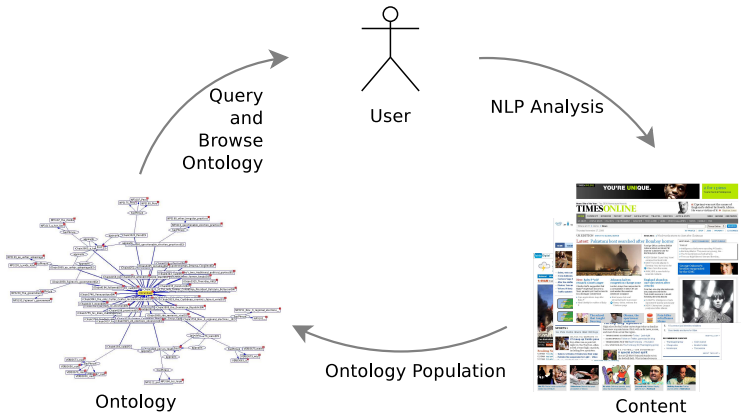
Similar challenges are faced by everybody dealing with (natural language) content. Writers of reports or research papers need to survey extensive amounts of existing documents; email and other communication forms take up significant amounts of time, with no integrated semantic processing support available that could ease the analysis of questions or composition of answers while writing them. Both in research and industry, employees spend an ever increasing proportion of their time searching for the right information. Information overflow has become a serious threat to productivity.

In this paper, we focus on *user-driven NLP* for managing large amounts of textual information. That is, NLP analysis is explicitly requested by a user for a certain task at hand, not pre-computed on a server. To access the results of these analyses, we propose an NLP ontology that focuses on the *tasks* (like summarization or opinion analysis), rather than the domain (like news, biology, or software engineering).

Background. In recent years, the fields of natural language processing and text mining have developed a number of robust analysis techniques that can help users in information analysis and content development: automatic summarization [1, 2], question-answering [3] and opinion mining [4] can be directly applied to the scenarios outlined above.

Text analysis as described above is typically done within a component-based framework such as GATE [5] or UIMA [6]. However, these frameworks are targeted at language engineers, not end users. So far, none of the text mining methods described above have become available to a mass user base. To bring NLP directly to an end user's desktop, we previously developed *Semantic Assistants* [7], a service-oriented architecture that brokers the NLP pipelines through W3C Web services with WSDL descriptions directly to desktop applications (like word processors or email clients). After solving the technical integration, we can now focus on the *semantic integration* of NLP into end users' tasks.

Proposed Approach. In this work, we focus on semantic NLP services that can support users in common, yet time-consuming tasks, like browsing product reviews or analysing daily news. We argue that this can be achieved by building an ontology that integrates original content with the results of text mining pipelines. The following diagram illustrates our idea:



A user is faced with a large amount of natural language content—for example, hundreds of reviews for a single product on Amazon, or a cluster of thousands of news articles for a single event in Google News. Rather than dealing with this huge amount of text manually, the user triggers an NLP analysis of the document set. The results of the analysis is captured in a rich NLP ontology that now contains detected topics, summaries, contrastive information, answers to questions the user might have submitted to the NLP analysis, and also links back to the source documents. The user can now browse, query, or navigate the information through the highly structured populated ontology, and also follow links back to the source documents when needed. This approach empowers users by providing them with sophisticated text mining services, which both save time and deliver a richer information structure: for example, rather than reading

Table 1. Main concepts in the NLP ontology and their definition

Concept	Definition
Document	Set of source URIs containing information in natural language (e.g., news articles, product reviews, blog posts)
Content	Natural language text appearing either in a source document or generated as a result from text mining pipelines
DocContent	Natural language text appearing in a source document
Summary	NLP analysis artifact derived through applying specific algorithms to a set of input documents with optional contextual information
SingleSummary	An automatically generated summary of a single input document
ShortSumm	A keyphrase-like automatically generated summary indicating the major topics of a single document
Classical SingleSumm	An essay-like text of user-configurable length containing the most salient information of the source document
MultiSummary	An automatically generated summary of a set of input documents
ClassicalMultiSumm	An essay-like text of user-configurable length containing the most important (common) topics appearing in all source documents
FocusedSumm	An essay-like text of user-configurable length that addresses a specific user context (e.g., concrete questions the user needs to be addressed by the summary, or another reference document in order to find related content)
ConstrastiveSumm	Multi-document summarization method that generates (a) the commonalities (shared topics) across all input documents and (b) content specific to a single or subset of documents (contrasts)
Chain	Single- or cross-document coreference chain (NLP analysis artifact)
Chunk	Specific content fragments generated or manipulated by NLP analysis (e.g., noun phrases, verb groups, sentences)

just a few of hundreds of product reviews in order to reach a conclusion, the user can now see an automatically generated summary of the most important comments common to all reviews, and also directly see contrastive information (like disagreements), represented by specific ontology classes.

2 Design

The central goal of this work is to provide a semantically rich yet dense representation of large amounts of textual information that allows novel way of accessing content. Users should be able to see a highly summarized top-level view, but also be able to "drill-down" into specific aspects of the analyzed information. For example, a large number of product reviews could be summarized to a few sentences that contain information shared by the majority of individual entries. In addition, the major differences should also be analyzed and presented in a similar manner, allowing a user to detect the major diverging views without having to go through each individual review. Nevertheless, each of these summarized views should provide links to trace the analysis results back to their source statements. Essentially, we enrich content that is already available with automatically generated meta-information, thereby bringing the results of sophisticated text mining techniques to an end user, which is a significant improvement over current browsing methods, such as simple tag clouds.

Ontology Population. Such a semantically rich representation requires a suitable data model that can be queried, transformed, visualized, and exchanged between

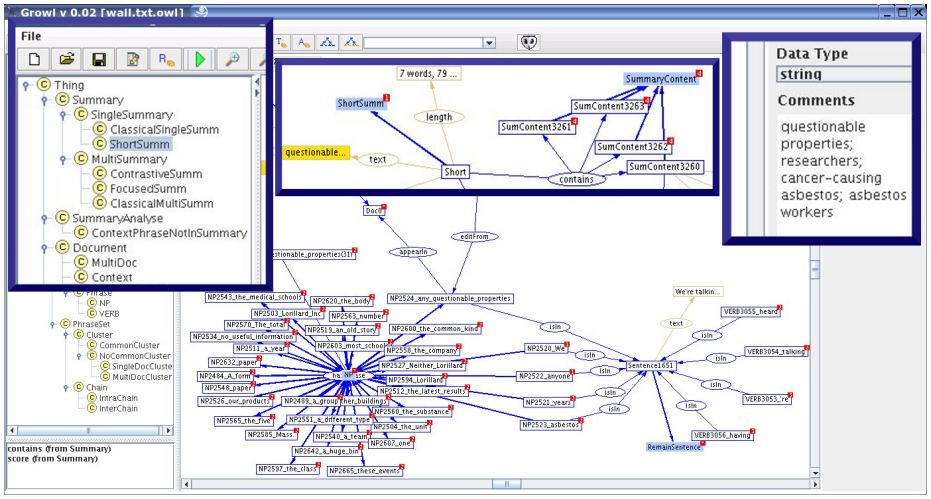


Fig. 1. An NLP-generated 10-word topic summary (right) of a single newspaper article represented in the NLP ontology

multiple applications. The solution presented here is to provide a generic ontology for *natural language processing results* that can then be automatically populated with the concrete results (OWL individuals) of a particular analysis task (e.g., all the reviews for one product). This now allows executing the complete semantic toolchain on NLP analysis results, including (SPARQL) queries, OWL-DL reasoning, and visualization, which is a significant improvement compared to the static XML result formats typically employed in today's text mining frameworks.

NLP Ontology. To facilitate the outlined document analysis tasks, we developed an NLP ontology. The domain of discourse for this ontology comprises the artifacts involved in automatic document analysis—texts and their constituents (like sentences, noun phrases, words) and the results of specific analysis pipelines, like the various types of summaries outlined above.

Table 1 shows the main concepts of our NLP ontology, together with a brief definition. Its main goal is to facilitate *content* access. Content individuals are (useful) snippets of information, which a user can read while performing his tasks (e.g., analysing product reviews). This can be content that appears in a source text, like a Web page (DocContent), or some text has been generated by a summarization algorithm (SummaryContent). To provide support for drilling down into NLP analysis results, original documents that form the basis for analysis are modeled explicitly as well. The ontology further distinguishes between a single document, like a single Web page (SingleDoc), a document collection (MultiDoc) and contextual information (Context). The main NLP result artifact modeled in our ontology are *summaries*, including all the various types discussed in the introduction.

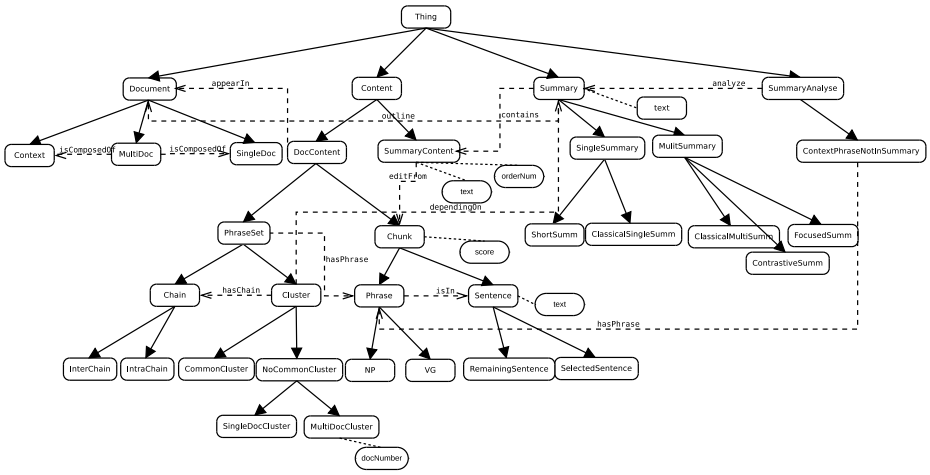


Fig. 2. Domain-Independent NLP Ontology Model

To allow further drill-down into the NLP analysis of a given document (set), we also model further important NLP concepts such as sentences, (phrase) chunks, coreference chains, and entity clusters. These provide for the navigational links between original documents and derived or generated content in the generated summaries—see Fig. 1 for an example of a populated ontology.

Fig. 2 shows an overview of the main concepts and relations in the resulting NLP ontology. For deployment, this ontology needs to be automatically populated with concrete analysis results (described in Section 3), and can then be used to facilitate content access across various domains and tasks (Section 4).

3 Implementation

Our implementation is based on the Semantic Assistants design [7], which allows bringing the results of NLP analysis to a connected desktop client (like a word processor or a Web browser) through semantic Web services. Natural language processing is performed within the GATE framework [5]. The concrete analysis pipelines deployed here, i.e., for summary generation and contrast analysis, are based on our previous work in automatic summarization [8]. Fig. 3 shows the processing pipeline with its various components. After some NLP processing (preprocessing, part-of-speech tagging, ...) the task specific application is run – in this case a summarization component. The visualization component then populates the ontology described in Section 2 with the NLP results and exports the ontology using the OWL format.

Population of the ontology is achieved through a custom processing component in GATE, the OwlExporter [9], which combines the results from the analysis pipelines with the loaded documents, thereby enabling the combined view of original content with NLP-discovered analysis results, such as contrastive

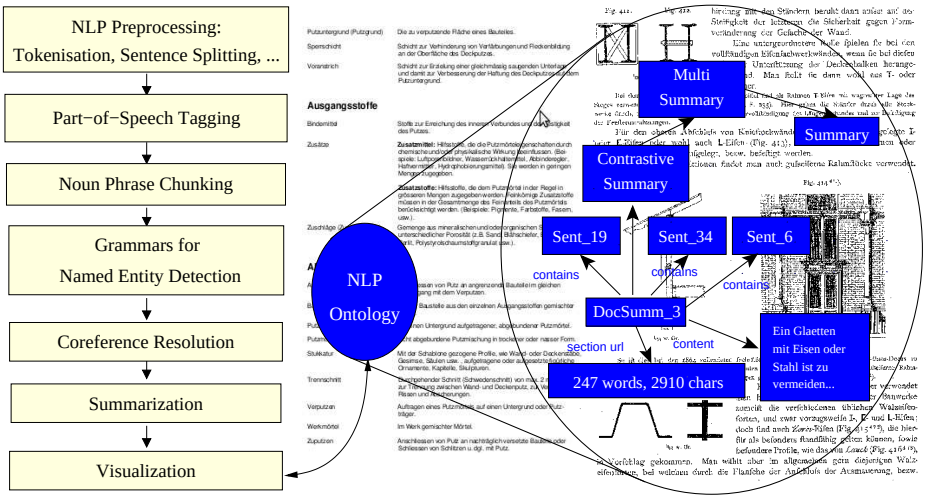


Fig. 3. Pipeline for populating the NLP ontology using text mining results

information. This ontology is rich enough to allow connected clients to present various views, depending on the output capabilities and the users' current need and context. For example, a Web browser plugin could directly annotate Web pages with the analysis results, while a small-screen device, such as a mobile phone, would need to present a more compressed view.

4 Applications

In this section, we evaluate the applicability of our ideas on three concrete scenarios: analyzing daily news, heritage documents, and product reviews. All three scenarios highlight the need for automated semantic support that so far has not been available to an end user.

Workflow. In all these scenarios, we employ the text mining pipelines discussed in Section 3. The documents for analysis, as well as relevant information about the user context—such as concrete questions or viewpoints the user needs to have addressed by the semantic services—is transmitted from the client to the NLP framework via the Semantic Assistants architecture described in [7]. The results of the analysis pipelines is captured in the OWL ontology described in Section 2 using a custom ontology population components as described in Section 3. This ontology is then transmitted back to the client.

Information Visualization. The product delivered to an end user is a populated OWL ontology that captures a number of NLP analysis results (topics, summaries, answers to explicit questions, contrastive information, etc.). How an end user interacts with this result ontology is an ongoing challenge. Within this work, we

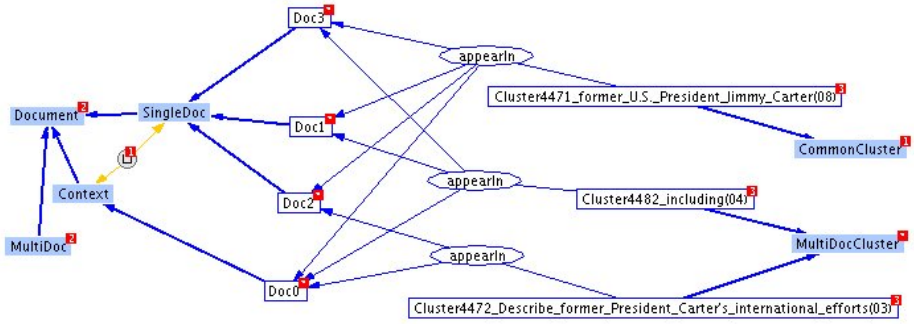


Fig. 4. Part of a populated NLP ontology showing topic clusters that are shared by a set of news documents (CommonCluster) and topic clusters that are specific to a subset of news articles (MultiDocCluster)

demonstrate the feasibility using standard ontology tools for browsing (SWOOP),[3] visualizing (GrOWL),[4] and querying (SPARQL)[5] the ontology. However, these require some level of understanding about OWL ontologies, which should not be presumed of an end user. Advanced query and visualization paradigms (e.g., graphical queries, NL queries) are a highly active area of research and the results from these efforts can be directly leveraged by our framework.

4.1 News Analysis

Despite the surge of user-generated content such as blogs and wikis, traditional newspaper and newswire texts continue to be an important source of daily news. News aggregators such as Google News[6] provide a condensed overview of current topics and the articles covering them. A number of research prototypes such as Newsblaster[7] or NewsExplorer[8] apply information extraction and summarization on the daily news to provide semantic query and navigation facilities. Compared with these server-side, precomputed content access options, our work provides semantic analysis "on demand" triggered by a user. This kind of analysis can be executed on a specific document set as selected by a user, and also be based on additional context information, e.g., a particular question a user needs to have addressed.

Topic analysis and summarization can augment a list of news articles with a brief list of topics (see Fig. 1, right side, for an example) and a summary of the main content of the article. This analysis aims to help a user in deciding whether

[3] SWOOP Ontology Browser/Editor, http://code.google.com/p/swoop/
[4] GrOWL ontology visualiser, http://ecoinformatics.uvm.edu/dmaps/growl
[5] SPARQL RDF query language, http://www.w3.org/TR/rdf-sparql-query/
[6] Google News, http://news.google.com
[7] Columbia Newsblaster, http://newsblaster.cs.columbia.edu/
[8] EMM NewsExplorer, http://press.jrc.it/NewsExplorer

he wants to read a specific article in full or not. However, our semantic analysis services can provide additional result ontologies when applied on a document set: multi-document summaries can extract the major common topics across a set of documents; contrastive summaries can detect differences between documents in a set, and focused summaries can extract information related to a user's current context—e.g., a concrete question or another document he is working on, like an email or a report.

Example Use Case. A user is faced with a large set of news relating to a single event. In order to find the topics shared by all news, he initiates an NLP analysis for topic detection and common multi-document summarization, which provides him with a list of topics and a summary (of adjustable length). While it is often sufficient to summarize the commonalities, the user might also be interested in specific differences between the news: for example, when the same event is reported differently across North American, European, Asian, and Middle Eastern newspapers. Finding such differences is possible with contrastive summarization [8], which detects topic clusters that only appear in a single or subset of documents. Using the populated NLP ontology (see Fig. 4), the user can navigate directly to such topic clusters, and then also see an automatically generated summary containing these differences (see [8] for some examples).

4.2 Heritage Data Analysis

In addition to analyzing current news, blogs, or web pages, heritage data can provide a rich source of information. This data is usually not easily accessible on a semantic level. Outdated terms, different styles of writing, or just the huge amount of heritage data makes it impossible for experts to exploit this knowledge satisfactorily nowadays. In this context, contrasting the heritage data with current data and make these differences visible and browsable is a big asset.

Example User Case. We applied our contrast summarization framework to an old, German encyclopedia of architecture from the 19th century [10] and compared it to present-day Swiss construction regulations. Practitioners, like building historians or architects, can navigate through the heritage data based on extracted contrasts between the historic knowledge and the state-of-the-art building engineering regulations. Fig. 3 shows a page of the old encyclopedia together with a contemporary standards document.

As with the news analysis described above, the populated ontology enables the user to find contrasts for particular methods or materials and browse through the source text of the found information.

Not only the different styles and formats makes it difficult for architects/historians to compare the heritage data with new data, but also the huge amount of available information. The populated ontology enables the user to find contrasts for particular engineering methods or building materials and browse through the source text of the found information.

4.3 Analyzing Product Reviews

E-commerce websites such as Amazon have long integrated user-contributed content in form of wikis and product reviews. These can provide important information to both buyer and vendor. However, when it takes longer to read all reviews for a book than the actual book itself, they are no longer a viable means for saving time.

In this application scenario, a user would delegate the analysis of a large set of reviews to an NLP service that analyses them for commonalities and differences. In the resulting ontology, the main topics in agreement will be detected and supplemented by a single summary. In addition, the contrastive cluster analysis will detect topics that only appear in a single or subset of reviews, allowing a semantic navigation of reviews based on content, rather then simple structure or surface feature as they are common today.

Example Use Case. In this example, a user obtained a number of product reviews for a computer science book in order to help in a purchase decision. Rather than reading all the reviews manually, the user invokes an NLP analysis of the obtained reviews and receives the populated ontology. Fig. 5 shows an excerpt of such a large, populated ontology summarizing multiple book reviews of a single book. As can be seen, contrastive statements can be easily identified and using the ontology a deeper analysis of the review content is possible. From this ontology, the user can navigate to the generated summary of the common opinions, as shown in Fig. 6. However, not all reviews share the positive outlook: some of diverging opinions can be found as OWL individuals in the ContrastiveSumm class,

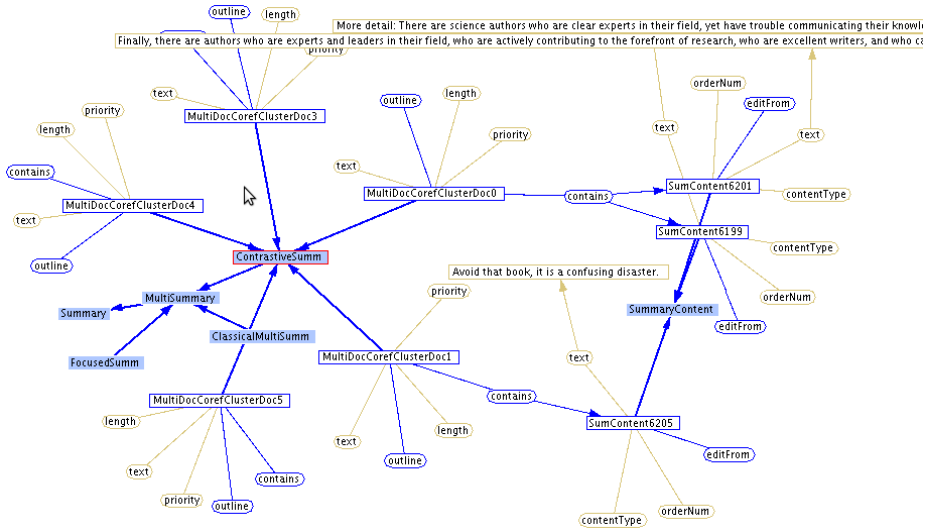


Fig. 5. Visualization of an Excerpt of an Ontology for Book Reviews

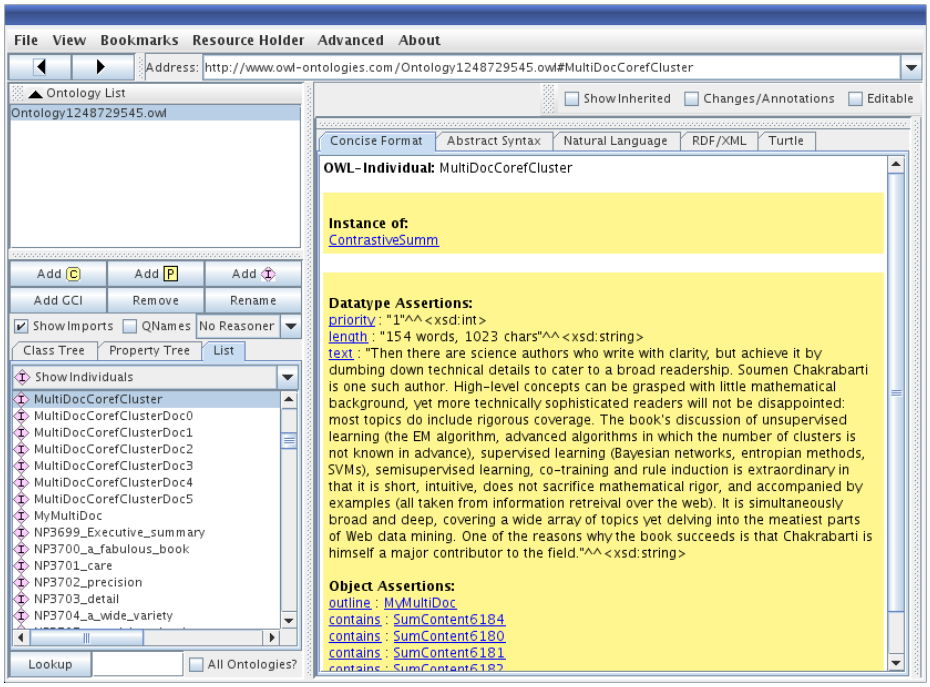


Fig. 6. An automatically generated summary of the commonalities of all the reviews of a book in the NLP ontology

as shown in Fig. 5 (in particular the text in the middle of the figure). As before, each of the generated summaries and concepts can be traced back to sentences in the original reviews, in case a user needs to understand the results of the automated NLP analysis in their original context.

5 Related Work

Using ontologies to facilitate the access to natural language documents was explored before. In [11], the authors develop an ontology for linguistic concepts to ease the sharing of annotated linguistic data and provides for searching and browsing of inhomogeneous corpora. The overall goal was to develop a way to conserve the rich linguistic concepts of (endangered) languages. The focus was therefore on creating an ontology that can deal with dynamic, changing data and with different source material.

The authors of [12] demonstrate the benefits of view-based search methods for RDF(S) repositories including semantic recommendations. The data is represented within an ontology and the user can refine his query by browsing through the views. This combines ontology-based search with multi-faceted search and enables the user to find information beyond keyword-based search results.

Ontology population from linguistic extractions is also the main focus of [13]. Knowledge acquisition rules are used to map concept tree nodes to ontology instances. The nodes are the result of extraction and annotation of documents. In particular the paper describes how to connect two components to work together in one framework: One for modeling domain knowledge using ontological concepts and one for linguistic extractions. They present their results in the legal domain and show how an ontology can be populated from annotated documents.

In [14] one way for a semantic representation of natural language documents is described. The presented system is capable of outputting its internal semantic representation of a document to the OWL format, allowing a semantic motivated browsing of the textual data. Also the possibilities of multiple agents are described, exchanging semantic information based on RDF or OWL.

The authors of [15] argue that ontologies need a linguistic grounding. RDFS or OWL offer not enough support for adding linguistic information like part-of-speech or subcategorization frames. They present a new model to associate ontological representations with linguistic information.

In contrast to these works, our approach focuses on the end-user and allows visualizing specific NLP analysis results. Also, we base our approach on user-initiated semantic annotation, whereas other systems use server-side, standard extractions to modeled documents using ontologies.

6 Conclusions and Future Work

In this paper, we presented a novel ontology-based approach for semantic document navigation. Accessing unstructured content is facilitated by modeling the results of general-purpose NLP algorithms, in particular various forms of automatic summarization, in a domain-independent ontology. We demonstrated the feasibility with a complete implementation including NLP analysis pipelines and applied it to a number of concrete application scenarios, including news analysis, cultural heritage data management, and product reviews. However, our approach is not limited to these examples; many tasks require a comparative study of a document set: applied to paper reviews in a conference system, contrastive summarization can help a conference chair find agreements and diverging views. In collaborative editing environments, like a wiki, a structured view that highlights the semantic differences between different versions can greatly facilitate the work of a maintainer—e.g., applied to Wikipedia articles where strong disagreements often lead to "edit wars."

A particular feature of our approach is that is user-driven: rather than relying on existing, pre-computed semantic annotations we envision a user that is supported by semantic analysis services in his tasks. NLP-driven analysis pipelines are executed on demand and the result ontology can be used for further document navigation, content access, or solving specific tasks.

Future work is specifically needed in two areas: first, more user-friendly ways of presenting the analysis results to the user that hide the OWL-specific implementation details. This can be achieved with client-specific plugins that

provide user-friendly ontology browsing and querying facilities. And second, user studies that evaluate the effect of the provided semantic support on concrete tasks, comparing them with current approaches. While this will undoubtedly result in new requirements, we believe that empowering users by providing them with sophisticated semantic annotation support for existing content will be a significant improvement for accessing and processing content in the Web.

Acknowledgements. Ting Tang contributed to the OWL NLP ontology and its population GATE component.

References

1. Mani, I.: Automatic Summarization. John Benjamins (2001)
2. NIST: DUC 2001. In: Proceedings of the ACM SIGIR Workshop on Text Summarization DUC 2001, New Orleans, Louisiana USA, NIST (2001)
3. Dang, H.: Overview of the tac 2008 opinion question answering and summarization tasks. In: Proceedings of the First Text Analysis Conference (TAC 2008), Gaithersburg, Maryland, USA, NIST (November 17-19 2008)
4. Pang, B., Lee, L.: Opinion mining and sentiment analysis. Foundation and Trends in Information Retrieval **2**(1-2) (2008) 1–135
5. Cunningham, H., Maynard, D., Bontcheva, K., Tablan, V.: GATE: A Framework and Graphical Development Environment for Robust NLP Tools and Applications. In: Proc. of the 40th Anniversary Meeting of the ACL. (2002) http://gate.ac.uk.
6. Ferrucci, D., Lally, A.: UIMA: An Architectural Approach to Unstructured Information Processing in the Corporate Research Environment. Natural Language Engineering **10**(3-4) (2004) 327–348
7. Witte, R., Gitzinger, T.: Semantic Assistants – User-Centric Natural Language Processing Services for Desktop Clients. In Domingue, J., Anutariya, C., eds.: ASWC. Volume 5367 of Lecture Notes in Computer Science., Springer (2008)
8. Witte, R., Bergler, S.: Next-Generation Summarization: Contrastive, Focused, and Update Summaries. In: International Conference on Recent Advances in Natural Language Processing (RANLP 2007), Borovets, Bulgaria (September 27–29 2007)
9. Witte, R., Khamis, N., Rilling, J.: Flexible Ontology Population from Text: The OwlExporter. In: Int. Conf. on Language Resources and Evaluation (LREC). (2010)
10. Witte, R., Gitzinger, T., Kappler, T., Krestel, R.: A Semantic Wiki Approach to Cultural Heritage Data Management. In: Language Technology for Cultural Heritage Data (LaTeCH 2008), Marrakech, Morocco (June 1st 2008)
11. Farrar, S., Lewis, W.D., Langendoen, D.T.: A common ontology for linguistic concepts. In: In Proceedings of the Knowledge Technologies Conference. (2002)
12. Hyvönen, E., Saarela, S., Viljanen, K.: Application of ontology techniques to view-based semantic search and browsing. The Semantic Web: Research and Applications (2004) 92–106
13. Amardeilh, F., Laublet, P., Minel, J.L.: Document annotation and ontology population from linguistic extractions. In: K-CAP '05: Proceedings of the 3rd international conference on Knowledge capture, New York, NY, USA, ACM (2005) 161–168
14. Java, A., Nirenburg, S., McShane, M., Finin, T., English, J., Joshi, A.: Using a Natural Language Understanding System to Generate Semantic Web Content. International Journal on Semantic Web and Information Systems **3**(4) (2007)
15. Buitelaar, P., Cimiano, P., Haase, P., Sintek, M.: Towards linguistically grounded ontologies. In: 6th Annual European Semantic Web Conference. (2009) 111–125