Semantic Description of Text Mining Services

Katja Pfeifer
SAP Research Dresden
SAP AG
01187 Dresden, Germany
katja.pfeifer01@sap.com

Alexander Schill
Computer Networks Group
Technische Universität Dresden
01062 Dresden, Germany
alexander.schill@tu-dresden.de

***Abstract*—Today, a huge amount of crucial business knowledge is hidden in unstructured text sources, such as word documents, web pages or forum entries. In order to exploit this knowledge text mining techniques were developed that are able to automatically extract or annotate entities, their relations or sentiments from textual sources. Recently, a number of text mining services that offer REST or SOAP APIs for easy consumption were published. These services differ strongly in their mining abilities and result quality and are often constructed for specific use cases. In practice, it is often desirable to combine results of multiple services to increase quality and functionality. However, this result combination is difficult since descriptions of service functionalities are often rarely documented and not standardized so that searching for specific text mining characteristics is time consuming and complex. In this paper we introduce a categorization of text mining services and provide a novel description ontology for describing functional characteristics of a text mining service. The ontology, being of interest for practitioners as well as researchers, is completed by application examples and descriptions that are made publicly available. Through the ontology and the descriptions presented in this paper the automatic use and combination of different text mining services is enabled.**

Keywords*-*Text Mining; Semantic Description; Service Oriented Architecture.

I. Introduction

Today, more than 80 percent of business-relevant information only exists in unstructured, mostly textual form such as web pages, office documents or forum entries as estimated in [7]. Exploiting this knowledge in business intelligence applications will be crucial for business competitiveness in future. In order to satisfy the need for knowledge extraction from text, a large quantity of text mining approaches have been developed (see [10] for an overview). These support a wide range of knowledge harvesting tasks like the classification of text documents, the recognition of entities and relationships or the identification of sentiments. Recently, more and more of these text mining solutions were made publicly available as Web - or Rest Services (e.g., OpenCalais [23] and AlchemyAPI [17]) to simplify consumption and integration.

Even though there are many text mining solutions available, some major problems remain unsolved. Text mining often still faces the problem of inaccuracy and incompleteness, and therefore, limits the confidence in information extracted by text mining solutions. Moreover, most of the solutions are developed for specific use cases and are not usable for others.

In order to alleviate these problems, we strive for a combination of multiple text mining services as described in [20]. The idea is to raise the quality of text mining by combining the strength and weaknesses of different approaches. Unfortunately, searching for specific text mining functionalities and combining these is cumbersome and often leads to a great amount of manual work.

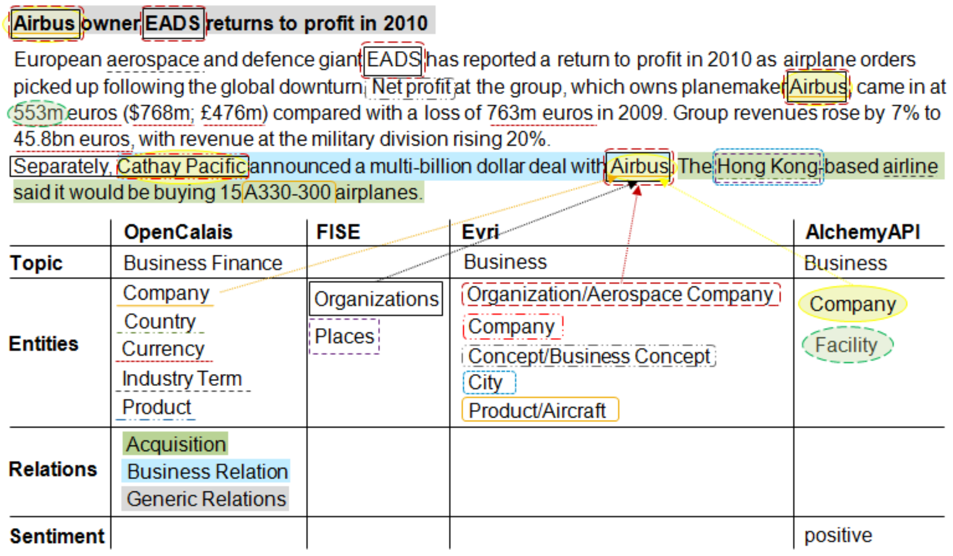
In this paper, we address the need for a comprehensive semantic description of text mining services to simplify the search for specific mining functionalities and therefore allow to automate the combination of text mining services. In particular we make the following contributions that are of interest for practitioners as well as researchers:

- We classify existing text mining services and highlight their similarities and differences.
- We propose a novel description ontology that can be used to comprehensively describe text mining services.
- We present descriptions for common text mining services and made them publicly available.
- We show that the descriptions can be used to select text mining services based on their functionalities.

The remaining paper is structured as follows: In Section II, we further motivate the need for combining text mining results and introduce a system architecture that supports automatic combination of existing services. Related work is reviewed in Section III. To infer the information necessary for the descriptions, we classify existing text mining services in Section IV. The novel description ontology is introduced in detail in Section V and complemented by application examples in Section VI. Finally, Section VII concludes our paper.

II. Motivating Text Mining Service Combination

In order to further motivate the need for a combination of text mining services, an example is given in Figure 1. The figure shows an extract of a BBC news article together with text mining results that were extracted by four different services - in particular, OpenCalais, AlchemyAPI, FISE [12]



Airbus owner EADS returns to profit in 2010

European aerospace and defence giant EADS has reported a return to profit in 2010 as airplane orders picked up following the global downturn. Net profit at the group, which owns planemaker Airbus, came in at 553m euros ($768m; £476m) compared with a loss of 763m euros in 2009. Group revenues rose by 7% to 45.8bn euros, with revenue at the military division rising 20%.

Separately, Cathay Pacific announced a multi-billion dollar deal with Airbus. The Hong Kong-based airline said it would be buying 15 A330-300 airplanes.

	OpenCalais	FISE	Evri	AlchemyAPI
Topic	Business Finance		Business	Business
Entities	Company Country Currency Industry Term Product	Organizations Places	Organization/Aerospace Company Company Concept/Business Concept City Product/Aircraft	Company Facility
Relations	Acquisition Business Relation Generic Relations			
Sentiment				positive

Figure 1. Analysis of a business news by several text mining services

and Evri [5] (the news article and the text mining results were retrieved on March 9, 2011). All of them are able to extract some entities such as companies, cities or products, but differ in the completeness and correctness of the results and the used annotation taxonomies. In addition, some services extracted more enhanced information such as relationships or overall topics and sentiments.

It is desirable to combine the results of these different services as proposed in [20]. Figure 2 illustrates a possible architecture of a system that is able to combine several text mining services. The lower part depicts a number of exemplary services (S1-S3) offering text mining functionalities with inconsistent interfaces and different entity taxonomies (T1-T3). We introduce a layer of wrappers that harmonize the individual service interfaces on the syntactical level and are considered in-depth in future work. These wrappers should be manually or possibly semi-automatically provided by the community or the service provider. They are simple services rewriting and adapting the original service answers to a unified format in order to facilitate the reuse and combination of the service results.

Additionally, we propose that each service functionality is semantically described using a standardized text mining ontology. This is needed since most services are often only rarely documented or documented in non machine-readable form on a website. Furthermore, available descriptions only specify the services on a syntactical level regarding their interfaces, their data types and bindings and their access modalities. The semantic description can then be put into a registry that helps to automatically find the adequate services for an envisioned text mining task. A text mining combination system is able to call multiple text mining services and combine their results.

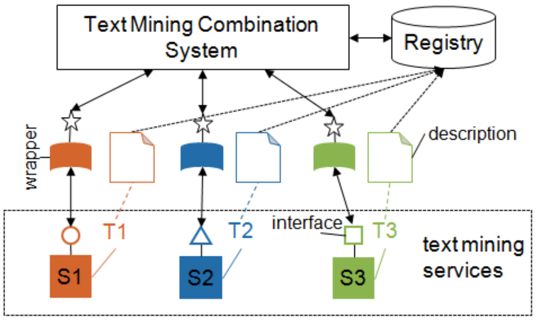


Figure 2. Architecture of service-oriented Text Mining

By extending the work of [20], the current paper focuses on the semantic description of text mining services. Before we start to introduce our work, we will give an overview on related work and distinguish it from our contributions.

III. Related Work

The first service-oriented text mining approaches, especially in the domain of information extraction, have been discussed by Habegger et al. [9] and Grover et al. [8]. Both approaches break down information extraction processes into single partial operations and offer a language or accordingly an ontology for the basic description of these service artifacts. In contrast to our ontology, none of them offers a semantic description capable to specify text mining functionalities for complex services. A first service-oriented information extraction system that uses text mining services is presented by Starlinger et al. [21]. It connects biomedical text mining services having standardized interfaces and a common taxonomy and combines their results to improve extraction quality. An automatic identification of services

Service	**Domain**	**Taxonomy**	**Text Mining Functionalities**					
			NER	**RE**	**TC**	**CT**	**KE**	**SA**
AIIAGMT[11]	biomedical		(en:) genes	-	-	-	-	-
AlchemyAPI	generic	list	en,fr,es,de,it,pt,ru,sv; LD, ED, QE	-	en,fr,es,de,it, pt,ru,sv	en:LD	en,fr,es,de,it, pt,ru,sv	en,fr,es,de,it,pt,ru,sv: polarity; d-, e-, k-level
BeliefNetworks [1]	generic		-	-	-	en	-	-
Evri	generic	service call (types & facets)	en; links to Evri knowledge base	-	en	-	-	-
Extractiv [6]	generic	list	en; LD, ED	en; (QE)	en	-	-	-
FISE	generic	list (DBpedia types)	en; basic LD	*planned*	-	-	-	-
OpenAmplify [15]	generic		(en: proper nouns)	(en: actions)	en	-	en	en: polarity; d-level
OpenCalais	business, finance, generic	owl (types & attr.) for en/ list (types & attr.) for es,fr	en; LD, ED es,fr	en	en: list	en: social tags	-	-
PIE [22]	biomedical		(en:) protein	en: protein interactions	-	-	-	-
uClassify [24]	generic	list (topic hierarchy)	-	-	en	-	-	en: polarity, mood; d-level

Table I
OVERVIEW OF EXISTING TEXT MINING SERVICES

based on their functionalities and corresponding descriptions is completely missing in this approach.

The CLARIN project [4] has the vision to create a research infrastructure of language resources and therefore also touches the problem of descriptive meta data for language services. In [14], a minimal set of meta data for language tools is detected. In contrast to our work, the CLARIN project mainly focuses on basic language tools (e.g., tokenizer, POS-tagger) and the (semi-)automatic build of chains between those tools. CLARIN does not review complex end user services as we do and additionally does not provide an ontology for describing the functionality of end user services like OpenCalais or AlchemyAPI.

Different web service description languages exist to describe services regarding their functionalities, the used data types, the protocols and the provided interfaces. The W3C standard Web Services Description Language (WSDL) [3] was established for the syntactic description of services. More recent approaches [13], such as the Ontology Web Language for Services (OWL-S), Web Service Modeling Ontology (WSMO) and Semantic Annotations for WSDL and XML Schema (SAWSDL) added additional semantic descriptions in order to allow the automatic selection of services based on their functionalities. Nevertheless, all semantic descriptions need a well-defined ontology for describing the service features (even the functionality description of the OWL-S profile needs complementary ontology elements to specify the input, output, precondition and effect properties). As stated above, there has been no such ontology for describing text mining services comprehensively. To the best of our knowledge, we are the first to approach this problem providing an ontology for describing text mining services.

IV. Classification of Text Mining Services

We intensively studied existing text mining services with regard to the functionalities they offer and their special characteristics and limits. We provided a first overview of existing services in [20], which we extended in Table I, focusing on the text mining specific characteristics. We selected text mining services from different domains with different functionalities (named entity recognition, relationship extraction, categorization, concept assignment, keyword extraction, sentiment analysis) and tried to cover the most established one. First of all we studied the domain the services have been designed for. We mainly discovered generic services (i.e., not being specialized for any specific domain) and services from the biomedical and business domain. We further analyzed the different text mining functionalities and identified six main types:

- *Named Entity Recognition (NER)* where entities are identified and classified into predefined categories (e.g., person, organization),
- *Relation and Interaction Extraction (RE)* for the identification of relationships between two or more entities,
- *Text Classification/Categorization (TC)* where categories are assigned to text documents,
- *Concept Tagging (CT)* for the assignment of specific terms that are derived from the text content (the terms do not have to be included in the text),
- *Keyword Extraction (KE)* where the essence of the text is extracted through the identification of the main keywords of a text and
- *Sentiment Analysis (SA)* for the extraction of any subjective information from text (e.g., polarity, attitudes, mood).

We studied these six text mining types more extensively and identified their essential properties. All types are language-dependent and most functionalities are currently only provided for English text input. The information extraction tasks NER, RE and in parts also TC are identifying elements of predefined categories. Therefore, the service providers generally release a taxonomy defining the entities, relations and classification categories (by an ontology file, a list on the service website or indirectly via some service calls). The taxonomies differ in their semantic and syntactic

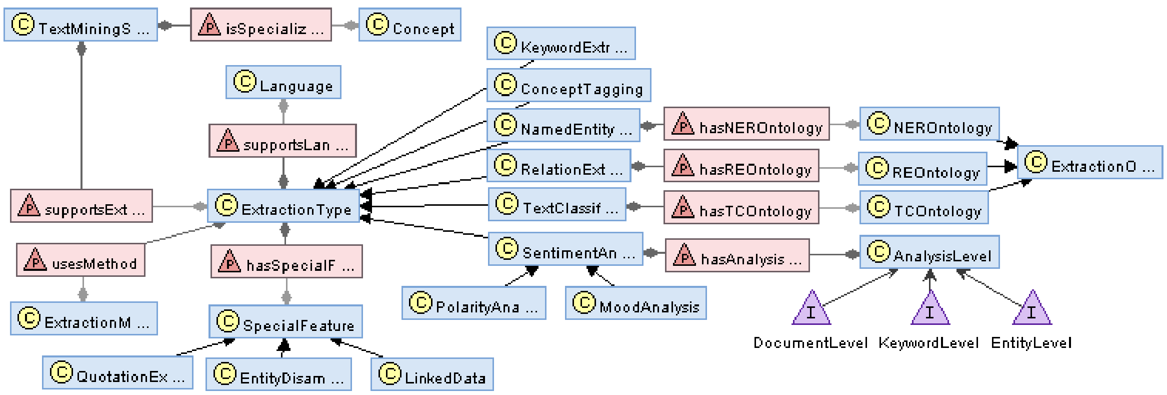


Figure 3. Ontology for the description of text mining services

granularity - some are enhanced with attributes and facets, others are only providing flat basic types. Under SA, we recapped all text mining functionalities touching subjective information. Although some of the features could also be classified under other text mining types like NER (as the subjective information is extracted on entity level), we hold that it is a text mining type on its own. For better specification, we differentiated between three analysis levels - document (d), entity (e) and keyword (k) - and indicated the exact sentiment type a service provides.

Further extraction features are offered by several services for some text mining types. The most common feature is the Linked Data (LD) support, where the extracted objects are linked to existing LD sources with additional information (e.g., a link from the person entity *Barack Obama* to a LD URL characterizing him). Another feature is the disambiguation of entities (ED) where detected instances are resolved to a unique instance (e.g., *IBM* and *IBM Corp.* are resolved to one entity). One service additionally provides a quotation extraction (QE), where person entities are complemented by quotations from this person found in the input text.

Based on the analysis of existing text mining services, we conclude that a description language for text mining services should satisfy the following requirements: describe the domain the service is designed for (R1), indicate the text mining type(s) provided by the service (R2) plus the respective languages (R3), point to the ontologies used for the predefined category types (R4) and describe the special features being available for the text mining types (R5). Additional information on the used extraction methodology, service charges and limitations can complete the description. In the following we will present an expandable ontology for the description of text mining services satisfying the mentioned requirements.

V. The Text Mining Description Ontology

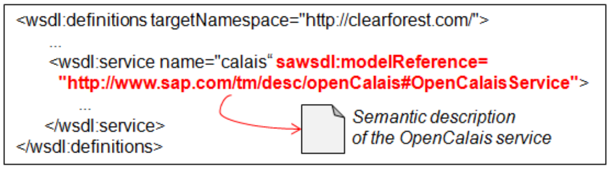
Figure 3 presents our high level ontology for the description of text mining services. We especially focused on the expandability and simplicity of the ontology and modeled it with the Resource Description Framework (RDF) Schema [2]. We chose RDF since it is easy to use and provides sufficient mechanisms to define classes, properties and their relationships. RDF allows for easy extensibility and re-use of existing well-defined and more specific ontology parts and also supports user service specific extensions. In addition, it is not a problem to interlink the syntactic service description to a description provided in RDF. Before we explain this interlinking of classical descriptions with our extension, we will shortly introduce the classes and properties of our text mining description ontology.

The class *TextMiningService* is the entry point for a semantic description of a text mining service and can be used for the interlinking of the classical service description and the text mining specific description. An instance of this class represents a service with a well defined interface that offers some text mining functionalities. Via the property *supportsExtractionType*, specific text mining functionalities indicated by instances of the class *ExtractionType* are linked to the service (R2). Several subclasses of *ExtractionType* are available for the exact specification of the text mining tasks being provided. If a text mining service is specialized for a certain domain (R1), this will be indicated with the property *isSpecializedFor* that connects a *TextMiningService* with a *Concept* from the Simple Knowledge Organization System (SKOS) [25] ontology. The supported languages of a service or a concrete text mining type (R3) are given with the property *supportsLanguage* pointing to a *Language* from DBPedia. Characteristics of the *ExtractionType* can be specified with additional properties (e.g., *hasSpecialFeature*, *hasNEROntology*, *hasREOntology*, ...). In order to indicate special features provided by the services (R5), the class *SpecialFeature* and some subclasses for concrete features are provided by our ontology. The extraction ontology used by a service (R4) can be indicated with instances from the class *ExtractionOntology*. These instances are mainly used to link to the existing taxonomies of the services. The methods used for the extraction can optionally be specified with instances from the class *ExtractionMethod*. We decided to model the subtypes of *SpecialFeature* and *ExtractionOntology*, as well as *ExtractionMethod* as classes, as we believe that they should be further specified with extra properties in future.

VI. Application of the Ontology

After having explained our proposed ontology in the previous section, we will now show how our ontology can be applied in practice. Semantic service descriptions using our presented ontology can for example be linked to the WSDL [3] description of a service through Semantic Annotations for WSDL and XML Schema (SAWSDL) [13]. In the following, we show an exemplary semantic description of the text mining service OpenCalais and the integration into the syntactic WSDL description.

Figure 4 shows an extract from the annotated WSDL file of OpenCalais (The original WSDL file can be found at [16]). The pointer to the semantic description can be integrated as SAWSDL *modelReference* into any service model element in the WSDL description. However, as our semantic description characterizes the service, we prefer a linkage from the WSDL service element. Other linking concepts between the syntactic and the semantic description are of course possible as our semantic descriptions build upon open standards. We extended the OpenCalais WSDL *service* element in Figure 4 with a SAWSDL *modelReference* pointing to a semantic description of the OpenCalais functionalities.



```
<wsdl:definitions targetNamespace="http://clearforest.com/">
    ...
    <wsdl:service name="calais" sawsdl:modelReference=
     "http://www.sap.com/tm/desc/openCalais#OpenCalaisService">
       ...
    </wsdl:service>
</wsdl:definitions>
```

Figure 4. Extract from WSDL of OpenCalais service annotated with SAWSDL

Listing 1 displays an extract of the semantic information that can be found under the linked URI and all its connected resources. This exemplary semantic description of the OpenCalais service makes use of our previously introduced ontology (notice that this is not a complete description of all the functionalities of the OpenCalais service.).

```
@prefix oc: <http://www.sap.com/tm/descr/openCalais#> .
@prefix tm: <http://www.sap.com/tm/descr/ontology#> .
@prefix dbpedia: <http://dbpedia.org/resource/> .

oc:OpenCalaisService a tm:TextMiningService;
  tm:isSpecializedFor dbpedia:Category:Business;
  tm:supportsExtractionType oc:NEREnglish,
                                          ...
                                          oc:DocumentCategorization .
oc:NEREnglish a tm:NamedEntityRecognition;
  tm:supportsLanguage dbpedia:English_language;
  tm:hasSpecialFeature  oc:EntityDisambiguation,
                        oc:LinkedData;
  tm:hasNEROntology oc:OntologyEnglish .
oc:EntityDisambiguation a tm:EntityDisambiguation .
oc:LinkedData a tm:LinkedData.
oc:OntologyEnglish a tm:NEROntology;
  nie:url http://www.opencalais.com/files/owl.opencalais
      -4.3a.xml .
...
oc:DocumentCategorization a tm:TextClassification;
  tm:supportsLanguage dbpedia:English_language .
...
```

Listing 1. Extract of a semantic description for the OpenCalais service in N3 notation

We started describing a number of text mining services with our ontology and will continuously add and extend descriptions. The ontology as well as the descriptions files can be found under [18]. The fake URIs have to be replaced for usage. The text mining service combination system makes use of the descriptions. Therefore, the describing triples are stored in a triple store like Sesame. Adequate services are then searched as follows.

A. Searching specific Services

As our descriptions of text mining services use RDF triples, it is obvious to query the descriptions with the help of SPARQL [19] a query language based on graph patterns. We will now demonstrate how to identify and select text mining services with specific functionalities that are described with our ontology. The given queries are just examples. Actual queries may be much more complex. The first query (Listing 2) selects text mining services for NER on Spanish text documents where the extracted entities are connected to Linked Data resources if possible. Figure 5 shows the corresponding query pattern for this SPARQL query.

```
PREFIX tm:<http://www.sap.com/tm/descr/ontology#>
PREFIX dbpedia:<http://dbpedia.org/page/>

SELECT DISTINCT ?service
WHERE {
  ?service tm:supportsExtractionType ?type .
  ?type a tm:NamedEntityRecognition ;
    tm:supportsLanguage dbpedia:Spanish_language ;
    tm:hasSpecialFeature ?feature .
  ?feature a tm:LinkedData .
}
```

Listing 2. SPARQL query to select services providing NER with Linked Data for Spanish text

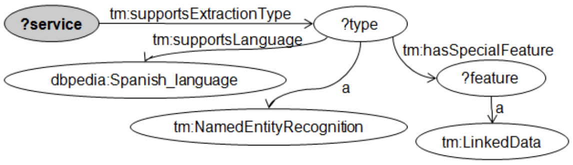


Figure 5. Query pattern for SPARQL query in Listing 2

Listing 3 shows another SPARQL query that selects services capable to analyze the mood of an English text document.

Copyright (c) IARIA, 2012. ISBN: 978-1-61208-227-1

```
PREFIX tm:<http://www.sap.com/tm/descr/ontology#>
PREFIX dbpedia:<http://dbpedia.org/page/>

SELECT DISTINCT ?service
WHERE {
  ?service tm:supportsExtractionType ?type .
  ?type a tm:MoodAnalysis ;
    tm:supportsLanguage dbpedia:English_language ;
    tm:hasAnalysisLevel tm:DocumentLevel .
}
```

Listing 3. SPARQL query to select services providing sentiment analysis (mood) for English text on document level

VII. Conclusions and Future Work

In this paper, we made a number of contributions that help to simplify the description, search and reuse of text mining services and their result combination. First of all, we gave an overview on existing text mining services and classified them according to their functionalities. We further derived a novel description ontology for text mining services capable to describe complex end-user services. In contrast to previous work, we explicitly covered the real mining functionalities into the descriptions. Auxiliary, we built on open standards to easily connect descriptions using our ontology to already existing descriptions and standardizations.

As starting point for further work on the selection, reuse and combination of text mining services, we described a number of such services and made them publicly available. With this basis, the combination of text mining services as proposed in [20] is enabled. Future work will have to focus on the well-directed extension of the ontology as well as the derivation of rules and heuristics for the combination of the service results and the evaluation of the system. Another research area we investigate is the matching of service taxonomies to retrieve mappings between them and possibly even a global taxonomy. These mappings can then complement the service descriptions presented in this paper.

References

[1] BeliefNetworks. http://beliefnetworks.net/bnws/, retrieved: April, 2012.

[2] D. Brickley and R. V. Guha. Rdf vocabulary description language 1.0: Rdf schema. *W3C Recommendation*, 10, 2004.

[3] R. Chinnici, J.-J. Moreau, A. Ryman, and S. Weerawarana. Web Services Description Language (WSDL) Version 2.0 Part 1: Core Language. Technical report, World Wide Web Consortium (W3C), 2007.

[4] CLARIN. http://www.clarin.eu/, retrieved: August, 2012.

[5] Evri. http://www.evri.com/developer/, retrieved: July, 2012.

[6] Extractiv. http://extractiv.com/, retrieved: August, 2012.

[7] S. Grimes. Unstructured Data and the 80 Percent Rule. Clarabridge Bridgepoints, 3rd quarter 2008.

[8] C. Grover, H. Halpin, E. Klein, J. L. Leidner, S. Potter, S. Riedel, S. Scrutchin, and R. Tobin. A Framework for Text Mining Services. In *AHM'04 Proc.* EPSRC, 2004.

[9] B. Habegger and M. Quafafou. Web Services for Information Extraction from the Web. In *ICWS'04 Proc.*, page 279. IEEE Computer Society, 2004.

[10] A. Hotho, A. Nürnberger, and G. Paaß. *A Brief Survey of Text Mining*, volume 20, pages 19–62. Gesellschaft für linguistische Datenverarbeitung, 2005.

[11] C.-N. Hsu, Y.-M. Chang, C.-J. Kuo, Y.-S. Lin, H.-S. Huang, and I.-F. Chung. Integrating High Dimensional Bi-directional Parsing Models for Gene Mention Tagging. *Bioinformatics*, 24:286–294, 2008.

[12] Interactive Knowledge Stack. Furtwangen IKS Semantic Engine. http://wiki.iks-project.eu/index.php/FISE, retrieved: August, 2012.

[13] M. Klusch. *CASCOM - Intelligent Service Coordination in the Semantic Web*, chapter Semantic Web Service Description, pages 41–68. Birkhuser Basel, 2008.

[14] L. Lemnitze, E. Hinrichs, and A. Witt. Language Resources, Taxonomies and Metadata. In G. Heyer, editor, *Text Mining and Services Proc.*, volume XIV of *Leipziger Beiträge zur Informatik*, pages 25–39, Leipzig, 2009.

[15] OpenAmplify. http://www.openamplify.com/, retrieved: August, 2012.

[16] OpenCalais WSDL. http://api.opencalais.com/enlighten/?wsdl, retrieved: August, 2012.

[17] Orchestr8. AlchemyAPI. http://www.alchemyapi.com/, retrieved: August, 2012.

[18] K. Pfeifer. Text Mining Ontology and Descriptions. http://areca.co/20/Text-Mining-Service-Descriptions, retrieved: August, 2012.

[19] E. Prud'hommeaux and A. Seaborne. SPARQL Query Language for RDF. Technical report, W3C, 2006.

[20] K. Seidler and A. Schill. Service-Oriented Information Extraction. In *Joint EDBT/ICDT Ph.D. Workshop'11 Proc.*, pages 25–31, New York, NY, USA, 2011. ACM.

[21] J. Starlinger, F. Leitner, A. Valencia, and U. Leser. SOA-Based Integration of Text Mining Services. In *SERVICES '09 Proc.*, pages 99–106, Washington, DC, USA, 2009. IEEE Computer Society.

[22] K. Sun, S.-Y. Shin, I.-H. Lee, S.-J. Kim, R. Sriram, and B.-T. Zhang. PIE: An Online Prediction System for Protein-Protein Interactions From Text. *Nucleic Acids Research*, 36:411–415, 2008.

[23] Thomson Reuters. The OpenCalais Web Service. http://www.opencalais.com/, retrieved: August, 2012.

[24] uClassify. http://uclassify.com/, retrieved: August, 2012.

[25] World Wide Web Consortium. *SKOS Simple Knowledge Organization System Reference*, August 2009.