Knowledge Obtention Combining Information Extraction Techniques with Linked Data

Ángel Luis Garrido
Heraldo de Aragón
Zaragoza, Spain
algarrido@heraldo.es

Pilar Blázquez
IIS Department, University of Zaragoza
Zaragoza, Spain
blazquez@unizar.es

María G. Buey
IIS Department, University of Zaragoza
Zaragoza, Spain
mgbuey@unizar.es

Sergio Ilarri
IIS Department, University of Zaragoza
Zaragoza, Spain
silarri@unizar.es

ABSTRACT

Today, we can find a vast amount of textual information stored in proprietary data stores. The experience of searching information in these systems could be improved in a remarkable manner if we combine these private data stores with the information supplied by the Internet, merging both data sources to get new knowledge. In this paper, we propose an architecture with the goal of automatically obtaining knowledge about entities (e.g., persons, places, organizations, etc.) from a set of natural text documents, building *smart data* from raw data. We have tested the system in the context of the news archive of a real Media Group.

Categories and Subject Descriptors

H.4 [**Information Systems Applications**]: Miscellaneous; I.2.4 [**Knowledge Representation Formalisms and Methods**]: Representations (procedural and rule-based)

General Terms

Theory

Keywords

knowledge obtention; text mining; linked data

1. INTRODUCTION

When searching the Web with search engines, the typical procedure is to introduce a number of keywords to search, and then receive a huge number of links which must be manually filtered and examined in order to identify the desired information. This problem is starting to happen in company's private data stores with large amounts of data. Certainly, the efficiency of searching for information by hand is very limited. The problem is compounded when the data are embedded in natural language texts. Moreover, an additional problem is that these data are usually isolated, so if the user wants to enrich the results with information from the Internet, then he/she will be forced to make a second search with the same problem defined above and the additional problem of combining data properly.

To solve this kind of problem we propose a new approach, based mainly on obtaining knowledge from the raw data combined with information from reputable Internet sources (for example, verified linked data repositories), in order to be able to answer frequently asked questions in a more precise way by giving the user less information but more relevant. This information will be supplied in a structured form (for instance, a report) more readable by users. An illustration of the general architecture of this proposal can be seen in Figure 1.

As an example, consider a database of textual medical reports where the information sought is generally related to doctors, patients and diseases. In this scenario the idea would be to create a knowledge base that includes the attributes of each of these entities, as well as data that would complete these attributes. Thus, when a user looks for information related to a particular disease, the system in addition to the typical list of links to records that contain the searched term, could directly receive a report describing the symptoms of the disease, treatments, links to the records of patients who have suffered this illness, access to the records of doctors who has written these reports, the various drugs used in treatment and its effectiveness, a distribution by months of cases, or even a geographic histogram of the origins of patients. This information can also be enriched with recognized medical information websites. Furthermore, if exists, we could query a SPARQL endpoint in order to recover more information to enrich our report. This kind of report could save hours of work of gathering information, or even could be useful to reply a very concrete question made by a user by using a query-answering system.

WWW 2015 Companion, May 18–22, 2015, Florence, Italy.
ACM 978-1-4503-3473-0/15/05.
http://dx.doi.org/10.1145/2740908.2741708.

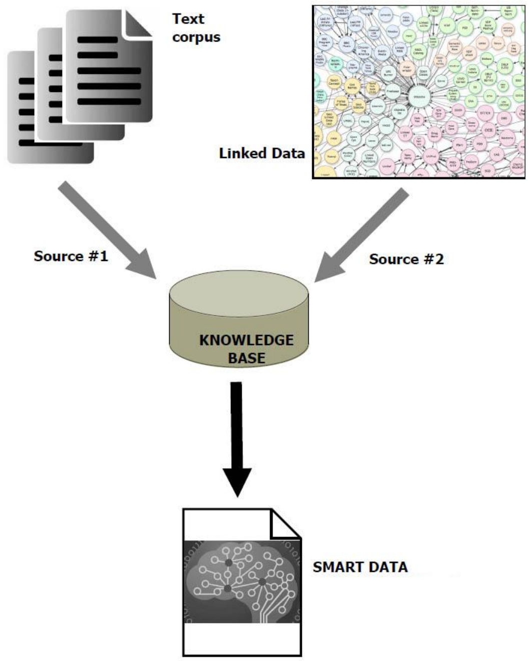

Figure 1: General architecture description.

In this paper, we describe the general architecture of the proposed solution and we show an experience work performed in a real media company: Heraldo de Aragón[1]. We have developed a software, integrated with the news archive of the company that is be able to automatically generate informative data sheets about featured entities, namely people, places, and organizations, enriched with information obtained from the Web. This process thus generates *smart data* from a set of heterogeneous and raw data sources and it is very useful because it helps journalists to find concrete information about these issues without having to search in a huge set of links from the news archive and from the web.

The main contribution of this work is the development of a system able to create a knowledge base which facilitates recovering the most important information about featured entities from a corpus of semi-structured documents, and with the ability of adding information obtained from the linked data in the Web. We have contributed to the state of the art by analysing an experience of the application of techniques related to information extraction, text mining, disambiguation, natural language processing (NLP), and semantics, with the aim of getting the information in order to build a knowledge base from a text corpus.

The rest of this paper is structured as follows. Section 2 explains the general architecture of the proposed system. Section 3 refers to the development and evaluation of a prototype of the system with a real corpus of data. Section 4 analyses other related works. Finally, Section 5 provides our conclusions and future work.

2. METHODOLOGY

In this section we provide an overview of the proposed architecture. The main objective is to automatically generate a knowledge base with the relevant information about the entities that appear in a text corpus by using, on the one hand, this corpus, and on the other hand, data obtained from the Web about this entity. The format of the data may be either text from linked data sources, or multimedia content such as images, videos, web pages or sounds, expressed through its paths or its hyperlinks. The attributes from the objects of the knowledge base could be not predefined, but be created dynamically from the information extracted from the private corpus and from the Internet.

2.1 Pre-processing

Performance considerations have been very important when adopting design decisions, that have led to the design of preprocessing techniques over the information stored in the text corpus:

- We have decided to create a catalogue of predefined *named entities* [21] in order to improve the efficiency and with the aim of disambiguating between different entities. In case of availability of a predefined list of terms for the corpus, we could link it to this catalog, and even we could link them with other entries obtained from external sources (free-access ontologies, linked data, etc.). In our experiments we have used mainly DBpedia[2] as the external data source, since it is one of the most complete and well known repositories of linked data, but any other data source of this nature could help us to complete the list of terms. The whole process of creating a named entity catalogue is explained in Section 2.4.

- We have chosen to perform an additional statistical task before the creation of the knowledge base: obtaining information concerning the frequencies of entities. This process is based on the well-known algorithm TF-IDF [19] combined with Freeling[3], a specific NLP software. This information about the frequencies will be useful to disambiguate entities, as well as to score the most relevant texts concerning an entity.

- Finally, we decided to tag off-line all the texts stored in the private repository by labelling them with their most relevant entities, assigning several scores to each text for each entity. Thus, we took advantage from having at our disposal a closed dataset and then we will be able to directly access the most relevant texts related with each entity in a faster way by using a sorted inverted index. For this purpose, we have included in the system a process to rank text documents regarding the catalogue of named entities. This task is explained in Section 2.5.

2.2 Creating the Knowledge Base

It has been also established as a prerequisite to decide the number of entities to be included in the knowledge base (NEK) and the degree of depth in the corpus (DDC) required to gather information from each of them. So, the system will take the NEK most relevant entities, i.e., those with greater presence in the corpus. Then, the system will recover the DDC texts with higher scores containing that

[1] http://www.heraldo.es

[2] http://dbpedia.org/

[3] http://nlp.lsi.upc.edu/freeling/

entity. If we increase NEK and DDC the time spent building the knowledge base will be higher, but the information will be more accurate. In order to collect these texts, we take advantage of the pre-processing tasks described above.

After these previous works, it begins the process of creating the basis of knowledge: For each entity in the catalogue the system will retrieve the most relevant DDC texts and will perform a filtering of sentences, saving only those in which the entity appears. Each sentence is passed through a parser that syntactically analyses sentences where that entity appears (E), extracting the used verb (V) and grammatical complements (C). The verb and the complements are lemmatized using a morphological analyser, in order to the process may return a set of triplets with semantic information related to the (E, V, C) entity that is stored in an RDF repository.

On the other hand, if that entity is linked to an object of type linked data (for example, an entity of DBpedia), by using SPARQL queries we can also retrieve information about that entity concerning their attributes and their relationships with other entities. Here we will have a different level of depth (DDLD) to determine how many "jumps" between relations the system must execute to recover other related entities and relations. Both attribute values, such as relationships, will be stored in the same repository as RDF triplets, as described above, unifying the description of the entity.

When this process is carried out for each of the NEK entities, the system performs a process of merging and simplifying the triple store, based mainly on finding common elements among all the triplets stored in the RDF repository. For doing this, the system uses the methods detailed in TM-Gen [6], which is a tool that extracts information from any number of texts and represents them in a topic map format. Once a set of RDF triplets has been generated for each text, the system performs a merging of all of them with the aim of reducing it. To do this, we use a method previously developed in [14]. The method for merging is called SIM (Subject Identity Measure) and it is responsible for describing the relation among two subjects or topics. For simplifying, the system conducts an analysis to search redundancies and, in case of finding them, removes and reduces them into a single concept, using for this purpose a lexical database that contains semantic information of the words of a language. For example, if the system find the entities "car" and "automobile", it will search and select their best meaning and reduce them into only one topic, gathering and connecting their associations with it. Finally, we will obtain a final knowledge base with the mixed and simplified entities, joining the knowledge stored in the corpus text with the data obtained through linked data. The whole process can be seen in Figure 2.

2.3 Obtaining information

When a user queries the system about an entity, the system follows a sequence of steps to generate an informative report. First, the system performs a preliminary search in order to determine the presence of an entity in the catalogue of named entities (NEs), in the documental database, and in DBpedia. This task is performed in order to discard searches about entities that do not appear in any of those sources.

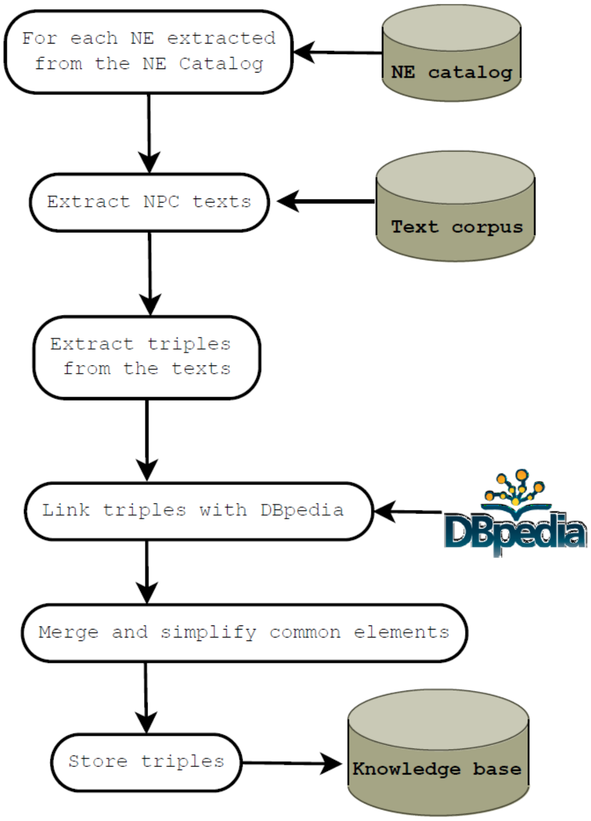
For each NE extracted from the NE Catalog
NE catalog
Extract NPC texts
Text corpus
Extract triples from the texts
Link triples with DBpedia
DBpedia
Merge and simplify common elements
Store triples
Knowledge base

Figure 2: General process of the knowledge base obtention.

In case of ambiguity problems, a disambiguation process needs to be carried out. It consists of searching the entity in the catalogue of named entities, which has been created previously, and searching it also in DBpedia. If the entity appears in the catalogue, it is prioritized. Then, the system recovers from the knowledge base the attributes and the relations with other entities and generates a report with this merged information. A predefined number of links to corpus elements could also appear in the report. Finally, it searches on the Internet and it extracts relevant links of webs, blogs, books, other news, videos, and social networks in order to complete the entity's report.

2.4 Creating a catalogue of named entities

This catalogue is used by the knowledge base generator to help in the searches for text and in the disambiguation of entities. It consists of several steps:

1. Extracting a list of entities from the linked data repository. For each entity the system extracts its complete name and it stores information about it.

2. If a thesaurus exists, the system recovers the list of entities from it.

3. Creating the named entity catalogue where both lists of entities are linked. In order to carry out this step, for each element identified as an entity in the thesaurus, the process searches if it appears in the list of entities extracted from linked data. If it appears, it creates

a new record with the information of that entity and links it with its corresponding thesaurus; otherwise, it only creates a new record referencing the thesaurus.

4. Completing the named entity catalogue with more information about the entities, which may not appear in the previous lists but in the text corpus. So, for each text, the system uses the morphological analyser to identify named entities. Then, it stores the information about the entities, and the system also saves the previous and subsequent words that appear next to a named entity, to be able to determine what type of entity is by using later a set of templates.

This catalogue of named entities will be used to improve their search, thanks to its pre-calculated links to the thesaurus and/or linked data. Besides, the catalogue allowed us to standardize the naming of entities.

2.5 Developing a ranking of relevant texts

In order to choose the most relevant texts, we have also designed an algorithm that provides the closest results desired, according to their relevance. Inspired by classical approaches [13], we have developed a weighting algorithm based on the frequency of occurrence of words and applying pre-filtering. The solution is based on the idea that the most repeated words are probably more relevant, except in the case of words belonging to certain lexical categories (determiners, pronouns, prepositions, auxiliary verbs, etc.).

First, the number of words in the news is taken account by selecting those that exceed a certain threshold. Occurrences of a named entity are scored taking into account other attributes (if they exist). Secondly, we have consider specific weights that assign a score for different ranges of occurrences of each name entity (very low frequency range, low frequency range, normal frequency range, etc.), such that texts with the higher scores can be selected. We have performed some preliminar tests of the system with different weights, and we have determined empirically parameters that provide acceptable results for those tests..

The relevance assessment process is launched frequently to keep the information in the database up-to-date. Besides, thanks to this process, the entity report generator can access the most relevant texts for each entity in a faster way.

3. CASE STUDY

The system can be applied for example in big companies, research centers, public administrations, big libraries, and media companies. Our case of study is Grupo Heraldo[4], a major media company in Spain that manages written publications and audiovisual businesses. This company owns several local newspapers and boasts an archive with more than 120 years of regional information. Therefore, the company owns a valuable amount of specific information about certain Spanish regions: news, photographs, interviews, reports, and multimedia data like animations, web pages, graphics, and videos. All these data are stored in a relational database with over 10 millions of registers including large amounts of text. The whole repository is closed to the general public, and it is only accessible to documentalists, journalists, and researchers. The stored documents are managed through a system called EMMA, which is a Content Management System (CMS) specific for media companies, distributed by Hiberus Technology[5].

3.1 Specific problems

The EMMA CMS system provides a set of tools to search, filter, and reach the desired information. In previous works [7, 8, 9], we have enhanced the system by applying techniques related to natural language processing (NLP), machine learning, and ontologies, in order to help the documentalists in their work of categorization and tagging of news. However, these kinds of systems still have two constraints:

1. They just basically retrieve potentially-relevant sources. After searching, users obtain a set of links pointing to documents. Then, they must analyse them one by one to identify which ones are relevant. Finally, users have to manually extract the desired information.

2. They are closed systems and often offer incomplete information. Therefore, when users search specific information about an entity (e.g., a town, an event, a company, or a person), it would be highly beneficial if they could complete the desired report by searching also in other external sources, such as the Web.

3.2 The prototype

We have developed a prototype that focuses on the obtention of a knowledge base from the archive news corpus merged with DBpedia information, with the aim of generating detailed data sheets of relevant entities (people, organizations, places,...etc.). Some screenshots of the prototype and a video demo can be found in the website of *The Genie Framework Project*, in the Downloads section[6]. The developed system consists of an application designed to be integrated into the EMMA CMS system, and it is focused on searching information, links, relevant news and photographs related with a certain character, by using a knowledge base.

Before starting, we have found some initial difficulties that require special attention:

- Extracting specific information from a large number of documents in natural language with a journalistic writing style is not an easy task. There are many studies that have faced this problem like [10, 15, 17] and it is not solved yet. We have had to tackle searches over 10,000,000 documents that in some cases host large texts, such as the pages of the newspaper.

- The system must unambiguously solve conflicts between two characters with the same name or between a character and a name of another kind of entity. For example, if we seek information about Michel York and the user does not provide the name, all the information related to York as a city must be manually filtered.

- Extracting information from a structured Linked-Data repository may seem easy, but we also found some problems. Just to mention the most significant: ambiguity, bad integrity of the data, or the lack of certain fields in some registers.

[4] http://www.grupoheraldo.es

[5] http://www.hiberus.com

[6] http://sid.cps.unizar.es/SEMANTICWEB/GENIE

The first step was to develop a set of offline services to perform the different tasks of data preparation (described in Section 2): NLP tasks, obtaining a catalogue of entities, and scoring the news. We have set to 100, 20 and 2 the values of NEK, DDC and DDLD respectively. We have obtained an automatic knowledge base containing near two hundred entities and populated with near 80.000 triplets in the RDF store. Afterwards, we developed a graphical user interface to support an easy interaction of end users with the system. A keyword-based search box is available for the user. The system handles the searches on the knowledge base, and it presents the results to the user.

For extracting data from the Internet, we have used Google APIs that let us obtain results from Google Search, Google Books, Google Blogger, Google Images, and Google News. To search videos, we have used the Youtube API. Finally, to extract information from DBpedia, we have developed another module with a set of functions which construct SPARQL queries which allow us to locate the desired information about a particular character. With all this information, the internal modules generate an XML file from the knowledge, which will be interpreted by the interface in order to finally present the data to the user in HTML format.

3.3 Evaluation

Assessing the quality of the knowledge base and the significance of the automatically-produced data sheets in the case of study is a very subjective task and entails reading all the news about a particular character. Therefore, it is complicated to perform a typical information retrieval evaluation based on the use of metrics such as recall and precision. For example, we have found 52,000 news stored in the repository of the newspaper Heraldo de Aragón talking about Ramón y Cajal. It is very difficult to precisely evaluate the correctness of the searching and data extraction algorithms over such large amounts of data; even if we could invest the human effort needed, the evaluation will be subjective, requiring the participation of a significantly-wide range of users. Instead, we benefited from the participation of users in order to fine-tune some internal parameters of the algorithms used (e.g., in the ranking of news the weight to use for different news depending on whether they appear on an even/odd page, or on the front cover, etc.).

4. RELATED WORK

Systems that process collections of documents in a certain domain to extract relevant information in a structured way are usually "tailor-made". They take a particular template to fill in with information collected from the texts. The main obstacle is the low portability, as templates are designed for a specific objective and they are hardly reusable.

If we aim at combining information extracted from different texts, a first problem is solving the issue of identifying and disambiguating named entities. We can find in [3] some methods to identify and classify named entities. Moreover, [20] is an interesting work about the recognition of named entities like people names and organizations. Mooney and Bunescu in [16] describe information extraction algorithms for identifying entities and relations in a text.

Infoboxes and Wikipedia are also used in [5] to solve the named entity recognition problem.

In relation to the problem of mixing structured data and texts, we find in [4] a method that identifies terms in the text document. Using this terms we can query the structured data in order to identify fragments that are relevant to the document. The method in [18] describes how to summarize a web entity (e.g., a person, place, product, etc., described in a web page) based on the entity's appearance in web documents.

About generating knowledge related with people, we can cite [2]. It describes a project which seeks to automatically extract knowledge about artists from the Web, populate a knowledge base, and use it to generate personalized narrative biographies. YAGO [23] is another well-known project related with extracting knowledge from the Web, in this case from Wikipedia and WordNet

Regarding tools related with information extraction in media companies, we can mention, for example, [11, 22, 24]. An automatic query-based technique has been developed in [1] to retrieve documents useful for the extraction of user-defined relations from large text databases such as media databases. A procedure to obtain significant paraphrases (different narrative styles to refer to the same event on the same day) from news articles by using named entity recognitionis described in [22].

The BBC is a referent in its use of DBpedia and linked data. In [12], the authors describe how the BBC is working to integrate data and linking documents across BBC domains by using Semantic Web technology, for instance, by using DBpedia or MusicBrainz.

5. CONCLUSIONS AND FUTURE WORK

Our main objective was to develop a system able to automatically generate a knownledge base from a private corpus of texts, integrated with data that can be extracted from the Internet. We have implemented the system in a real media company, and the knowledge base has been used to automatically generate informative report from a set of entities. The report consists of a number of fields with descriptive information, a selection of suitable news and relevant photographs, and a set of links to websites, videos, social networks, and links to other relevant information appearing on the Web. It significantly reduces the time needed to locate information, and it facilitates the daily work of documentalists and journalists.

The main contribution of this work are:

- Application of information extraction and natural language processing techniques to find ways to extract useful information with the purpose of generating a knowledge base from a set of semi-structured and unstructured texts, solving conflicts related to the ambiguity of similar names by applying disambiguation techniques.

- Integration in the same information retrieval system of techniques to obtain data sheets from both private repositories and other sources available in the Web. The system benefits from the Semantic Web, using a semantic repository like DBpedia to retrieve specific information.

A clear difference with other existing approaches is that the proposed system supports the creation of knowledge (*smart data*) from both private raw data and public data from the Internet. So, it clearly complements the capabilities of other tools such as DBpedia or infoboxes, because they only operate with public information.

As future work, we would like to convert part of the automatically generated *smart data* into linked data, in order to publish them and let them accessible. Disambiguation is still an open problem, for which we have proposed a solution that can be improved. Contrasting information extracted from the web with that extracted from the private database is also an open problem, which has been solved for the moment by showing both (and their corresponding sources) if they are contradictory. Similarly, the evaluation of the freshness of the data, in order to distinguish current data from old data, is also an interesting topic for future work. Finally, we plan to perform an evaluation of the accuracy of the data extraction approach.

6. ACKNOWLEDGMENTS

This research work has been supported by the CICYT project TIN2013-46238-C4-4-R and DGA-FSE. We also thank the Heraldo de Aragón.

7. REFERENCES

[1] E. Agichtein and L. Gravano. Querying text databases for efficient information extraction. In *Data Engineering, 2003. Proceedings. 19th International Conference on*, pages 113–124. IEEE, 2003.

[2] H. Alani, S. Kim, D. E. Millard, M. J. Weal, W. Hall, P. H. Lewis, and N. R. Shadbolt. Automatic ontology-based knowledge extraction from web documents. *Intelligent Systems, IEEE*, 18(1):14–21, 2003.

[3] X. Carreras, L. Màrquez, and L. Padró. A simple named entity extractor using adaboost. In *Seventh Conference on Natural Language Learning at HLT-NAACL*, volume 4, pages 152–155. Association for Computational Linguistics, 2003.

[4] V. Chakravarthy, H. Gupta, M. K. Mohania, and P. Roy. Automatically linking documents with relevant structured information, 2011. US Patent 7,899,822.

[5] G. Chrupała and D. Klakow. A named entity labeler for German: exploiting wikipedia and distributional clusters. In *Conference on International Language Resources and Evaluation (LREC)*, pages 552–556, 2010.

[6] A. L. Garrido, M. Buey, S. Escudero, S. Ilarri, E. Mena, and S. Silveira. TM-Gen: A Topic Map Generator from Text Documents. In *25nd International Conference on Tools with Artificial Intelligence*.

[7] A. L. Garrido, M. G. Buey, S. Escudero, A. Peiro, S. Ilarri, and E. Mena. The GENIE project - a semantic pipeline for automatic document categorisation. In *10th International Conference on Web Information Systems and Technologies*. SCITEPRESS, 2014.

[8] A. L. Garrido, M. G. Buey, S. Ilarri, and E. Mena. GEO-NASS: A semantic tagging experience from geographical data on the media. In *17th East-European Conference on Advances in Databases and Information Systems (ADBIS)*, volume 8133, pages 56–69. Springer, September 2013.

[9] A. L. Garrido, O. Gomez, S. Ilarri, and E. Mena. NASS: News Annotation Semantic System. In *23rd International Conference on Tools with Artificial Intelligence*, pages 904–905. IEEE, 2011.

[10] J. R. Hobbs, D. Appelt, J. Bear, D. Israel, M. Kameyama, M. Stickel, and M. Tyson. Fastus: A cascaded finite-state transducer for extracting information from natural-language text. *Finite-State Language Processing*, page 383, 1997.

[11] C. S. Khoo, J. Kornfilt, R. N. Oddy, and S. H. Myaeng. Automatic extraction of cause-effect information from newspaper text without knowledge-based inferencing. *Literary and Linguistic Computing*, 13(4):177–186, 1998.

[12] G. Kobilarov, T. Scott, Y. Raimond, S. Oliver, C. Sizemore, M. Smethurst, C. Bizer, and R. Lee. Media meets semantic web–how the BBC uses DBpedia and linked data to make connections. In *The semantic web: research and applications*, pages 723–737. Springer, 2009.

[13] H. P. Luhn. *Auto-encoding of documents for information retrieval systems.* IBM Research Center, 1958.

[14] L. Maicher and H. F. Witschel. Merging of distributed topic maps based on the Subject Identity Measure (SIM) approach. *Proceedings of Berliner XML tags*, 4:301–307, 2004.

[15] A. McCallum. Information extraction: Distilling structured data from unstructured text. *Queue*, 3(9):48–57, 2005.

[16] R. J. Mooney and R. Bunescu. Mining knowledge from text using information extraction. *ACM SIGKDD explorations newsletter*, 7(1):3–10, 2005.

[17] H. Nanba, R. Saito, A. Ishino, and T. Takezawa. Automatic extraction of event information from newspaper articles and web pages. In *Digital Libraries: Social Media and Community Networks*, pages 171–175. Springer, 2013.

[18] Z. Nie, J.-R. Wen, and L. Yang. Web-scale entity summarization, 2012. US Patent 8,229,960.

[19] G. Salton and C. Buckley. Term-weighting approaches in automatic text retrieval. *Information Processing and Management*, 24(5):513–523, 1988.

[20] S. Sarawagi. Information extraction. *Foundations and trends in databases*, 1(3):261–377, 2008.

[21] S. Sekine and E. Ranchhod. *Named Entities: Recognition, Classification and Use*. John Benjamins, 2009.

[22] Y. Shinyama, S. Sekine, and K. Sudo. Automatic paraphrase acquisition from news articles. In *Second International Conference on Human Language Technology Research*, pages 313–318. Morgan Kaufmann Publishers Inc., 2002.

[23] F. M. Suchanek, G. Kasneci, and G. Weikum. Yago: a core of semantic knowledge. In *Proceedings of the 16th international conference on World Wide Web*, pages 697–706. ACM, 2007.

[24] W. Van Atteveldt, J. Kleinnijenhuis, and N. Ruigrok. Parsing, semantic networks, and political authority using syntactic analysis to extract semantic relations from dutch newspaper articles. *Political Analysis*, 16(4):428–446, 2008.