Extraction, Merging, and Monitoring of Company Data from Heterogeneous Sources

Christian Federmann, Thierry Declerck

Language Technology Lab German Research Center for Artificial Intelligence Stuhlsatzenhausweg 3, D-66123 Saarbrücken, GERMANY {cfedermann,declerck}@dfki.de

Abstract

We describe the implementation of an enterprise monitoring system that builds on an ontology-based information extraction (OBIE) component applied to heterogeneous data sources. The OBIE component consists of several IE modules—each extracting on a regular temporal basis a specific fraction of company data from a given data source—and a merging tool, which is used to aggregate all the extracted information about a company. The full set of information about companies, which is to be extracted and merged by the OBIE component, is given in the schema of a domain ontology, which is guiding the information extraction process. The monitoring system, in case it detects changes in the extracted and merged information on a company with respect to the actual state of the knowledge base of the underlying ontology, ensures the update of the population of the ontology. As we are using an ontology extended with temporal information, the system is able to assign time intervals to any of the object instances. Additionally, detected changes can be communicated to end-users, who can validate and possibly correct the resulting updates in the knowledge base.

1. Introduction

Due to the sheer endless amount of business information that are published online, there exists a growing demand for high-quality business intelligence (BI) tools, which can support rating agencies, banks, governmental organizations or publicly available information portals in the maintenance of their company information. The European R&D project MUSING (see http://www.musing.eu/ for more information) aims among others to respond to some of those issues in the area of financial risk management (FRM). For this, we are exploring new methods to reliably extract information on companies from the internet. The available data sources are of heterogeneous structure, ranging from free text contained in newspapers, loosely structured data such as company imprint websites, to more structured "info boxes" of Wikipedia articles or even DBpedia entries.¹

While the information extraction from the different sources is per se a challenging task, we moreover "aggregate" information from those sources and store the merged results as instances in our company ontology. Since information on companies is not static, we need to be able to detect changes over time and to monitor those. In order to respond to the intrinsically dynamic aspect of information about companies (and other entities involved in the business domain), our group has developed a temporal representation framework, which implements a perdurantist view of entities, and a corresponding time ontology (Hans-Ulrich Krieger, 2008). Both the temporal information extracted from the source and the date and time of the extraction processes are

¹Quoting from the English Wikipedia entry for DBpedia: "DBpedia is a community effort to extract structured information from Wikipedia and to make this information available on the Web. DBpedia allows users to ask expressive queries against Wikipedia and to interlink other datasets on the Web with DBpedia data.". In a sense, we cannot speak of information extraction from DBpedia entries, but rather of querying a structured semantic resource. attached to the instances we create in the ontology and so we can effectively build up a knowledge base about companies that is structured by temporal information.

Our paper describes the implementation of such an ontology-based enterprise monitoring system, which is capable of:

- 1. extracting and merging information about companies from heterogeneous data sources,
- 2. of detecting changes with respect to the current state of the knowledge base, and
- 3. of using the extracted and aggregated information to update the population the knowledge base.

2. System Overview

Figure 1 on the following page shows the basic components and the data workflow of our monitoring system. The following listing briefly summarizes the system architecture:

- **storage layer** we collect the imprint URL and some additional company information; data is stored either in our ontology or in a database, depending on usage.
- **information extraction** our IE tools take care of extracting, cleaning, aggregating and merging company information from heterogeneous sources.
- **monitoring module** extracted information is compared to the current state of our knowledge base to detect changes and/or new information.
- **ontology updates** updated information is used to increase knowledge stored inside the underlying ontology.
- **notification services** end users can choose to be notified whenever company information has changed.

Figure 1: Monitoring architecture overview



2.1. MUSING Ontologies

An integrated set of ontologies is guiding our information extraction (IE). This set has been developed with the help of domain experts using so-called competency questions (Gruninger and Fox, 1994) that supported the ontology engineers in the design and implementation of the domain ontologies. The resulting ontology structure then in turn helped to design and implement the IE tools. In our system, we went for a multi-layered architecture for the integration of the relevant ontologies:

- 1. A general level for "upper" ontologies. This layer responds to the needs of a foundational axiomatic approach, realized for example in the form of interval order axioms for the time ontology.
- 2. A standards level for adapting industry standards, following a model driven approach. For example, we have included an ontology for accounting principles, which is based on an ontology definition meta model applied to the XBRL taxonomies².
- 3. A domain level for ontologies relevant to one or more application domains. This layer needs a best practices approach (supported as well by the competency questions methodology).
- 4. A pilot level for classes and relationships specific or adapted to application needs. This layer needs explicit guidance and iteration cycles with partners responsible of actual applications—in our case the MUSING industrial partner primarily interested in the monitoring of company information.

We have implemented the ontology layer using Pellet (Sirin et al., 2007), OWLIM (Kiryakov et al., 2005), Sesame (Broekstra et al., 2002), and the Jena framework (HP, 2002).

2.2. Information Extraction

In our experiments, the MUSING IE tools extract a subset of information which is typical for companies, and which is described by the ontology classes. The IE tools are applied first on imprint web pages of German companies. We consider this to be a good first source for IE, since according to a German law (Telemediengesetz, TMG)³, companies have to publish compnay relevant information, like (amongst others):

- name of the company
- postal address
- legal form
- authorized executives

Company imprints are retrieved employing a method that searches the corresponding website using only a given base URL. This web page is then downloaded and converted to what we call *WebText*, a special text format that removes all HTML markup and normalizes whitespace and line breaks. Conversion to WebText helps to reduce pattern complexity and thus improves the overall performance and precision of the information extraction module.

After the company imprint has been cleaned up, we apply pattern matching to extract relevant information. Handwritten rules are employed resulting in a high precision of the extracted attributes. Our main focus is set on a high precision rather than a large recall as we want to minimize the need of having human operators involved. We can reliably extract the aforementioned points of interest for the tested set of German companies. It is worth noting that while part of the extracted company information could also be retrieved from central registers we have found that imprint websites usually contain more information at no cost. Retrieval from registers instead can become quite costly.

²XBRL stands for eXtended Business Reporting Language standard, see http://www.xbrl.org/ for more information.

³See http://www.gesetze-im-internet.de/bundesrecht/ tmg/gesamt.pdf for more information.

Figure 2: Extraction results for Adam Opel GmbH

```
<?xml version="1.0" encoding="UTF-8" ?>

      <url>http%3A//www.opel.de/legal/index.act</url>
     <crefo>
        <name>Adam Opel GmbH</name>
         <management>
           /management.
<manager>
<firstName>Reinald</firstName>
               <lastName>Hoben</lastName>
<function>Geschäftsführer</function>
           </manager>
           <manager>
              <firstName>Mark</firstName>
           >idstName>James</lastName>
<function>Geschäftsführer</function>
</manager>

           <manager>
<firstName>Holger</firstName>
              <lastName>Kimmes</lastName
              <function>Geschäftsführer</function>
           </manager>
            <lastName>McMillen</lastName>
<function>Geschäftsführer</function>
            </manager>
           <manager>
              <firstName>Alain</firstName>
              <lastName>Visser</lastName
               <function>Geschäftsführer</function>
           </manager>
           <manager>
<firstName>Walter</firstName>
              <lastName>Borst</lastName
              <function>Aufsichtsrat</function>
           </manager>
                   ement>
         </management>
<address>
<street>Friedrich-Lutzmann-Ring</street>
<postcode>65423</postcode>
            <place>Rüsselsheim</place>
         </address>
         <communication>
          ccommunication>
<fon>496142 770</fon>
<fax>+496142 778800</fax>
<email>kunden.info.center@de.opel.com</email>
<email>datenschutz.aogmbh@de.opel.com</email>
:/communication>
        <tradeRegister>
           <id>HRB 84283</id>
           <localCourt>Darmstadt</localCourt>
           <legalForm>GmbH</legalForm
        </tradeB
                     gister
          taxId>DE111607872</taxId>
      </crefo>
   </data>
</profile>
```

2.3. Extraction Results

Results are saved into XML files that conform to a XML scheme developed by one of the partners in the MUSING consortium. We transform these results into valuable input for the MUSING ontology using a mapping between the XML scheme and the ontology. For German car producer Adam Opel GmbH, website: http://www.opel.de/, we obtained the results given in figure 2. Please note: complex entries such as the manager names are available as several components, i.e. we do get first name, last name and position in different fields. All information is retrieved in a fully automatic manner.

3. Information Merging

Our system allows to merge company information extracted from various sources, such as imprint information, structured company profiles, or even newspaper snippets. It is also possible to combine other sources of company information with the extracted imprint data. For this, we implemented a merging module that takes two or more XML files and tries to combine them. The XML documents have to conform to our MUSING scheme in order to allow comparison. We have successfully applied merging to imprint data, DBpedia extracts and Wikipedia info boxes. It can also be extended to other information sources.

4. Information Monitoring

The IE tools analyse the imprint data of companies on a regular basis. The aim is here to detect changes in the information on a company, whereas the changes can be approved or rejected by a human operator. The current version of the merged company information from all extraction sources is compared to the last known revision contained within the knowledge base. Whenever any of the attributes has changed, a new revision is created and saved back into the data store. All revisions are labelled with the date and time of their creation, allowing us to keep track of the history of both companies and their management.

End users can choose to be notified whenever a change is detected. Updates are accessible via a dedicated RSS feed or using alert emails. Figure 3 shows an example of how such an email alert looks like. The alert contains some key information regarding the change event such as company name, id as well as the areas in which changes have been detected, e.g. *management board* or *address*. A clickable link to the source is also given to allow operators to easily verify the alert.

Figure 3: Example of a monitoring alert email



5. Ontology Update

As a final step of our workflow, the merged results are populating the MUSING ontology. Once the final merging result has been computed, we obtain another XML file that includes the updated company profile. Using a mapping from the XML scheme to the MUSING ontology schema, we can then submit the information to the ontology where it may update existing or even introduce new instances.

6. Evaluation

We have evaluated the performance of our enterprise monitoring system by setting up two evaluation tasks for German business information service provider Creditreform⁴ who was also part of the MUSING consortium.

6.1. Internal Evaluation

We first collected data on around 800 German companies and performed monitoring on a daily basis. Differences were used to update the underlying ontology and also generated alert emails that had to be checked by Creditreform.

⁴See http://www.creditreform.de/ for more information.

Figure 4: Information extraction evaluation results



Figure 4 shows the error distribution measured in the internal evaluation. Summarized, we observed that:

- a fraction of 6% of the companies did not have an imprint website and hence could not be monitored.
- around 7% of the companies produced errors.
- some 8% generated slight errors such as missing parts of the available information.
- the majority of 79% of all companies would produce perfect information extracts.

6.2. External Evaluation

For the external evaluation, we have set up an enterprise monitoring system for 800 different companies; seven companies from this second batch caused extraction problems and hence were dropped from the test set. Monitoring was performed once a week over a period of one month. In total, 709 monitoring alerts were sent to the Creditreform validators. This time, extracted information was also compared to the Creditreform database to effectively identify outdated information there. Monitoring alerts were checked by human operators who would normally collect data and update company profiles by hand.

Most monitoring alerts were accepted without remarks. Around 10% of the alerts were marked as "erroneous", another 20% produced minor errors such as superfluous or outdated information. Overall, our industry partner found the monitoring system helpful and called it successful.

7. Conclusion

We have presented the outlines of an enterprise monitoring system that relies on ontology-based information extraction, implements information merging from heterogeneous sources and which comes with a temporal representation mechanism. The system has been developed in collaboration with an industrial partner and was already evaluated with them. A main contribution of our work is about the possibility to extract from heterogeneous data sources relevant information and to merge it, before storing it in a persistent storage layer. In doing so, we put at disposal of a large number of potential users an updated set of information about a specific topic. One could think for example that our results could also be put at the disposal of Wikipedia (or other information portal), so that this it can achieve more consistency in the information on a company contained in its info boxes across different entries in different languages.

But this goes beyond the actual scope of MUSING, for which the main user of the Monitoring system is typically a rating agency or a bank. We have reported successful evaluation results for both an internal evaluation that measured IE quality as well as an external validation that proved the usefulness of such a monitoring system for our industrial partner.

Acknowledgements

We thank the anonymous reviewers for their comments regarding the initial draft version of this paper. We also want to thank our Creditreform partners for their efforts regarding the evaluation of the enterprise monitoring. This work was supported by the MUSING project (IST-27097) which is funded by the European Community under the Sixth Framework Programme for Research and Technological Development.

8. References

- Jeen Broekstra, Arjohn Kampman, and Frank van Harmelen. 2002. Sesame: A generic architecture for storing and querying RDF and RDF Schema. In Ian Horrocks and James Hendler, editors, *Proceedings of the first Int'l Semantic Web Conference (ISWC 2002)*, pages 54+, Sardinia, Italy. Springer Verlag.
- Michael Gruninger and Mark S. Fox. 1994. The role of competency questions in enterprise engineering. In *Proceedings of the IFIP WG5.7 Workshop on Benchmarking Theory and Practice*.
- Thierry Declerck Hans-Ulrich Krieger, Bernd Kiefer. 2008. A framework for temporal representation and reasoning in business intelligence applications. In Knut Hinkelmann, editor, AI Meets Business Rules and Process Management. Papers from AAAI 2008 Spring Symposium, Technical Report, pages 59–70. AAAI Press.
- HP. 2002. Jena a semantic web framework for Java. available: http://jena.sourceforge.net/index.html.
- Atanas Kiryakov, Damyan Ognyanov, and Dimitar Manov. 2005. OWLIM - a pragmatic semantic repository for Owl. In Mike Dean, Yuanbo Guo, Woochun Jun, Roland Kaschek, Shonali Krishnaswamy, Zhengxiang Pan, and Quan Z. Sheng, editors, WISE Workshops, volume 3807 of Lecture Notes in Computer Science, pages 182–192. Springer.
- E. Sirin, B. Parsia, B. Grau, A. Kalyanpur, and Y. Katz. 2007. Pellet: A practical Owl-DL reasoner. Web Semantics: Science, Services and Agents on the World Wide Web, 5(2):51–53, June.