Semantic Information Extraction on Domain Specific Data Sheets

Kai Barkschat
barkschat@fh-aachen.de

FH Aachen, University of Applied Science, Germany

Abstract. The development of information retrieval and extraction systems is still a challenging task. The occurrence of natural language limits the application of existing approaches. Therefore the approach of a new framework which combines natural language processing and semantic web technology is discussed.
This paper focuses on ontology based knowledge modelling for semantic data extraction. Therefore, semantic verification techniques which can be used to improve the extraction are introduced.

Keywords: #eswcphd2014Barkschat

1 Introduction and Motivations

More and more modern companies in the IT sector offer their services online in the World Wide Web (WWW). As a basis for such services, i. e. online market places or product comparison portals, these companies usually depend on data aggregation of huge and steadily growing information amounts from online resources.

According to the state of the art, companies use Extract-Transform-Load (ETL)[1] processes as the basis for their data processing. ETL consists of three stages, where the first task,"Extraction", is responsible for data aggregation. In our case we focus on extraction from electronic documents in PDF format. The "Transform" task defines the mapping between extracted data and the companies' internal models (for instance database schemas). The third task, "Load", intends to populate the internal storage with the prepared data amounts.

ETL approaches require rigid structures to operate on. This requirement is not fulfilled in many cases where product documents contain free form texts. Information defined by free text forms are hardly parsable by conventional approaches because of natural language problems: Among others, high complexity and strong ambiguity have the effect that textual descriptions of product data hardly resemble each other across documents.

As a result, free form texts have to be analysed before the relevant data can be identified and extracted. Today, this task is processed by human beings, because they are able to develop a semantic understanding of the text, and they are able to interpret the content in its context.

Another problem lies in legally liability issues. For example, a special domain of the energy sector deals with pricing of energy-tariffs. Some companies act as service providers for government institutions, whose task is to uncover illegal price fixing. The government makes companies responsible for the offered services to hedge itself against accusations.

To assure high quality standards during extraction, human manpower is indispensable, although the extraction task itself contains strongly repetitive parts.

A major drawback of manual extraction results in poor scaling of the total process. In contrast to the increasing amount of electronic documents, the extraction process is limited by available human manpower. The recruitment of additional staff is not a feasible option, as this would make the process economically unviable. Therefore, the described solution approach aims to relieve employees and also can help to reduce costs.

This paper describes a solution based on an ontology-based information extraction (OBIE) framework [2]. As starting point, product data sheets containing natural language product descriptions for two highly specialized domains (life sciences area, energy sector) are processed. We used ontologies to express domain specific knowledge about semantic relations and restrictions in the given domains. The key idea is to mimic human behaviour during manual extraction using this formal representation of knowledge. This enables the development of a machine processable text understanding for improving the data extraction.

The paper is structured as follows: Section 2 gives a brief review of ongoing research on semantic information retrieval and extraction systems. Section 3 shows a general overview of the described solution approach. Further, the approach is discussed in detail in Section 4. The evaluation plan is presented in Section 5. Finally, the results are summarized in Section 6.

2 Related Work

Wimalasuriya and Dou [2] give an overview of OBIE systems and fit them into the overarching theme of general information extraction (IE) systems. IE is known as the process concerning the analysis of natural language content, the identification, and the extraction of relevant information amounts. The authors clarify the importance of OBIE approaches, as they describe such systems as a bridging technology which combines text understanding systems and IE systems.

In general, OBIE systems use ontologies to model domain knowledge for a special area of interest. They can guide the extraction process because they define the relevant pieces of information and how these information can be identified during extraction. Additionally, ontologies can be exploited to express the semantic context as a graph based structure. According to this definition, the discussed approach of this paper represents an OBIE system.

Semantator [3] is a plugin-tool for the popular Protégé framework [4]. It supports the user during the annotation process. Similar to our approach, Semantator adds semantic annotations to natural language text. The possible annotation categories and relations are retrieved from domain specific ontologies. Annotated text entities are then written back to the ontologies as class instances or properties. Reasoning capabilities are proposed for detection of inconsistencies during annotation. In contrast to our approach Semantator does not focus on natural language processing (NLP) techniques although it can be connected with a few existing tools from that domain. Semantator lacks on automatic relation extraction which is treated by this paper. Additionally input documents often contain typographic structures which are not considered by this approach, but could be helpful for semantic extraction.

SREC [5] is a mathematical computation method for automatic detection of semantic relations in the ACE-2003 corpus [6]. It combines statistical and linear algebra calculation and centers on singular value decomposition. Although promising results are shown on the given corpus, too few relations are taken into account. For our task, this approach is too general. It lacks on relation types, which are sufficient for representing natural language (NL) structures. Semantic verification of extraction results is not considered and there is no mechanism to avoid or to detect false negatives: that are entities which are mistakenly marked as relation by the system.

Sbatella and Tedesco [7] present an advanced ontology driven semantic information retrieval system which uses a tripatite domain model for information extraction from plain text. This approach is focused on the highly specialized domain of the wine business. OWL-Ontologies are used to model the domain knowledge. Classes describe the important data categories, and properties describe the relevant relation between these categories. Text entities from source documents are then mapped to the categories and saved as instances of the OWL-classes. To recognize these entities the approach uses a combination of the lexical database WordNet [8] and several stochastic computations. Because the implemented framework always extracts plain text from different source documents, it is not possible to consider semi-structured content.

In 2007 Rusu et al. [9] published two different pseudo-algorithms for semantic triple extraction from typed dependency graphs and constituency based parse trees. The extracted triples in form of subject-predicate-object are suitable for mapping to standard ontology structures as defined by RDF [10] or OWL [11]. This forms a good basis for research projects as domain knowledge is implemented with ontologies increasingly common.

3 Architecture of Information Extraction System

The central research question of this project is: how can ETL processes be improved to automate the extraction process of product documents which contain natural language?

Zhang and Sidorov [12, 13] focus on statistical machine learning methods to analyse natural language content, and to identify relevant product data. Concerning liability obligations, pure statistically approaches are not sufficient to guarantee semantic cohesion. A validation mechanism is required to assure that the identified data entities are valid and do not contradict.

Therefore, it is necessary to develop semantic technologies which are able to automatically detect not only relevant data entities, but also semantic relations and the word sense within NL sentences via its context. The extraction results need to be secured from a semantic point of view.

Regarding the fact that most electronic product documents contain typographic structures and semi-structured parts close to the natural language content, investigating the influence of typographic structures for semantic IE improvement is another topic of this approach.

In the following section the overall scenario of our approach which intends to improve the existing ETL process is presented. The focus of this paper is on the OBIE task, which is highlighted in Figure 1. In the very beginning of the ETL process, data sheets are collected from multiple sources, like the web or via email communication. Given several data formats, e.g. PDF or HTML, these documents are then homogenized. The Open Document Text (ODT) format was chosen as the common data format, because it defines a large set of standardized markup elements for office applications, which are exploited to represent the structured and unstructured content of the aggregated data sheets. In addition to this, it is an open standard and can easily be expanded, if required.

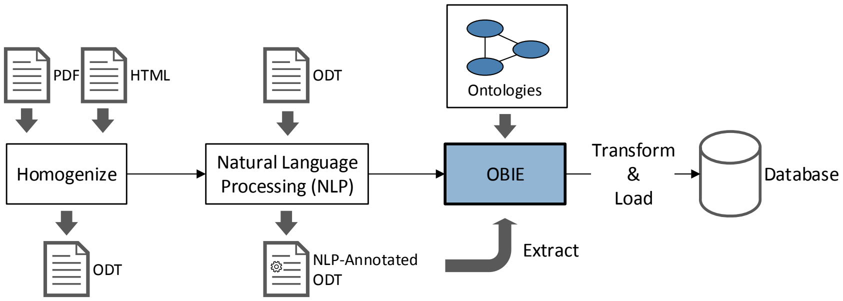


Fig. 1. The OBIE task embedded into the whole extraction process

In the second step, the ODT documents are passed to a NLP framework, which combines different tools to annotate the NL text content. Currently this framework uses implementations of OpenNLP [14] and Standford NLP [15] to generate annotations for the NLP tasks: Sentence Segmentation, Tokenization, Part-Of-Speech, Lemmatization, Named Entity Recognition (NER) and constituency based parse trees. These annotations are embedded into the ODT document and required by the third task (OBIE).

The OBIE task includes the domain knowledge in form of semantic graphs, which are defined as OWL ontologies [11]. Using this knowledge base on the one hand and the NLP annotations on the other, the OBIE process analyzes the document content, searches for relevant product data and tries to verify them.

As a note, the NLP process from the previous step already tried to mark relevant text named entities. Furthermore, we call these marked entities candidate concept (CC) and handle them purely as proposals for relevant product data. This is due to the fact applied NLP tools use neither semantic features nor context viewing for their annotation tagging.

Upon finishing, the OBIE process transforms the extracted data to a predetermined data schema and passes them to a persistent storage device like the database in Figure 1. Here it can be used by the company for building up services.

4 OBIE Approach in Detail

As already mentioned, the OBIE task takes in to account both natural language information and domain knowledge from the ontologies. Section 4.1 describes the preparation algorithm, which is used to identify triple structures in a NL sentence. These triple statements will then be matched against the domain model, which is described in Section 4.2. Section 4.3 and 4.4 explain how *semantic verification* can be done to secure and improve the extraction process.

4.1 Triple Extraction

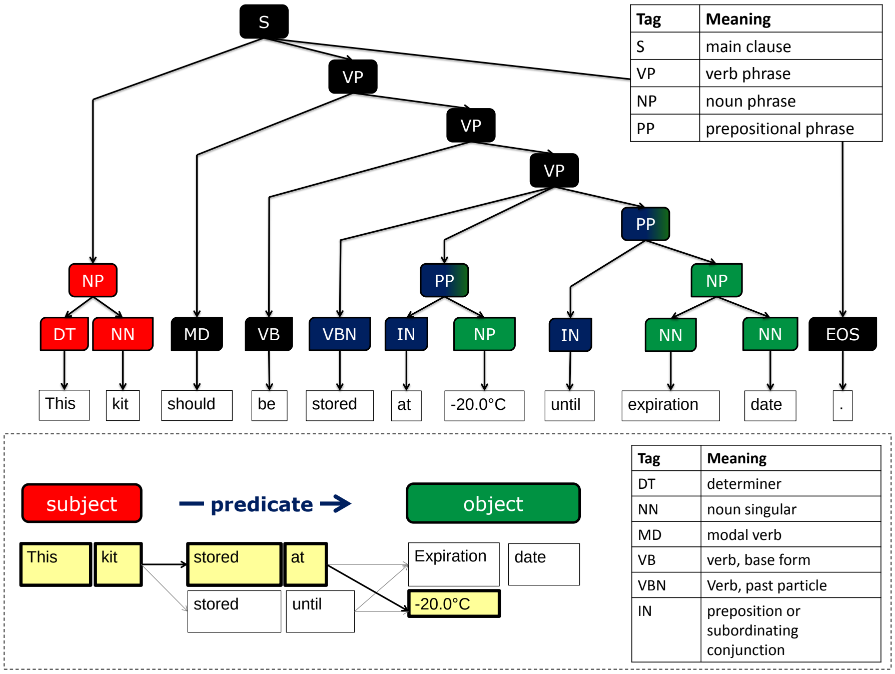
Starting from a constituency based parse tree, we use a slightly modified version of the “triple extraction algorithm” by Rusu et al. [9] to obtain possible text entities for the grammatical roles: subject, predicate and object.

An example for such a parse-tree and the application of the algorithm can be seen in Figure 2.

The main difference between this and the original algorithm is that the predicate element in the triple structure does not exactly match to its grammatical meaning. Indeed, the algorithm looks for the deepest verb descent (excluding modal verbs) in the parse-tree and tries to connect it with the subsequent associated preposition of the object element. This is required to identify the semantic relations between subject and object in the sentences.

Figure 2 demonstrates an example: *stored at* and *stored until* imply two completely different meanings in the sentence. The first case *stored at* indicates some kind of storage condition, which in the sampled domain is equivalent to a concrete temperature value. The second case *stored until* indicates an expiration date, a special subclass of the more generic data category *date*.

The semantic relations are important, because they form the edges or properties in the ontology based domain model and they are part of the triple structure, which is matched against the domain model later.



Tag	Meaning
S	main clause
VP	verb phrase
NP	noun phrase
PP	prepositional phrase

Tag	Meaning
DT	determiner
NN	noun singular
MD	modal verb
VB	verb, base form
VBN	Verb, past particle
IN	preposition or subordinating conjunction

Fig. 2. triple extraction on constituency based parse-tree

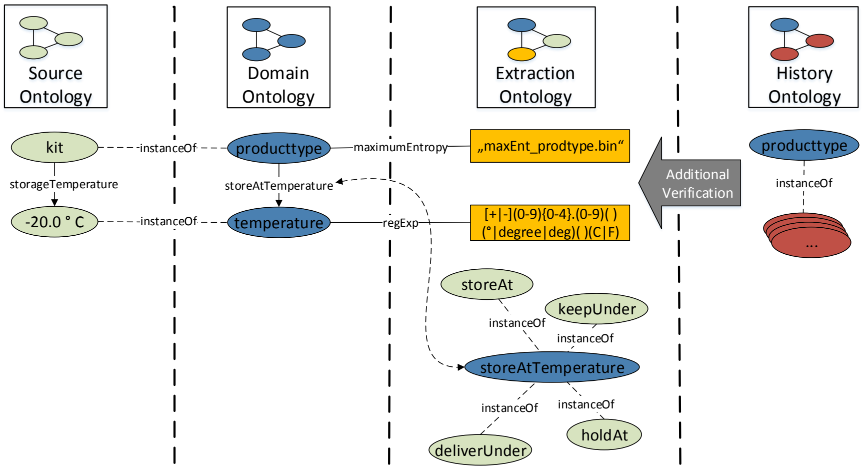


Fig. 3. Concept model for knowledge representation

4.2 The different roles of ontologies in the OBIE task

The ontology model of this approach is implemented in OWL and consists of four parts as seen in Figure 3. The domain ontology, the extraction ontology, the source ontology and the history ontology.

The domain ontology The domain ontology defines the important data categories which have to be extracted. In our setup we worked on high specific domains from the energy sector and the life sciences. Considering product descriptions from these areas, there were no existing ontologies, which could be reused to model the semantic setup of our data sheets. The domain ontology had to be created from scratch in coordination with a domain expert.

The semantic relations between the previously defined data categories are modelled as properties. There are fairly close restrictions, which can be modelled via cardinality constraints.

The extraction ontology The extraction ontology defines the knowledge about how to identify concepts of the domain ontology in natural language texts. Therefore, each data category is associated with a detection method. In the first attempt, we defined three different detection methods: maximum entropy, regular expression, and closed world list. As ongoing research, it is intended to compare the results of the different detection approaches. Our hypothesis is that detection quality of the different methods greatly varies, depending on the data category to be extracted.

The detection method closed world lists (CWL) can be seen as a classical dictionary lookup. It is limited to a finite set of text forms, e.g. the different units of amount in an operating manual for an experiment. The possible text forms are saved in a normalized form, also known as lemmatized form, as instances of the corresponding data category.

Regular Expressions are defined for concepts with strong similar text forms, such as telephone numbers or postal codes. Due to the fact of missing standards in our domains, there are a lot of further concepts which have strongly varying text forms.

The last detection method describes concepts which have to be trained with a more generic approach. We used a machine learning algorithm called “Maximum Entropy Algorithm” implemented by the OpenNLP framework here. Other statistical methods are conceivable.

Relations of the domain ontology are modelled in the extraction ontology in the same way of data concepts: each relation is defined as a seperate concept class. In contrast to normal concept classes, relation classes are always associated with CWL. This is needed for linking different descriptions in the natural language which carry the same meaning in the text. We intended to expand those lists with lexical knowledge databases like WordNet in further steps.

The source ontology A source ontology can be seen as the semantic representation of the associated source document. The concrete text entities from the source document are mapped to the triple structures and after semantic verification they are saved as instances to the concept classes from the domain ontology. Each source ontology contains the machine processable data content of one single source document. Instances of ontology classes can be seen on the left side in Figure 3.

The history ontology In our project setting, we are able to access large amounts of old product data. This is because the companies have already collected data from datasheets by their manual process. The results of the extraction by hand are stored in the companies' databases.

The history ontology defines a mapping from these already collected datasets to the OBIE process. The downside, however, is that there is no information about the source documents the data were extracted from. The history ontology will be populated using ontology-based data access (OBDA) mapping techniques as applied by Rodriguez [16].

We consider to use the history ontology as an additional knowledge source for the automated extraction process. The expectation is that it might improve the extraction as it can fulfill missing data entries via comparisons to older data.

For example, let us consider a supplier only sells products in a fixed package size. The package size could then be concluded, even if the data sheet itself does not contain any information about the size.

4.3 Semantic Verification on Natural Language

Using detection methods described in Section 4.2, we retrieve possible CC for the domain model. To verify and hedge these results a task called "semantic verification" is processed.

For each sentence it is checked if the concept candidates match the triple roles of a subject or object. If there is more than one match, the domain ontology is searched for related concepts of each concept candidate class. If the ontology contains a triple structure, whose outgoing and incoming concept node match the correct roles of the subject and object in the sentence, the relation itself has to be evaluated. Therefore, the second triple element, the predicate-preposition element, is queried against the CWL of the corresponding relation or property in the ontology. On success, we have a text based triple matching on a predefined ontology structure. This means additionally to the entity matching, we also verified the semantics of this single sentence as this semantic relation was predefined in the domain ontology. It is very likely that this step sorts out the CC which were mistakenly annotated by the NLP process. This is because in most cases, where wrong CC occur, the semantic context of the sentence contradicts that assignment.

Before such identified triples are stored in their lemmatized form in the source ontology, tools like OWL-Reasoner can be used to check if this insertion would violate any ontology constraint and would make the ontology inconsistent. In this case, the triple would be dropped.

4.4 Semantic Verification on typographic Structures

Because the input documents of our scenario are not plain text, but sometimes also contain semi-structured parts, we are able to semantically verify content regarding to its typographic context. For this task, we extended the extraction ontology to the indicator concept. Indicators are structures like headings, tables, or footnotes. Looking at the source documents of our domain, it has been observed that specific data concepts often occur in the same typographic context. For example, the components or parts of a given product usually are listed in tables and rarely seen in free form parts of the document.

Typographic structure dependent concepts are marked in the extraction ontology as additional properties, which relate to specific structure element concept classes. Those concept classes do not occur in the domain ontology, as this is specific extraction knowledge. Whenever a data concept is proposed for a text entity by the detection methods, it is checked if this concept also has relations to specific document structures. In this case the surrounding structure is evaluated against the related structure class. On match, a corresponding triple structure of the form component–indicatedBy–table is added to the source ontology. The indicator concept is especially helpful in cases where no closed sentences occur or where only fragments of sentences are present, e.g. tables. In these cases NLP results are not viable, as their CC proposals drops drastically.

5 Evaluation Plan

The described scenario builds up on ODT documents but these are not the original source documents of the domain. Indeed, documents have data formats like PDF, HTML, XLS. ODT is used as the common data format to enable a uniform processing. At the moment a small set of nlp-annotated ODT-documents is created to define valid input documents for the OBIE process. We call this “gold standard input documents for semantic extraction”. Gold standard in this term of use means that ODT documents are first checked on their typographical correctness comparing to the original sources and then enriched with proofed annotations by NLP. Finally the source ontology is populated with the correct data entries.

The evaluation of the semantic verification will be executed against this test set and compared to ETL extraction results which are generated without applying the OBIE task.

At the moment it is unclear which impact the typographic verification will have on the extraction process and how specific or general the typographic relation have to be. Possible documents might have to be clustered by producer, producttype or other criteria to retrieve the best results.

6 Conclusion

The described approach demonstrates how semantic verification can be applied to secure and validate data extraction based on entity proposals generated by state of the art tools. The inclusion of typographic elements to improve the data extraction has been barely inspected in previous research projects. Therefore, it is assumed that new insights will bring a deeper understanding for extraction of real world documents to the semantic web community.

References

1. Taniar, D., Chen, L., eds.: Integrations of Data Warehousing, Data Mining and Database Technologies - Innovative Approaches. Information Science Reference (2011)
2. Wimalasuriya, D.C., Dou, D.: Ontology-based Information Extraction: An Introduction and a Survey of Current Approaches. Journal of Information Science **36**(3) (June 2010) 306–323
3. Tao, C., Song, D., Sharma, D.K., Chute, C.G.: Semantator: Semantic annotator for converting biomedical text to linked data. Journal of Biomedical Informatics **46**(5) (2013) 882–893
4. Stanford Center for Biomedical Informatics Research: The Protégé Ontology Editor and Knowledge Acquisition System. http://protege.stanford.edu/
5. Zahedi, M.H., Kahani, M.: SREC: Discourse-level semantic relation extraction from text. Neural Computing and Applications **23**(6) (2013) 1573–1582
6. Mitchell, A., Strassel, S., Przybocki, M., Davis, J., Doddington, G., Grishman, R., Meyers, A., Brunstein, A., Ferro, L., Sundheim, B.: Ace-2 version 1.0. Linguistic Data Consortium, Philadelphia (2003)
7. Sbattella, L., Tedesco, R.: A novel semantic information retrieval system based on a three-level domain model. Journal of Systems and Software **86**(5) (2013) 1426–1452
8. Fellbaum, C.: WordNet: An Electronic Lexical Database. Language, Speech and Communication. MIT Press (1998)
9. Rusu, D., Dali, L., Fortuna, B., Grobelnik, M., Mladenic, D.: Triplet extraction from sentences. Proceedings of the 10th International Multiconference "Information Society - IS 2007" **A** (2007) 218–222
10. RDF Primer: W3C Recommendation 10 February 2004. http://www.w3.org/TR/rdf-primer/
11. OWL Web Ontology Language Overview: W3C Recommendation 10 February 2004. http://www.w3.org/TR/owl-features/
12. Zhang, S., Elhadad, N.: Unsupervised biomedical named entity recognition: Experiments with clinical and biological texts. Journal of Biomedical Informatics **46**(6) (2013) 1088–1098
13. Sidorov, G., Velasquez, F., Stamatatos, E., Gelbukh, A., Chanona-Hernández, L.: Syntactic N-grams as machine learning features for natural language processing. Expert Systems with Applications **41** (2014) 853–860
14. The Apache Software Foundation: Apache OpenNLP. http://opennlp.apache.org/
15. The Stanford Natural Language Processing Group: Stanford NLP. http://www-nlp.stanford.edu/software/index.shtml

16. Rodriguez-muro, M., Lubyte, L., Calvanese, D.: Realizing ontology based data access: A plug-in for Protégé. In: Proceedings of the Workshop on Information Integration Methods, Architectures, and Systems (IIMAS 2008), IEEE Computer Society Press (2008) 286–289