USING ASSOCIATION RULES TO LEARN CONCEPT RELATIONSHIPS IN ONTOLOGIES

Jon Atle Gulla, Terje Brasethvik and Gøran Sveia Kvarv
Department of Computer and Information Sciences
Norwegian University of Science and Technology, Trondheim, Norway

Keywords: Ontology engineering, text mining, association rules, domain modelling.

Abstract: Ontology learning is the application of automatic tools to extract ontology concepts and relationships from domain text. Whereas ontology learning tools have been fairly successful in extracting concept candidates, it has proven difficult to detect relationships with the same level of accuracy. This paper discusses the use of association rules to extract relationships in the project management domain. We evaluate the results and compare them to another method based on tf.idf scores and cosine similarities. The findings confirm the usefulness of association rules, but also expose some interesting differences between association rules and cosine similarity methods in ontology relationship learning.

1 INTRODUCTION

Traditional ontology engineering approaches are tedious and labor-intensive, requiring a wide range of skill sets as well as an ability to deal with very complex and formal representations. In the modeling process it is hard to manage and coordinate the contributions from various types of domain experts and ontology modelers. There are also technical, political and economical challenges that severely hamper the construction and maintenance of ontologies. At the same time, the ontologies are important in Semantic Web applications and integration projects, as they provide the vocabulary for semantic annotation of data and help applications to interoperate and people to collaborate.

Most ontology engineering methods today are based on traditional modeling approaches and emphasize the systematic manual assessment of the domain and gradual elaboration of model descriptions (e.g. (Cristiani & Cuel, 2005; Fernandez *et al.*, 1997)).

Ontology learning is the process of automatically or semi-automatically constructing ontologies on the basis of textual domain descriptions. The assumption is that the domain text reflects the terminology that should go into an ontology, and that appropriate linguistic and statistical methods should be able to extract the appropriate concept candidates and their relationships from these texts. Numerous approaches to ontology learning have been proposed in recent years (e.g. (Haase & Völker, 2005; Navigli & Velardi, 2004; Sabou *et al.*, 2007)), and they seem to allow ontologies to be generated faster and with less costs than traditional modeling environments.

Even though many of the approaches display impressive results, the complexities of ontologies are so fundamental that the generated candidate structures often just constitute a starting point for the manual modeling task. Advanced approaches with deep semantic analyses of text or whole batteries of statistical tests tend to yield better results, but are expensive to develop and may still not compete with traditional ontology modeling with respect to accuracy and completeness. So far, the best results are for the learning of prominent terms, synonyms and concepts. For more advanced constructions, like relationships and rules, there are still very few good tools out there to help us. Even though there are some ontology learning tools with relationship learning included, the accuracy of these relationships are questionable and there has only been limited work on comparing the various approaches to relationship learning. This is unfortunate, as there are indications that many of these approaches may be successfully combined into more reliable relationship learning approaches.

In this paper we present an approach to ontology relationship learning that makes use of association

Atle Gulla J., Brasethvik T. and Sveia Kvarv G. (2008).
USING ASSOCIATION RULES TO LEARN CONCEPT RELATIONSHIPS IN ONTOLOGIES.
In *Proceedings of the Tenth International Conference on Enterprise Information Systems - ISAS*, pages 58-65
DOI: 10.5220/0001686200580065

rules. The theory of association rules comes from data mining, though it can easily be adapted to the task of extracting relationships between concepts in domain text. The underlying idea is that concepts tend to be related if it can be shown that they show up together in documents with a certain predictability. The technique neither distinguishes between types of relationships nor identifies relationship labels, but gives a first rough set of candidate relationships to the ontology modelers.

The paper is structured as follows. Section 2 discusses the role of relationship learning in ontology engineering. We then introduce the association rules in Section 3 and briefly explain how they are used to extract relationships between concepts in Section 4. Section 5 introduces an alternative approach to relationship learning, using cosine similarities between concept vectors. An evaluation and comparison of the two approaches follows in Section 6, while some related work is discussed in Section 7. The conclusions are found in Section 8.

2 LEARNING ONTOLOGY RELATIONSHIPS

An ontology can be regarded as a representation of a set of domain concepts (also called classes or objects) and their relationships. The concepts may be taxonomically related by the transitive IS_A relation or non-taxonomically related by a user-named relation, for example, hasPart (Maedche & Staab, 2000). Some also make a distinction between non-taxonomic relations about whole/parts, class/instance or associations in general.

Web Ontology Language (OWL) is a semantic markup language recommended by the World Wide Web Consortium for the representation of ontologies. For the learning of relationships, OWL has four primitives of particular interest:

- Class: A class defines a group of objects or concepts that belong together.
- subClassOf: Stating that a class is a subclass of another gives us the ability to create generalization hierarchies of classes
- Property: Properties are used to define relationships between concepts. A property of a class Person, for example, can be hasChild or ownsCar.
- subPropertyOf: Hierarchies of properties can be useful in structuring the ontology for easy maintenance and extension. For example, the property hasRelative for a class Person may be specialized into the subproperty hasSibling.

Classes represent concepts that are taxonomically related, while properties define non-taxonomical relationships between concepts. Association rules do not distinguish between these two types of relationships and merely suggest relationships of some kind between two or more concepts. Moreover, the method is not able to derive any candidate names of the relationships identified.

Used in an ontology learning environment, association rules may give us a rough overview of potential relationships between concepts in the ontology. Other techniques or manual inspection are needed to categorize the relationships and – if needed – give them descriptive labels. Mapping approved relationships to the OWL constructs shown above, for example, still remains a manual task.

The technique may be used to relate already modeled concepts, but it usually includes a concept extraction pre-phase that identifies the concepts to be analyzed with association rules afterwards.

3 ASSOCIATION RULES FOR TEXT MINING

Association rules is a data mining techniques that identifies data or text elements that co-occur frequently within a dataset. They were first introduced in (Agrawal *et al.*, 1993) as a technique for market basket analysis, where it was used to predict the purchase behavior of customers. This was primarily done for large databases of items purchased on per-transaction basis. An example of such an association rule is the statement that *"90% of the transactions that purchased bread and butter also purchased milk."*

The problem in association rules mining can be formally stated as follows:

Let I be a set of literals, called items. Let D be a set of transactions, where each transaction T is a set of items such that $T \subseteq I$. A transaction T contains X, a set of some items in I, if $X \subseteq T$.

An assocation rule is an implication of the form

$X \Rightarrow Y$, where $X \subset I, Y \subset I, X \cap Y = \emptyset$

A rule $X \Rightarrow Y$ holds in the transaction set D with *confidence c* if c% of the transactions in D that contain X also contain Y. The rule $X \Rightarrow Y$ has *support s* in the transaction set D if s% of the transactions in D contain $X \cup Y$.

The idea is to generate all association rules that have support and confidence greater than a user specified minimum support and minimum confidence.

The most important algorithm for the generation of association rules is the Apriori algorithm, introduced in (Agrawal & Srikant, 1994). The algorithm finds all sets of items that have support greater than the minimum support. These sets are called *frequent item sets*. For every itemset *l* in the frequent itemset, L_k, it finds subsets of size *k-1*. For every subset X, it produces a rule $X \Rightarrow Y$, where $Y = l - X$. The rule is kept if the confidence

$$\text{support}(X \cup Y) / \text{support}(X)$$

is greater than or equal to the minimum confidence.

In a text mining context, association rules may be used to indicate relationships between concepts. Let us assume that an item set is a set of one or more concepts. If the rule $X \Rightarrow Y$ has been confirmed, we conclude that there is a relationship between the concepts in X and the concepts in Y. With item sets of size 1, we have rules that indicate relationships between two concepts.

In order to run association rule mining on text, we need to structure the text to mirror the situation in data mining. Following (Delgado *et al.*, 2002; Haddad *et al.*, 2000), we consider documents – rather than sentences or paragraphs – to correspond to transactions in data mining. Furthermore, we are only interested in extracting relationships between potential concepts, which means that we can restrict the analysis to noun phrases only. We reduce the noun phrases to their base forms, so that *project plans* and *project plan* count as the same term and only include noun phrases that have a certain prominence in the document set. We then have documents as item sets and lemmatized prominent noun phrases as items and can run a standard association rules analysis to extract relationships between these prominent noun phrases.

4 LEARNING RELATIONSHIPS FOR PROJECT MANAGEMENT ONTOLOGY

Our ontology learning tool is built as an extension to the GATE environment from the University of Sheffield (Gaizauskas *et al.*, 1996).

General Architecture for Text Engineering (GATE) is an open source Java framework for text analysis. It contains an architecture and a development environment that allows new components to be easily added and integrated with existing ones. The architecture defines the organization of a text engineering system, in which each component is assigned particular responsibilities. The framework comes with a set of built-in components that can be used, extended and customized to the specific needs of the analysis. This includes NLP components like tokenizers, POS taggers, sentence splitters and noun phrase extractors, but also more extensive plug-ins for multi-language stemming, WordNet retrieval, machine learning and ontology editors.

An analysis with GATE typically consists of a chain of components that one by one goes through the text and annotate it with information that will be needed by later components. With our own components for association rules added, we built the analysis chain shown in Figure 1 and explained in more detail below. The analysis is run on a repository of documents representative to the project management domain. Whereas the GATE components work on individual documents, we developed our own modules for association rules that pulled the individual files together, extracted prominent noun phrases as keywords, and suggested relationships between these phrases. The components of the chain are:

- *Tokenizer* and *Sentence splitter* are GATE components that split the document texts up in tokens and identify sentences for analysis. A token can be a simple word of something like a number or a punctuation mark.
- The *GATE tagger* is a statistical tagger that associates every word in the text with a part-of-speech tag.
- Having identified the parts-of-speech of a term, the *lemmatizer* can look it up in a dictionary and retrieve its *lemma*, or base form. The lemma is the common base form of all inflections of the same lexical entry, like the lemma process for verb forms like processes, processing, processed, etc.
- The *noun phrase extractor* identifies noun phrases in the text of the form Noun (Noun)*, i.e. phrases that consist of consecutive nouns. This means that a phrase like *very large databases* will not be recognized, since *very* is an adverb and *large* is an adjective, whereas *project cost plan* is a perfectly recognized phrase.

- The *noun phrase indexer* is responsible for indexing noun phrases found in the documents. After removing stopwords, the component extracts and counts the frequencies of noun phrases in the document set. A normalized term-frequency score (tf score) is used to select those prominent noun phrases that are most likely to be concepts in the domain. The result is a set of candidate concepts.
- The association rules miner uses the Apriori algorithm and extracts association rules between the noun phrases (concepts) found by the previous component. These association rules constitute possible relationships between ontology concepts of the domain.

The relationship learning tool was set up for the project management domain, using documentation from a petroleum company as domain text. The integration of GATE components with internally developed components was unproblematic, though the performance of the system would need to be improved for large-scale document collections. A 76 document collection was loaded in 0,88 seconds on average, and the complete analysis with this collection took 6 minutes and 57 seconds.

5 AN ALTERNATIVE RELATIONSHIP LEARNING METHOD

An alternative method to association rules is the traditional information retrieval approach with calculations of cosine similarities between concepts. Solskinnsbakk (Solskinnsbakk, 2007) presents an implementation of such a system for the same project management domain.

In this approach we make use of a vector of weighted terms for each concept in an already existing ontology. This concept vector is constructed on the basis of a domain text collection and contains words that tend to co-occur with the concept itself in the text. If the term *estimate* appears with weight 0.21 in concept *Cost*'s concept vector, it means that *estimate* and *Cost* are to some extent used in the same context (sentence, paragraph or document) and should display some semantic similarities. The weights are based on the tf.idf score, though we boost co-occurrences in the same paragraph and even more in the same sentence.

When all concepts are described in terms of concept vectors, we may calculate the relatedness between concepts using the cosine formula

$$cos(\vec{x},\vec{y}) = \frac{\sum_{i=1}^{n} x_i y_i}{\sqrt{\sum_{i=1}^{n} x_i^2}\sqrt{\sum_{i=1}^{n} y_i^2}}$$

where $\vec{x}$ and $\vec{y}$ are concept vectors and x_i is the weight of the i^{th} word of vector $\vec{x}$. If the cosine similarity is above a certain threshold, we conclude that there is a relationship between the concepts. The set of all cosine similarities above this threshold for all concepts pairs in the ontology is the system's suggested list of relationships in the domain.

6 EVALUATION

Evaluating ontology relationships learning systems is notoriously difficult, as there are potential relationships between all concepts and only subjective judgment can tell the important ones from the others.

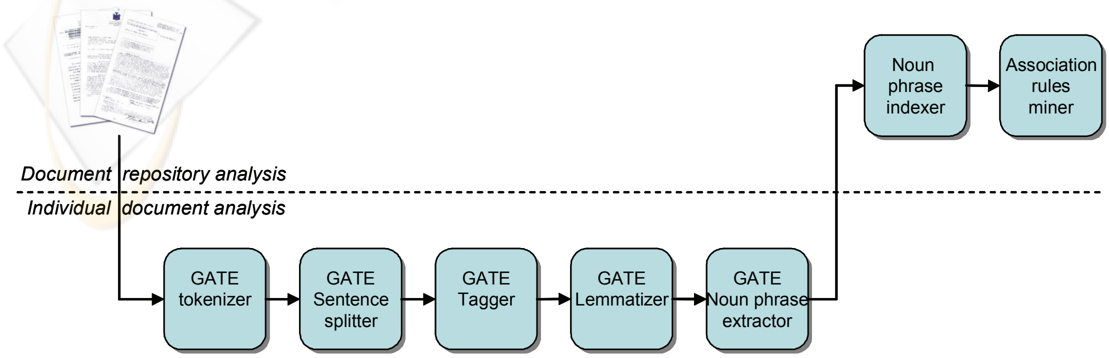

Figure 1: Process for extraction relationships using association rules.

A comparative evaluation of ontology relationship learning may be more interesting than absolute evaluations, as it may expose the differences between the systems and reveal to what extent they may be combined in hybrid approaches.

Concept Extraction. The domain chosen for the evaluation was project management in STATOIL, a large Norwegian petroleum company. They use a particular project management methodology, PMI, that is documented in handbooks and also reflected in project documentation from their own projects. Domain experts from STATOIL have together with ontlogy modelers built a project management ontology (Gulla *et al.*, 2007), which served as a gold standard for our concept extraction part.

Our association rules mining system was run on STATOIL's documentation of their project management methodology, PMBOK (PMI, 2000). This is a book of about 50.600 words (tokens) divided into 12 chapter.

The system extracted a total of 196 concepts, compared to the manually constructed ontology's 142 concepts. 50 concepts were identical in both sets, whereas some other 61 concepts found were abstractions of similar concepts in the manual ontology. If we assume that both the 50 perfect matches and the 61 abstract matches are valid, we have a precision of 56.7% and a recall of 78.2% for the concept extraction part.

Relationship Learning. For the relationship part, we compared the association rule approach to the cosine similarity system explained above. The manual ontology did not contain enough relationships to be of much use in this part of the evaluation. We first made a distinction between three types of relationships found by the two systems:

- Relationships suggested only by the association rule approach
- Relationships suggested only the cosine similarity approach
- Relationships suggested by both approaches

Slightly more than 50% of the relationships found were also identified by the cosine similarity method.

A selection of concepts were chosen. For each of the three groups above, all suggested relationships to/from these concepts were shown to four persons that all had project management experience. Each person individually rated each relationship as *not related* (these two concepts are not related), *related* (there is probably a relationship between the two concepts) or *highly related* (there is definitely a relationship between these two concepts). An average score for each relationship was calculated on the basis of the individual scores from the test persons. Figure 2 shows the related concepts suggested for the ontology concept *Cost* for the three groups, as well as their average scores.

Adding the results for all concepts together, we can compare the quality of relationships for the three groups. As shown in Figure 3, association rules and cosine similarities tend to produce the same share of good relationships (score *Related* and *Highly related*). The two methods suggested 82% and 86% good relationships, respectively, which is a fairly good result for such a small document collection. It should be noted, though, that this does not mean that they necessarily suggest the same relationships.

The share of very good relationships is worth a closer inspection. Whereas the association rules method only generated 7% very good relationships, the cosine similarity method reached an impressive 24%.

A possible explanation for this difference lies in the mechanics of association rules and cosine similarity. For an association rule to be generated, the corresponding concepts need to occur is a wide range of documents. This will typically be the case for very general concepts and their rather general relationships. The cosine similarity method, on the other hand, makes use of tf.idf to characterize concepts by their differences to other concepts, and the relationships based on cosine similarities will be based on these discriminating concept vectors. The relationships get more specialized and precise and are easier to recognize as very good relationships. This may also explain why the association rule method had a larger share of normally good relationships (75%) than the cosine similarity method (65%).

Interestingly, a combination of the two methods seems to produce much better results that each individual method. Both methods carry some noise, but our results indicate that this noise is dramatically reduced if we only keep the results that are common to both methods. In total, 97% of the relationships suggested by both methods were rated as good relationships by the test group (right column in Figure 3). 30% were considered very good relationships. This suggests that the two approaches – although comparable in quality – are fundamentally different with their own weaknesses and strenghts. Since overgeneration is already a problem in relationship learning, a better approach

Related concepts only from association rules		Related concepts only from cosine similarity		Related concepts from both methods	
project management team	R	cost management	HR	activity	R
management team	R	cost baseline	HR	assumption	NR
organization	R	actual cost	HR	control	R
product	HR	schedule	R	cost estimate	HR
information	R	project schedule	R	performance	R
tool	R	earn value	R	process	R
project team	R	staff	R	project	HR
application area	R	project staff	R	project management	R
risk analysis	R	milestone	NR	project objective	R
result	R	plan value	R	project plan	R
risk	R	stakeholder	HR	quality	R
resource	R	project deliverable	R	scope	R
consequence	R	ev	NR	scope statement	R
estimate	R	earn value management	R		
phase	NR	management	R		
probability	R	scope definition	NR		
action	R	scope management	R		
analysis	R	customer	R		
seller	HR	sponsor	R		
		project management IS	R		
		constraint	R		
		project manager	R		
		project plan development	R		
		procurement management	NR		
		project plan execution	NR		
		quality management	R		
		work breakdown structure	R		

Figure 2: Relationships suggested for *Cost*. The average scores are NR (not related), R (related) or HR (highly related).

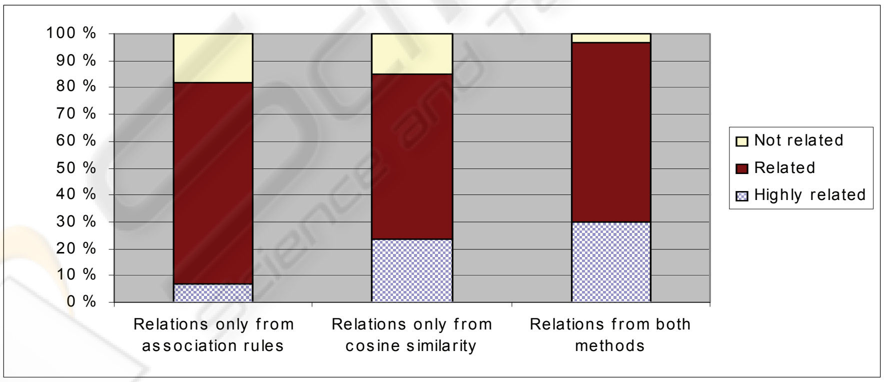

Figure 3: Evaluation results for three categories of relationships.

might be to combine approaches and only accept relationships that are supported by several methods. As far as association rules and cosine similarities are concerned, our research indicates that a combined approach will display substantially better results than each individual approach.

7 RELATED WORK

As discussed in (Cimiano *et al.*, 2006), there are today numerous ontology learning systems with facilities for learning relationships (see Figure 4). Association rules are already in use in some of these systems.

Systems	Synonyms	Concepts	Hierarchy	Relations
Text2Onto	clusters	X	X	X
HASTI			X	X
OntoBasis	clusters	clusters		
OntoLT/ RelExt			X	X
CBC/DIRT	clusters	clusters		
DOOBLE	X			X
ASIUM	clusters	clusters	X	X
OntoLearn	X	X	X	X
ATRACT	clusters	clusters		

Figure 4: Relationships learning in current systems.

Our approach to association rules is comparable to what can be find in other ontology learning tools. The accuracy of ontology relationship learning is still not very impressive and suffers from both over-generation and uncertainty. So far, the techniques have also failed in coming up with good labels for these relationships. Text2Onto has a particular structure, the Probabilistic Ontology Model (POM), that allows them to incrementally learn concepts and relationships (Cimiano & Völker, 2005).

Our approach draws on many of the ideas employed by Haddad et al. (Haddad et al., 2000). In their work they also use documents as transactions and focus on noun phrases as the carriers or meanings and the objects of analysis. A similar approach is taken in (Nørvåg *et al.*, 2006).

Our research is now focused on the integration of different relationship learning approaches. The combination of association rules and cosine similarity is promising, and has to our knowledge not been done before.

Another interesting application of association rules is presented in Delgado et al. (Delgado et al., 2002). Their idea is to use association rules to refine vague queries to search engine applications. After the search engine's processing of the initial query, their system weights the words in the retrieved documents with tf.idf and extracts an initial set of prominent keywords. Stopwords are removed and the remaining keywords are stemmed. Representing the stemmed keywords of each document as a transaction, their system is able to derive association rules that relate the initial query terms with other terms that can be added as a refined query.

Association rules have also been applied in web news monitoring systems. Ingvaldsen et al. (Ingvaldsen *et al.*, December 2006) incorporate association rules and latent semantic analysis in a system that extracts the most popular news from RSS feeds and identifies important relationships between companies, products and people.

8 CONCLUSIONS

This paper presented an ontology relationship learning approach that makes use of association rules to identify relationships between concepts. The approach is implemented as a text mining analysis chain, with GATE as the underlying architecture and our association rules components integrated with GATE through their standard API. As such, it is implemented as part of a comprehensive ontology learning workbench that also includes a battery of other ontology learning techniques.

Association rules provide a powerful and straight-forward method for extracting possible ontology relationships from domain text. The relationships extracted may be both taxonomic and non-taxonomic, though it is difficult to use the analysis alone to decide on the nature of the relationships. Of the relationships extracted for the project management domain, about 82% were considered valid relationships by a test group with previous experience in project management.

However, association rules do not seem to be substantially better than methods based on concept vector construction and cosine similarity calculations. The cosine similarity approach seems to generate more specialized and precise relationships than association rules. However, the methods are complementary, since association rules tend to focus on general relationships between high-level concepts and cosine similarity approaches focus on specialized relationships among low-level concepts.

We are now investigating to what extent several relationship learning techniques can be combined in an incremental learning strategy. An extension of the POM structure from Text2Onto may be useful in this respect, though we need to carry over more than just probability measures when these techniques are applied sequentially. The whole set of uncertainties, possibly supported by evidence in terms of vectors or raw calculations, need to come together in such a hybrid ontology learning framework.

REFERENCES

Agrawal, R., Imielinski, T., & Swami, A. N. (1993). Mining association rules between sets of items in large databases. In *Proceedings of the 1993 acm sigmod international conference on management of data*.

Agrawal, R., & Srikant, R. (1994). Fast algorithms for mining association rules. In *Proceedings of the 20th* *international conference on very large data bases (vlds'94)*.

Cimiano, P., & Völker, J. (2005). Text2onto. In A. Montoyo, R. Muñoz & E. Métais (Eds.), *Proceedings of the 10th international conference on applications of natural language to information systems (nldb'05)* (pp. 227-238). Allicante: Springer.

Cimiano, P., Völker, J., & Studer, R. (2006). Ontologies on demand? A description of the state-of-the-art, applications, challenges and trends for ontology learning from text. *Information, Wissenschaft und Praxis, 57*(6-7), 315-320.

Cristiani, M., & Cuel, R. (2005). *A survey on ontology creation methodologies*: Idea Group Publishing.

Delgado, M., Martín-Bautista, M. J., Sánchez, D., & Vila, M. A. (2002). Association rule extraction for text mining. In *Flexible query answering systems: 5th international conference (fqas'02)*. Copenhagen.

Fernandez, M., Goméz-Peréz, A., & Juristo, N. (1997). Methontology: From ontological art towards ontological engineering. In *Proceedings of the aaai'97 spring symposium series on ontological engineering* (pp. 33-40). Stanford.

Gaizauskas, R., Rodgers, P., Cunningham, H., & KHumphreys, K. (1996). Gate user guide. Url: Http://gate.Ac.Uk/sale/tao/index.Html#x1-40001.2.

Gulla, J. A., Borch, H. O., & Ingvaldsen, J. E. (2007). Ontology learning for search applications. In *Proceedings of the 6th international conference on ontologies, databases and applications of semantics (odbase 2007)*. Vilamoura: Springer.

Haddad, H., Chevallet, J., & Bruandet, M. (2000). Relations between terms discovered by association rules. In *Proceedings of pkdd'2000 workshop on machine learning and textual information access*. Lyon.

Haase, P., & Völker, J. (2005). Ontology learning and reasoning - dealing with uncertainty and inconsistency. In P. C. G. da Costa, K. B. Laskey, K. J. Laskey & M. Pool (Eds.), *Proceedings of the international semantic web conference. Workshop 3: Uncertainty reasoning for the semantic web (iswc-ursw'05)* (pp. 45-55). Galway.

Ingvaldsen, J. E., Gulla, J. A., Lægreid, T., & Sandal, P. C. (December 2006). Financial news mining: Monitoring continuous streams of text. In *Proceedings of the 2006 ieee/wic/acm international conference on web intelligence* (pp. 321-324). Hong Kong.

Maedche, A., & Staab, S. (2000). Semi-automatic engineering of ontologies from text. In *Proceedings of the 12th Internal Conference on Software and Knowledge Engineering*. Chicago, USA. KSI.

Navigli, R., & Velardi, P. (2004). Learning domain ontologies from document warehouses and dedicated web sites. *Computational Linguistics, 30*(2), 151-179.

Nørvåg, K., Eriksen, T. Ø., & Skogstad, K.-I. (2006). Mining association rules in temporal document collections, *Proceedings of the 16th International Symposium on Methodologies for Intelligent Systems (ISMIS'06)*. Bari.

PMI. (2000). *A guide to the project: Management body of knowledge (pmbok)*: Project Management Institute.

Sabou, M., VWroe, C., Goble, C., & Stuckenschmidt, H. (2007). Learning domain ontologies for semantic web service descriptions. *Accepted for publication in Journal of Web Semantics*.

Solskinnsbakk, G. (2007). *Ontology-driven query reformulation in semantic search.* Norwegian University of Science and Technology, Trondheim.