Ontological Profiles as Semantic Domain Representations

Geir Solskinnsbakk and Jon Atle Gulla

Department of Computer and Information Science
Norwegian University of Science and Technology
Trondheim, Norway
{geirsols, jag}@idi.ntnu.no

Abstract. Ontologies are conceptualizations of some domain, defining its concepts and their relationships. An interesting question is *how do we relate the ontological concepts to the general vocabulary of the domain?* In this paper we present the concept of ontological profiles, what they are, how they are constructed, and how they may be used. We propose that the ontological profile is a link between the vocabulary of the domain and its conceptual specification given by the ontology. This means that the ontological profile can be tailored to a specific document collection, reflecting the vocabulary actually used. Finally we demonstrate how the ontological profile may be utilized for ontology-driven search.

Keywords: ontological profiles, ontology-driven information retrieval, query reformulation

1 Introduction

Ontologies are now used in a wide range of applications and have been instrumental in many interoperability projects. They help applications work together by providing common vocabularies that describe all important domain concepts without being tied to particular applications in the domains.

Used as part of semantic search applications, however, the ontologies have so far had only limited success. Early semantic search engines tried to use ontology concepts and structures as controlled search vocabularies, but this was unpractical both functionally and from a usability perspective. Ontologies for query disambiguation or reformulation seem more promising, though there is a fundamental problem with comparing ontology concepts with query or document terms. Concepts are abstract notions that are not necessarily linked to a particular term. Sometimes there may be a number of terms that refer to the same concepts, and sometimes a specific term may be realizations of different concepts depending on the context. Using conceptual structures to index or retrieve document text requires that there is something bridging the conceptual and real world.

Another issue is the tailoring of ontologies to the retrieval task. Research indicates that ontologies are of little use if they are not aligned with the documents indexed by the search application. The granularity of the ontology needs

to match the granularity of the document collection. While there is no need to have an elaborated ontology for a sub domain with very few documents, it is often necessary to expand ontologies in areas that are well covered by the document collection.

This paper presents an ontology enrichment approach that both bridges the conceptual and real world and ensures that the ontology is well adapted to the documents at hand. The idea is to provide contextual concept characterizations that reveal how the concepts are referred to semantically in the document collection. The characterizations come in the form of weighted terms that are all - to some extent - related to the concept itself. The ontology together with the concept characterizations are referred to as an ontological profile of the document collection.

The approach has already been used for ontology alignment, and we are now experimenting with these profiles in search and ontology learning. Our initial search prototypes display a significant improvement of search relevance, provided that the quality of the characterizations are sufficient.

The structure of the paper is as follows: Section 2 gives a short overview of related work, while Section 3 deals with defining ontological profiles. Section 4 describes how such profiles are constructed, and Section 5 shows how the ontological profile may be used as a tool for enhancing information retrieval. Finally Section 6 concludes the paper.

2 Related Work

In the last years there have been many research projects concerned with semantic search, and we will here focus on applications that employ query expansion/reformulation techniques. [1] uses WordNet to expand the user query in the geographical domain. The query is expanded by POS tagging the query and expanding proper nouns with related words (synonymy and meronymy in WordNet). [2] describes a system that employs conceptual indexing (based on WordNet) and uses a variant of LSA to add conceptually similar words to the query. [3] describes a system that represents documents as a combination of concept instances and bag-of-words. The ontology is used at query time to disambiguate the query by presenting instances to the user. In [4] a system that employs conceptual query expansion is presented. Concepts are generated based on the top ranked document from a two-word (manually generated) query. A combination of words (represented as concepts) are added to the original query. [5] presents a ontology based search for portals. The system uses ontologies to contextualize and expand the user query with related words from the ontology. The new query is entered to Google through the Google API. The system that is most similar to the one we are developing is presented in [8]. This system is also based on building concept vectors, and the main difference lies in how these are constructed together with how they are used in the query expansion process.

3 Ontological Profiles

An ontological profile is an extension of a domain ontology. The ontology is extended with semantically related terms. These terms are added as vectors for each of the concepts of the ontology. This means that in the ontological profile each concept is associated with a vector of semantically related terms (concept vector). The terms are given weights to reflect the importance of the semantic relation between the concept and the terms.

Definition of concept vector The definition given here is adapted from [6]. Let T be the set of n terms in the document collection used for construction of the ontological profile. $t_i \in T$ denotes term i in the set of terms. Then the concept vector for concept j is defined as the vector $C_j = [w_1, w_2, \ldots, w_{n-1}]$ where each w_i denotes the semantic relatedness weight for each term t_i with respect to concept C_j.

We assume that the ontological profile is constructed on the basis of a document collection that covers the same domain as the ontology. By applying text mining techniques to the document collection we add terms that are semantically linked to the concepts of the ontology. Thus, each concept of the ontology is associated with one vector containing terms and weights that are specific to the concept. On an abstract level we may say that building an ontological profile is in fact building a weighted semantic dictionary, in which the concept vectors for each concept gives a list of terms and their weights that give an extended semantic characterization of the concept with respect to the document collection used as basis. We argue that the ontological profile, due to the construction based on a domain collection, gives a representation of the concept and its semantically linked terms that reflects how the concept is used in the language of the documents. This is an important point, since authors may have problems using a vocabulary consistent over a large domain. The concept vectors will typically contain terms that are synonymous to the concept and that are more indirect references to the concept or the use of it. One may argue that the use of thesaurus or WordNet may give much of the same information, but with one important difference. The information found in such formal sources are more general, and possibly not applicable to very specific concepts in large ontologies. Therefore we argue that the ontological profile possibly is better suited, since it is adapted to the document collection and the vocabulary used in it.

We argue that when applied to information retrieval (see Section 5) ontological profiles are generic to the search process. By this we mean that an information retrieval system that is built to utilize such profiles may be adapted to different document collections or even domains by exchanging the document collection and/or the ontology used. This is illustrated by Figure 1. We see that two different document collections are used to build two different ontological profiles for the ontology. By substituting the ontological profile the system can search more focused in a different document collection. We might also imagine that an

enterprise through the years have collected a large amount of documents, and the use of the vocabulary might have changed over time. Thus by building ontological profiles for certain time spans, these ontological profiles may provide bridging between the vocabulary used at different points in time.

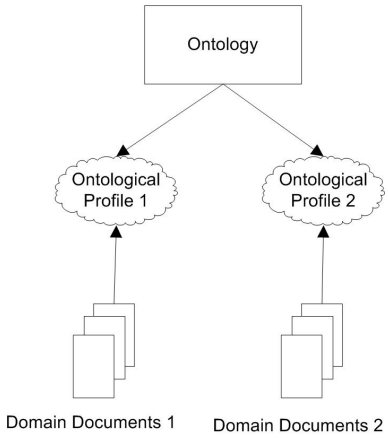

Fig. 1. The concept of ontological profiles.

Terms in the concept vectors are weighted in the range [0,1] where 0 means that there is no relation between the term and the concept, while 1 designates the term as being highly related to or even synonymous with the concept.

The terms contained in the concept vectors are semantic extensions of the concept. It could possibly be argued that the terms could be added as concepts to the ontology, further specializing the ontology. However, this is not necessarily desirable. First of all, the terms are very specialized, tailored towards the vocabulary of the underlying document collection, suggesting that they are not generic to the domain, and thus would only clutter the further use of the ontology with respect to other document collections. Secondly, the purpose of constructing the ontological profile is to do a more deep semantic analysis of the document collection, finding relations that are found in the documents, but that need not be generic to the domain. The terms are not generic enough, and are to fine-grained to be used as concepts in the ontology.

The construction of these ontological profiles is based on three different aspects of the content of the documents used. The first is that we apply statistical techniques, counting the frequency of the terms in the documents. Terms that co-occur with a concept more frequent are hypothesized to be more relevant for a concept than terms that do not co-occur as frequently. The second is that we apply linguistic techniques, i.e. stemming, to collapse certain terms into a single form. The third aspect is that we use a proximity analysis of the text. The assumption that lies behind the proximity analysis is that the closer terms

are found in the text, the more semantically related they are. These three aspects of the underlying document collection are the basis for the construction of ontological profiles that we suggest in Section 4.

Finally we will in this section show an example of a concept vector. We will be using the concept *Christmas Tree* from the IIP ontology [12]. The setup of the experiment can be found in Section 5.3. A *christmas tree* (see Figure 2) is by the ISO 15926 standard used in the petroleum industry defined as "*an artefact that is an assembly of pipes and piping parts, with valves and associated control equipment that is connected to the top of a wellhead and is intended for control of fluid from a well.*" The top 10 terms for the concept vector are shown below:

$$C_{christmastree} = [christma_{0.71}, tree_{0.60}, valve_{0.13}, master_{0.11}, wing_{0.11}, \\ bop_{0.08}, located_{0.08}, stack_{0.07}, choke_{0.07}, wellhead_{0.06}]$$

Note that we have applied stemming, resulting in christmas being stemmed to christma. The first two terms in the vector are the constituents of the concept name, and have also received the highest relevance score. The terms valve, and wellhead are clearly related to the concept (as we may note from both the definition and Figure 2). Master is also contained in the vector, and looking at Figure 2 we see that several valves are referred to as "master" valves. Bop (an abbreviation for blowout prevention) is certainly relevant, although it is not mentioned in the definition. This term demonstrates in a good way that terms are picked up in the process of building the ontological profile based on semantic relations. A point that is worth mentioning (although more a implementational issue) is that the vector does not contain any phrases (for instance the concept name is split into two separate terms). Adding phrases to the vectors would add even more semantics to the concept vector, and is an issue that will be addressed in the next stage of research.

4 Construction

The ontological profile is constructed on the basis of a domain relevant document collection. This assures that the connection we want between the vocabulary of the domain (at least the vocabulary in the document collection used) and the concepts of the domain is found, letting the ontological profile be a semantic characterization of the domain. A detailed description of the approach we have used for the construction of the ontological profile is found in [7], and is based on a method described by [6]. [8] describes another approach for the construction of ontological profiles (referred to as feature vectors).

The overall process of constructing the ontological profile is shown in Figure 3. The first step is to preprocess the documents used during the construction phase. During this process we remove all stop words, and stem the terms lightly

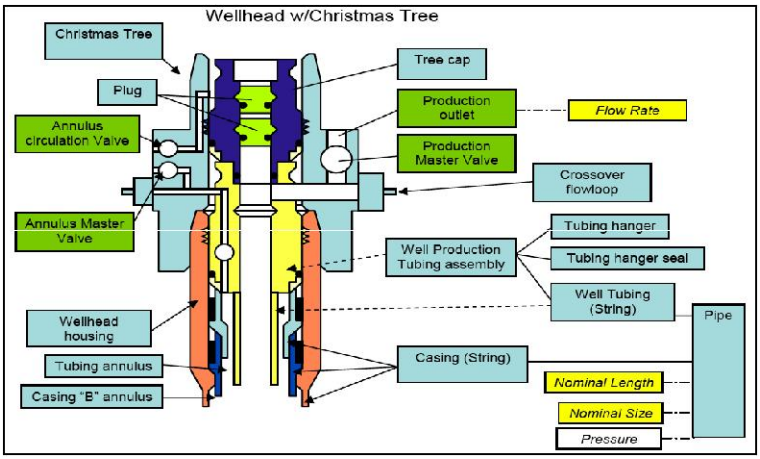

Fig. 2. Schematic view of a *christmas tree* [12].

(removing plural s), using the conversion $s \rightarrow \emptyset$ for all terms not terminated with *ss*. A light stemming algorithm is chosen to reduce the effect of decreased precision which is a problem with stronger stemming algorithms (e.g. the Porter stemming algorithm) [9].

Next, we build three separate indexes of the relevant documents, reflecting three different semantic views of the documents. In the first case, the whole document is viewed as a set of semantically related terms. The second case splits the documents into paragraphs, where each paragraph is considered a semantic entity in which the terms are closer related semantically than in the document. We have chosen to split the paragraphs at the boundary of two or more consecutive line breaks. The last case is where the document is split into sentences, and each sentence is considered a semantic entity. Terms found in a single sentence are considered to be even closer semantically related than in the paragraph. We have used punctuations as the boundaries between the sentences (".","!", and "?"). Thus we have formed a hierarchy of increasing semantics over the text. Once the documents have been split according to our schema we construct three separate indexes, one for whole documents, one for paragraphs, and one for sentences. The indexes are constructed based on Apache Lucene[1] and the vector space model.

The next step in the construction of the ontological profile is to assign to each of the concepts in the ontology the set of relevant documents (whole documents, paragraph documents, and sentence documents). We use the concept name as a phrase query into the three indexes, and all documents containing the phrase are assigned to the concept as relevant. Of course, using the concept name as a phrase query into the three indexes imposes a challenge; some of the concept

[1] http://lucene.apache.org/

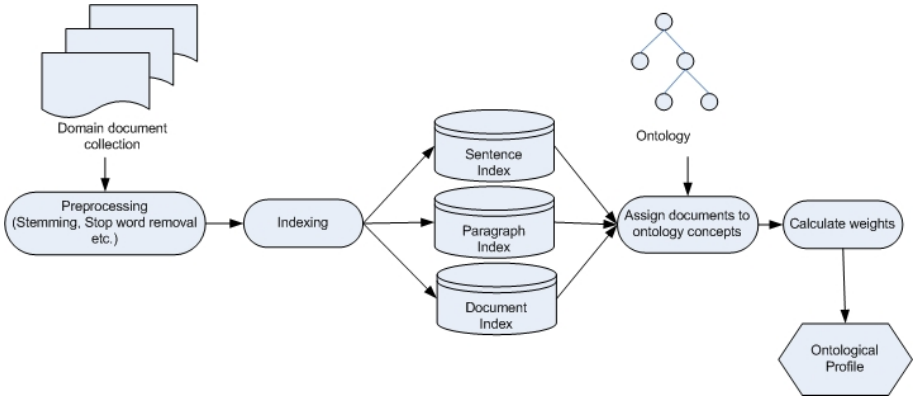

Fig. 3. The process of constructing the ontological profile.

names are artificial in their construction or are not used in the form given in the concept. This means that many of the concepts are not found during the assignment of documents to the concepts. We have not researched how to handle this, so this is a matter that needs further research.

The final step in the construction of the ontological profile is to calculate the weights for the terms assigned to the concepts. Recall that the previous step assigned all documents found to be relevant (i.e. contains the concept name as a phrase) to the concept. This means that all terms found in the relevant documents are also assigned to the concept. Having the text partitioned into three different views, we use this partitioning to boost terms that are found closer to the concept in the text. In effect this means that we give the highest weight to terms that are found in the same sentence as the concept name phrase (the highest semantic coherence), terms found in the same paragraph as the concept are given lower weight than sentence - terms, and higher than document terms. The basis for the weight calculation is the term frequency for each term found in the relevant documents. Equation 1 shows the calculation, where $vf_{i,j}$ is the term frequency for term i in concept vector j, $f_{i,k}$ is the term frequency for term i in document vector k, D, P, and S are the possibly empty sets of relevant documents, paragraph documents and sentence documents assigned to j, and $\alpha = 0.1$, $\beta = 1.0$, and $\gamma = 10.0$ are the constant modifiers for documents, paragraph documents, and sentence documents, respectively. The modifiers (α, β, and γ) simply reflect the relative importance of terms found in whole documents, paragraph documents, and sentence documents. Although the absolute numerical value for these have not been researched extensively, we have found that the set of modifiers shown perform quite well.

$$vf_{i,j} = \alpha \cdot \sum_{d \in D} f_{i,d} + \beta \cdot \sum_{p \in P} f_{i,p} + \gamma \cdot \sum_{s \in S} f_{i,s} \tag{1}$$

The vectors resulting after the calculations in Equation 1 are what we refer to as the basic vectors. Applying the familiar tf*idf [10] score to the frequencies we get closer to the final representation of the vectors. The idf factor gives more importance to terms that are found in few documents across the document collection, and is used in an analogue way here. The only difference is that we now use the basic vectors as our documents, meaning that the idf factor gives higher weight to terms that are found in few concept vectors. The calculations are shown in Equation 2, where $tfidf_{i,j}$ is the tfidf score for term i in concept vector j, $vf_{i,j}$ is the term frequency for term i in concept vector j, $max(vf_{l,j})$ is the frequency of the most frequent occurring term l in concept vector j, N is the number of concept vectors, and n_i is the number of concept vectors containing term i.

$$tfidf_{i,j} = \frac{vf_{i,j}}{max(vf_{l,j})} \cdot \log \frac{N}{n_i} \tag{2}$$

The ontological profile is now complete, although we may apply some final normalization, such as normalizing the vectors to unit length to ensure that prominence within the vectors is reflected when comparing several vectors with the same term. The last step is to index the vectors so that the vectors may be searched both by concept name and by terms (resulting in a ranked list of concepts).

In [11] we also introduced the notion of negative concept vectors, which are built in the same way but contain only terms that are not relevant for the domain. The documents used to build these negative concept vectors are strictly non-relevant to the domain.

5 Ontological Profiles in Search

This section will give a description of how the ontological profile may be used in a search application. Our approach to using the ontological profile in search, is to use it as a tool for semantic reformulation of queries on top of a standard vector space based search engine (we use Lucene), using the reformulated query as a query into the index. This approach lets the system hide from the user the fact that an ontology is used, and the user is only faced with entering familiar keyword queries.

The query reformulation process is based on two steps. The first is to interpret the user query, in effect this means that we map the user query on to a set of one or more concepts in the ontology. The second step is to expand the query with semantically related terms, i.e. we use the concepts from the first step. These steps will be described in the following.

5.1 Query Interpretation

The goal of the query interpretation is to map the user query (keywords supplied by the user) on to a set of one or more concepts of the ontology. In this mapping

process we use the ontological profile as a measure of the coherence between the concepts and the users query terms. We have suggested four such query interpretation techniques which produce a ranked list of concepts to choose from during the query expansion. Recall that the ontological profile contains for each concept a list of terms and their weights showing their importance for the concept. In effect we try to find the set of concepts that maximize the semantic coherence between the query and the concepts.

Simple query interpretation The simple query interpretation is the most basic schema, mapping each query term to a single concept. For each query term we find the concept which has the highest tf*idf score for that term in its vector. The concept with the highest tf*idf score is subsequently picked as the semantic representation of the query term and picked for expansion in the next step.

Best match query interpretation In the simple interpretation scheme we do not consider any relations between the user entered query terms. The best match approach assumes that there is in fact a relation between the user entered query terms, and tries to recognize this relationship by attempting to map the user query terms on to a single concept which has a good representation of the query terms collectively. This is done by requiring the candidate concepts to contain all query terms. The concept which maximizes the score given in Equation 3 ($ccore_c$ is the score for concept c, $t_{i,c}$ is the tf*idf score for query term i in concept c) is picked for subsequent expansion.

$$score_c = t_{0,c} + t_{1,c} + \ldots + t_{n-1,c} \tag{3}$$

Cosine similarity query interpretation The last two interpretation schemas are attempts at disambiguating the user query. This is done by attempting to recognize relations between the query terms and the concepts they map to. There could possibly be a relation between the first concept for term 1 and the third concept for term 2, which is not recognized by the simple interpretation. In the cosine similarity approach we use the cosine similarity between concept vectors as a measure of the relationship between the concepts. The first part of the interpretation is in fact similar to the simple approach, in which for each query term we generate a ranked list of the 15 most related concepts. For each pair of concepts we calculate a score (Equation 4, cin is the concept ranked as n with respect to query term i) which takes into account both the score for each term and the relationship between the concept vectors. The pair of concepts with the highest score will be picked as the semantic representation of the query and used for subsequent expansion.

$$score_{cin,cjm} = t_{i,n} * t_{j,m} * cos\_sim(cin, cjm) \tag{4}$$

Due to complexity we have only implemented and tested the last two approaches for two query terms.

Ontology structure query interpretation This query interpretation approach is similar to the cosine similarity interpretation, but uses a different measure for the similarity between the concepts. The similarity measure is based on the structure of the ontology, where we use the distance between the concepts as a measure of the similarity. We rely on the assumption that concepts that are close in the ontology have a higher semantic coherence than concepts that are further apart. We generate a graph structure representation of the ontology in which the concepts are nodes and the relations are edges. This graph is then traversed to find the distance for each of the concept pairs. As for the cosine similarity based approach, we generate a ranked list of the 15 most related concepts for each of the query terms. The score for each pair of concepts is calculated as shown in Equation 5 where the path function gives the distance between the concept pairs. The pair of concepts with the highest score is picked as the semantic representation of the query and used in the subsequent expansion process.

$$score_{cin,cjm} = t_{i,n} * t_{j,m} * \frac{1}{path(cin, cjm)} \tag{5}$$

5.2 Query Expansion

The query expansion process reformulates the original query by adding semantically related terms to the query. This is done by adding the top 15 terms to the original query with their weights. This means that the final query is a weighted query, and the original query terms are boosted to signal that these are the most important terms in the query. In addition we use the negative concept vectors to add for each concept the top 15 negative terms to the query. These terms are added as NOT terms, signaling that any documents containing these terms should not be included in the result set. This takes care of removing some of the noise introduced by the increased set of query terms. Finally the weighted query is fired against a vector space search engine (Lucene) and the result set is returned to the user.

5.3 Results

There are two main approaches to evaluating the usefulness of ontological profiles. The first is to evaluate it in an information retrieval system, and the second is to evaluate the concept vectors with respect to reflecting the quality of the concept vectors. First we give a short description of the evaluation of the ontological profile used in information retrieval (based on the approach described in Section 5). We have used the IIP [12] core ontology, which contains 18,675 concepts, covering the domain of oil production and drilling in subsea conditions. For the construction of the ontological profile we used the Schlumberger Oilfield Glossary [2] and successfully created concept vectors for 2,195 concepts. Our evaluation, which consisted of 7 queries and 5 test subjects, revealed that

[2] http://www.glossary.oilfield.slb.com

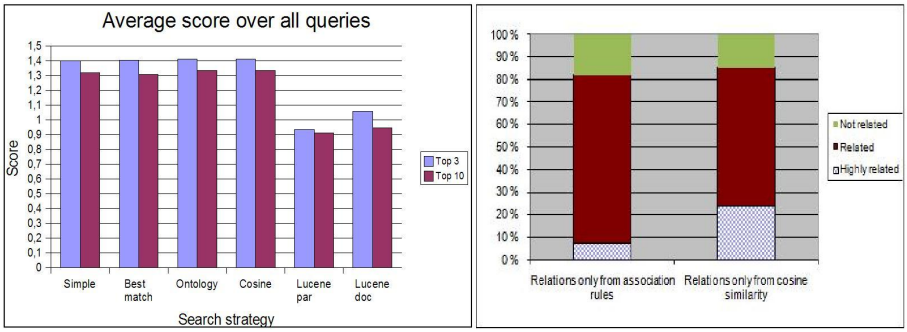

Fig. 4. Left: Evaluation of ontological profiles in search [13] Right: Evaluation of concept relations [14].

all four of our reformulation strategies performed significantly better than pure keyword search (see Figure 4, left). One surprising result was that all four of the strategies performed very equally, none of them stood out in any way with respect to query type. This suggests that the structure of the ontology is not as important as we first thought. Although the evaluation is not statistically significant, it gives an impression of the performance of our strategy. For a more thorough description of the evaluation, see [13].

As we introduce new terms to the query, we hypothesize that the reformulated queries enhance the recall of the system, as a broadened set of query terms will retrieve more documents. We have however not investigated how precision is affected by the proposed search strategy. The quality of the vectors is of critical importance for the system to perform well, as is also indicated by [8]. Further work includes tweaking of parameters and giving the system a more thorough evaluation.

The second evaluation considers the quality of the vectors. We did an evaluation (in the project management domain) in which we evaluated the quality of the relationships found by calculating the cosine similarity between the concept vectors. With relationships we mean not necessarily the relationships found in the ontology, but the relationships found among the concepts based on the concept vectors. The right part of Figure 4 show that the relationships found were generally of higher quality than those found by other means [14].

6 Conclusion

We have in this paper presented the concept of ontological profiles, and how they are constructed. We argue that they are more powerful semantic representations of a domain than an ontology is on its own. Text mining techniques are employed to extend the ontological concepts with terms that are semantically linked to the concepts, and weights showing the strength of these relations. Finally we

showed how a search system based on ontological profiles may be constructed. The evaluation of the search prototype showed promising results, and we will continue research on ontological profiles for search applications.

Acknowledgment. This research was carried out as part of the IS_A project, project no. 176755, funded by the Norwegian Research Council under the VERDIKT program.

References

1. D. Buscaldi, P. Rosso, E.S. Arnal. A wordnet-based query expansion method for geographical information retrieval. Working noted for CLEF workshop, 2005.
2. R.Ozcan and Y.A. Aslangdogan. Concept based information access using ontologies and latent semantic analysis. Technical report cse-2004-8, University of Texas at Arlington.
3. G. Nagypal. Improving information retrieval effectiveness by using domain knowledge stored in ontologies. OTM Workshops 2005, LNCS 3762, pages 780- 789. Springer-Verlag, 2005.
4. T.P. Weide, F.A. Grootjen. Conceptual Query Expansion. Data & Knowledge Engineering, (56):174-193, 2006.
5. W.A. Pinheiro, A.M. de C. Moura. An Ontology Based-Approach for Semantic Search Portals. Proceedings of the Database and Expert Systems Applications, 15th International Workshop on (DEXA'04) - Volume 00, IEEE Computer Society. 2004.
6. Su, X.: Semantic Enrichment for Ontology Mapping. PhD Thesis, Norwegian University of Science and Technology, Trondheim, Norway, 2004
7. Solskinnsbakk, G.: Extending Ontologies with Search-Relevant Weights. Technical Report, Norwegian University of Science and Technology, Trondheim, Norway, 2006.
8. Tomassen, S.L., Gulla, J.A., Strasunskas, D.: Document Space Adapted Ontology: Application in Query Enrichment. In: 11th International Conference on Applications of Natural Language to Information Systems (NLDB 2006), Vol. 3999. Springer-Verlag, Klagenfurt, Austria (2006) 46-57
9. Frakes, W.B. and Fox, C.J.: Strength and similarity of affix removal stemming algorithms. SIGIR Forum, 37(1):26-30, 2003.
10. Baeza-Yates, R., Ribeiro-Neto, B.: Modern Information Retrieval, ACM Press, New York, 1999.
11. Solskinnsbakk, G.: Ontology-Driven Query Reformulation in Semantic Search. MSc Thesis, Norwegian University of Science and Technology, Trondheim, Norway, 2007.
12. Gulla, J.A., Tomassen, S.L., Strasunskas, D.: Semantic interoperability in the Norwegian petroleum industry. In Karagiannis, D., Mayer, H.C.(Eds.), 5th International Conference on Information Systems Technology and its Applications (ISTA 2006), volume P-84 og Lecture Notes in Informatics (LNI), pages 81-94. Köllen Druck Verlag GmbH, Bonn, Klagenfurt Austria, 2006.
13. Solskinnsbakk,G. and Gulla, J.A.: Ontological Profiles in Enterprise Search. Submitted to EKAW 2008 - 16th International Conference on Knowledge Engineering and Knowledge Management, 29th September-3rd October 2008 - Acitrezza, Catania, Italy.
14. Gulla, J.A., Brasethvik, T., Sveia Kvarv, G.: Using Association Rules to Learn Concept Relationships in Ontologies. Accepted for publishing at ICEIS 2008: 10th International Conference on Enterprise Information Systems 12 - 16 June 2008, Barcelona, Spain.