An Approach to Semantic Query Expansion

Juraj Malečka and Viera Rozinajová

Institute of Informatics and Software Engineering
Faculty of Informatics and Information Technologies
Slovak University of Technology in Bratislava
Ilkovičova 3, 842 16 Bratislava, Slovakia
{malecka,rozinajova}@fiit.stuba.sk

Abstract. Reasoning and semantic ontology searching is one of the main benefits of the Semantic Web but still remains a challenge. Explicit conceptualization of a domain which contains all the relationships between concepts brought the possibility of expanding the user's query along these relations. However, ontologies could be inconsistent or could not contain every logical relationship. Various techniques to overcome this problem are known but there is not any algorithm which combines them together. In this paper we introduce a method for semantic ontology searching which combines the approaches of logical reasoning on OWL-DL, heuristic inferences and lexical analysis of the user's query to find related concepts in the ontology and evaluate their relatedness. Also, we present a prototype tool for semantic query expansion and several experimental results of using it on existing ontology.

1 Introduction

Ontologies and other new technologies of the Semantic Web bring new challenges in almost every aspect of the information retrieval process. One of them is querying the ontologies taking advantage of all the new modeling features they have. On the other hand, the concepts and the relations between concepts in the ontology is more likely to be inconsistent with the real world. Typical inconsistencies are synonyms without relation of equivalence or misclassified instances. Things are complicated from the user's view as well. The ontologies are transparent to them, they are not aware of the concepts nor the relations between them. Therefore, querying the ontology should be based on the natural language.

This paper presents an approach to semantic query expansion to solve this problem and a tool which implements it. The main objectives are to propose a method which uses reasoning techniques to evaluate user's query. The method does not depend on the ontology and overcomes various types of possible inconsistencies in the ontology. The required properties of the tool described are proactivity which means the tool tries to take action on behalf of the user whenever possible, and tractability which means the user should see the reason of the tool's behavior and what action should he take to obtain desired result. As the tool itself is part of the larger project, it's integration with other tools from the project is described.

2 Semantic Query Expansion techniques

We give in this section short description of common query expansion techniques and defines several terms used throughout this document [4]. Logic reasoning is the formal semantic reasoning based on the explicitly defined relations between concepts in the ontology. The main logic reasoning expansions are expansion to equivalent concepts, expansion to broader or narrower concepts and expansion the concepts with common superclass.

Heuristic inferences are used to find related concepts which relatedness is not explicitly modeled, e.g., due to inconsistent ontology. We define several types of heuristics. It is obvious that concepts with the same name and namespace are likely to be equivalent regardless of the usage of uppercase and lowercase letters and white characters. If the namespaces are different the relatedness is weaker. Another heuristic type covers the case when a concept is a prefix of the expanded concept or vice-versa. In this case in this case their relatedness is strengthened. Third heuristic type overcomes the missclassification inconsistency. We define that when an instance name contains a name of a class, it belongs to that class with certain probability, e.g., instance `Java_Programmer` belongs to the class `Programmer`. The last heuristic type helps to resolve acronyms. Every acronym can have different meaning in different domain (e.g., *MS* may mean *Mississippi* if it represents a region but also *Microsoft* if it represents an organization).

Lexical query analysis is about finding linguistic relations between concepts. This includes the word normalization (removing morphological endings), stop-words removal and finding lexically related words. The latter step is performed by the cooperating tool named Related Words Manager (RWM).

For making the expansion more precise we established the principle of the query focus. The focus contains the most specific concepts which should be expanded preferably. In order to identify the most specific concepts an ordering of the concepts is defined as:

- instances are more specific than named classes,
- named classes are more specific than predicates.

The user control over the expansion execution is achieved configuration of the parameters before the takes place. The parameters allow to specify whether to the algorithm should find only whole words that match the terms from user's query or not, whether to distinguish between lowercase and uppercase letters, which expansions and which heuristics should be used and whether the query focus should be used.

3 Method description

The method for expansion of the user's query defines number of consequent steps and orders described expansion techniques in one sequence as follows:

1. Ontology selection – a user can specify ontology URL, path of the filesystem or select one of the cached ontologies.

2. Configuration of the parameters to control the expansion execution.
3. Separation of the query to individual terms.
4. Normalization of the terms and stop-words removal.
5. Finding the terms in the concept names and datatype properties.
6. Creation of the initial set of concepts.
7. Query focus identification.
8. Application of the heuristic "Acronym expansion".
9. Expansion to equivalent concepts.
10. Expansion of classes with their instances and subclasses and relatedness calculation until the relatedness falls below certain threshold.
11. Expansion of the instances with their classes and relatedness calculation until the relatedness falls below certain threshold.
12. Expansion of the instances, classes and predicates with their siblings.
13. Expansion of the query focus with lexically related words.
14. Find the expanded terms from previous step in concept names and datatype properties.
15. Final adjustment of the relatedness measure with heuristics.

For the estimation of the expansion accuracy a measure of relatedness is used. If the measure has value of 1 the expanded concept is equivalent with the original. The value of 0 indicates the concepts are not related at all.

4 Prototype tool implementation

The Figure 1 shows architecture of the tool. The user enters the query and parameters through the query interface. Then it is processed by the term separator after which the individual terms could be normalized and query focus is found. The terms are then looked up in the term cache. If the expansions are not found there, the terms are sent to the query expansion and disambiguation engine where the above mentioned method is applied.

The query expansion and disambiguation engine contains three modules which correspond to the three main query expansion techniques. All three modules are implemented in Java programming language. The main module provides logic reasoning and uses Protégé OWL API and Jena2 OWL API with integrated reasoner. The module that handles heuristics contains XML configuration file where the heuristics are defined. The third module represents an interface for communication with external tool Related Words Manager which provides lexical reasoning services.

Actually a prototype tool is implemented to verify the functionality of the main logic reasoning module. The prototype is able to find equivalent instances, classes or predicates to a given concept or set of concepts. It is also capable of finding broader and narrower concepts and concepts with common superclass.

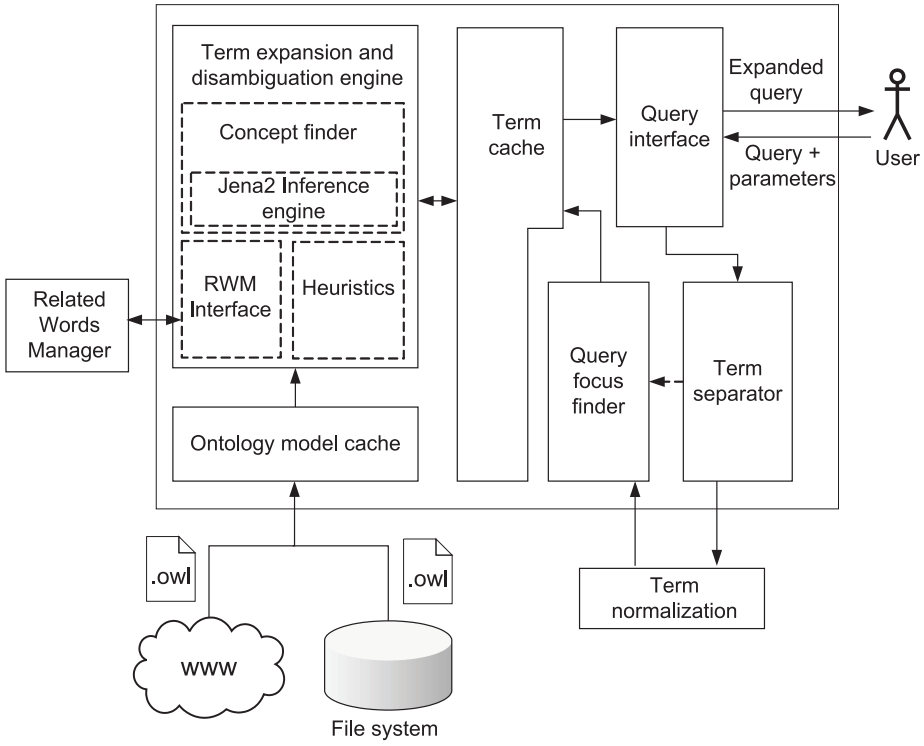

Fig. 1. Architecture of the prototype tool.

5 Experimental results

Various combinations of the basic logic inferences have been experimentally tested. For testing purposes we used test ontology developed within the research project aimed at support of acquisition, organization and presentation of information on the Web for the online labor market domain [1, 2, 3]. The ontology models a domain of job offers and contains many characteristics which can be used to verify our assumptions. For example, it is not fully consistent and contains fairly complex classification tree.

It the following test case we tried to find possible expansions for the class `c:engBiomedicalEngineering` which has a superclass `c:Engineering`. The expansion method was configured to find all sibling classes (classes with common superclass) with their respective equivalent classes. Results are shown in Table 1.

6 Conclusions and future work

The semantic searching in the ontologies can take advantage of various expansion techniques. The three main techniques are logical reasoning, heuristic inferences and lexical query analysis but method which would combine all three approaches

Table 1. Expansion example.

Expanded term	Equivalent
c:engAcoustics_Engineering	-
c:engAeronautical-Aerospace_Engineering	-
c:engChemical_Engineering	c:chChemical_Engineering
c:engCivil-Structural_Engineering	-
c:engElectrical-Electronic_Engineering	-
c:engEnergy_management	-
c:engEnvironmental_Engineering	c:eEnvironmental_Engineering
c:engGeotechnical_Engineering	-
c:engMaterial_Science	-
c:engMechanical_Engineering	-
c:engMetallurgy	-
c:engNanotechnology	c:phNanotechnology, c:bNanotechnology
c:engNuclear_Engineering	-
c:engRegulatory_affairs-IPR	c:bRegulatory_Affairs-IPR, c:chRegulatory_Affairs-IPR
c:engSemiconductor_Engineering	-
c:engSoftware_Engineering	-
c:engTelecommunications_Engineering	-
c:engTransportation_Engineering	-

is not standardized. The method proposed in this paper defines how to combine presented query expansion techniques with several enhancements like query focus. The core idea had been implemented and tested. In the future the rest of the functionality and a simple presentation layer will be implemented. The next development of the tool could also improve the searching with information about user's preferences which are stored in the user ontology.

The idea of query expansion is suitable for rather complex ontologies with many classes, instances and relations between them. Iterative usage of the expansion (expansion of the expansion) causes an interesting effect of gradual ontology discovery. That way, the user can orientate himself and find very quickly what is relevant to him.

Acknowledgements

This work was partially supported by the Slovak State Programme of Research and Development "Establishing of Information Society" under the contract No. 1025/04.

References

1. Návrat, P., Bieliková, M., Rozinajová, V. Methods and Tools for Acquiring and Presenting Information and Knowledge in the Web. In B. Rachev, A. Smrikarov

(Eds.): *CompSysTech 2005*, Varna, Bulgaria, June 2005. pp. IIIB.7.1-IIIB.7.6.
2. Návrat, P., Bieliková, M., Rozinajová, V. Acquiring, Organising and Presenting Information and Knowledge from the Web. In B. Rachev, A. Smrikarov (Eds.): *CompSysTech 2006*, Veliko Turnovo, Bulgaria, June 2006.
3. Návrat, P., Bieliková, M. Tools for Acquiring, Organising and Presenting Knowledge in Heterogeneous Environment of Information Sources. In Paralič, J., Dvorský, J., Krátký, M. (Eds.), *Proc. of 5th Annual Conference Znalosti'06*, 2006, pp. 237–242.
4. Malečka, J. *Document accessing using metadata.* Diploma project. Slovak University of Technology in Bratislava, 2006.
5. Akrivas, G., Wallace, M., Andreou, G., Stamou, G., Kollias, S. Context-Sensitive Semantic Query Expansion. In *Proc. of IEEE Int. Conf. on Artificial Intelligence Systems, ICAIS'02*, 2002, p. 109.
6. Stenmark, D. Query expansion using an intranet-based semantic net. In Proc. of IRIS-26, Porvoo, Finland, August 2003.
7. Navigli, R., Velardi, P. An Analysis of Ontology-based Query Expansion Strategies. In *Workshop on Adaptive Text Extraction and Mining, ATEM 2003*, 14th European Conf. on Machine Learning, ECML'03, Cavtat-Dubrovnik, Croatia, September 2003.