Contextual Ontology Module Learning from Web Snippets and Past User Queries

Nesrine Ben Mustapha, Marie-Aude Aufaure, Hajer Baazaoui, Henda Ben Ghezala

Contextual Ontology Module Learning from Web Snippets and Past User Queries

Nesrine Ben Mustapha[1,2], Marie-Aude Aufaure[1], Hajer Baazaoui Zghal[2], and Henda Ben Ghezala[2]

[1] Ecole Centrale Paris, MAS Laboratory, Business Intelligence Team, France
{nesrine.ben-mustapha,marie-aude.aufaure}@ecp.fr
[2] Laboratory RIADI, ENSI, La Manouba, Tunisia
hajer.baazaouizghal@riadi.rnu.tn}@riadi.rnu.tn

Abstract. In this paper, we focus on modularization aspects for query reformulation in ontology-based question answering on the Web. The main objective is to automatically learn ontology modules that cover search terms of the user. Indeed, the main problem is that current approaches of ontology modularization consider only the input existant ontologies, instead of underlying semantics found in texts. This work proposes an approach of contextual ontology module learning covering particular search terms by analyzing past user queries and snippets provided by search engines. The obtained contextual modules will be used for query reformulation. The proposal has been evaluated on the ground of semantic cotopy measure of discovered ontology modules, relevance of search results.

Key words: Ontology, modular ontology, knowledge, ontology learning

1 Introduction

With the increasing availability of ontologies on the Web, modularity principle has become an important issue to overcome scalability problems over ontology-based systems. Ontology module extraction (OME) approaches consist in reducing ontology to an ontology fragment covering a particular vocabulary. The extracted ontology modules are used for knowledge selection or reuse. Generally, the input of those approaches [1] consists in large ontologies and the output is a set of independent modules. Current approaches of modularization basically rely on static existant ontologies, which can be inconsistent to cover users need. Indeed, users search interest and also domain knowledge evolve with new discoveries and usages. As a consequence of this continuing evolution, automatic methods for ontology construction are required. For the best of our knowledge, machine learning strategies have not yet been explored for ontology modularization as mentioned in [1]. Unlike many previous approaches of modularization, the proposed method has been designed in an automatic and domain-independent way. Web information distributions were employed to assess the reliability of the extracted knowledge. We propose then a new approach of ontology module

extraction from web snippets, and user's context (past user queries and selected documents).

This paper is organized as following. In section 2, an overview of related works on ontology module extraction is presented. Section 3 describes the proposed approach of contextual *OM* extraction. In section 4, we describe an evaluation on the ground of two criteria which are the comparison of discovered ontology modules with an upper-level ontology MESH and the impact of *OM* extraction on the relevance and the ranking quality of search results. Finally, we conclude and discuss directions for future research.

2 Related Works

Ontology module extraction consists in reducing ontology to an ontology fragment that covers a particular vocabulary. In [5], the proposed approach called "ontology segmentation" takes one or several classes of the ontology as an input. It applies a generic algorithm to include all related classes that participate in the definition of the input classes, on the basis of class subsumption and OWL restrictions. Noy and Mussen [3] define a novel traversal view extraction technique for module extraction. Starting from one class of the considered ontology, relations of this class are recursively traversed to include related entities as in [5]. However, this technique is not automatic and takes into account the user involvement in selecting the relations to be traversed and associating to each of them a level of recursion, at which the algorithm should stop "traversing" relations. Besides, the proposed approach in in [4] is composed of: (1) the selection of relevant ontologies, (2) the modularization of the selected ontologies, (3) the merger of the relevant ontology modules in the case when the query terms are covered by several different ontologies. The input of the ontology modularization approach is made up with ontology and a set of terms that should be covered by the smallest part of the ontology. Unlike the algorithm in [5], all the superconcepts of a selected concept are not necessarily included (only the ones that are directly related to concepts of the module, i.e. the most specific common concepts).

In the approaches mentioned above, two main limits are noticed. First, the existing approaches of ontology modularization rely on static ontologies that can be inconsistent to cover basically user's need for information search on the Web. Second, modularization algorithms consider mainly the structure of the input ontology, instead of semantics or context. Consequently, we need semantics-based criteria so as to determine the border of ontology modules. Moreover, the contextuality of the module will considerably depend on the semantic covertness of the original input ontologies. However, ontologies on the web are not sufficiently consistent and contextualized to cover specific domain knowledge. Therefore, our proposed strategy should also differ from previously discussed strategies. Obtained modules have to be based on observing relevant interactions for knowledge selection, not on a human or human-driven dedicated specification, nor on the structural properties of the ontology (as in traversal view strategies). Ontol-

ogy learning (OL) techniques could be a way to overcome these limits. In fact, for the best of our knowledge, unsupervised machine learning strategies have not yet been explored for ontology modularization as mentioned in[1].

Ontology learning (*OL*) aims at building ontologies from knowledge sources using a set of machine learning techniques and knowledge acquisition methods. *OL* from texts has been widely used in the knowledge engineering community. By applying a set of text mining techniques, a granular ontology is enriched with concepts and relationships. in this paper, we focus mainly on two categories of unsupervised techniques that don't need any background knowledge: Lexico-syntactic patterns (*LSP*)and distributional measures. In the last decade, with the enormous growth of Web information, the Web has become an important source of information for knowledge acquisition. Its main advantages are its huge size and its large degree of heterogeneity. *OL* from Web documents requires the same techniques as those used for ontology extraction from texts. A study of several types of available Web search engines and how they can be used to assist the learning process (searching Web resources and computing IR measures) were explored in [6].

The main challenge of the present work is to use *OL* for *OM* construction and its integration in the search process. This work is part of the generic approach. It aims to develop its modular semantic layer from the associations between queries and documents results in order to improve the contextualization of user's goal search and consequently, the answers' relevance of semantic search[7]. In the next section, our proposed approach will be detailed.

3 Contextual Ontology Module Learning Approach

In this section, we describe an approach of *OM* building in an automatic and domain-independent way, using past users queries and resulted snippets (returned by web search engines). Note that the term "snippet" is used here to denote a fragment of a Web page returned by remote search engines (such as GOOGLE or YAHOO) and summarizing the context of searched keywords. Our underlying hypothesis is that an *OM* is an ontology fragment that represents a question on specific domain knowledge. This *OM* can be used to annotate documents related to the specific knowledge component.

The main steps of the proposal are the following: question analysis, candidate answers extraction, context map extraction and contextual module representation using attributed graph. The input of the proposed approach is made up with questions and results pairs (*URls*) related to a specific topic. First, each question is analyzed by identifying the answers' patterns to be used in the next step. Second, these patterns are employed to reformulate queries in order to collect relevant snippets provided by a web search engine. Next, a concept network called context map is extracted from the obtained textual snippets by applying ontology learning techniques (*LSP* and web co-occurrence scores (*WCS*)). A top-level ontology (such as Mesh, Sensus) describing very general concepts that are the same across all knowledge domains can be used to identify question

concepts and import related concepts and relations. The obtained context map acts as a skeleton on which the OM is built. it is represented using attributed graph.

3.1 Question Analysis

This step aims to obtain a reformulated question (RQ), taking into account answers patterns and users context (selected results from users). In order to conceptualize each question, a repository of predefined question patterns (RQP) is designed. Each query type (what, where, who) is associated to a set of answers patterns. According to the question pattern (QP) of the submitted query, answers pattern (AP) are selected and instantiated with question terms to search answers passage from web snippets. For instance, the query (*"what is a BMI?*) is a typical question of the following *QP1* ("$what < be >< name >$"). "$< name >< be >< Answer >$" is an AP assigned to this question pattern. A new RQ is represented by the following phrase: *"BMI is "*. However, query terms can be polysemic. Based on the same premise adopted in corpus-based approaches, we consider that the context can defined by a set of terms which co-occur frequently with query terms in the selected results". Those ones whose frequency is superior to a threshold are selected as belonging to the semantic signature (called also topic signature) of term "t".

In most of corpus-based approaches, the context of a word is usually defined as the word around them within certain of window of which size is usually set as two. Therefore, the analyzed query is extended with two terms that have the highest co-occurrence score (WCS)from the topic signature (TS). This score is described in section 3.2. The contextual query reformulation aims to eliminate the risk of collecting irrelevant snippets to the right sense of terms. For instance, the two high-ranked terms of the TS extracted from user's results related to the question "*What is BMI*" are "*height*" and "*weight*". The new RQ is the following query *"BMI is" AND height AND weight"*.

3.2 Candidate Answers Extraction

To extract candidate Answers (CA) from snippets, the RQ is submitted to the web search engine in order to collect the first β snippets. Using AP, words matching the tag $< answer >$ are selected as CA element. For instance, the following sentence *"BMI is a measure of body fat"* is tagged according to the pattern AP and the result is "BMI $< NAME >$, IS-A $< be >$ the measure of body fat $< answer >$". Then, the following term "measure of body fat" refers to a candidate answer. The corresponding CA values are selected based on WCS 1(superior to a threshold Tsα) (which can include the terms "*measure*", "*calculation*" and "*formula*").

$$score(\text{problem}, \text{choice}) = \frac{hits(problemAndchoice)}{hits(choice)} \quad (1)$$

These *CA* should be ranked and selected according to statistical assessment based on Web-based semantic similarity. Indeed, we used the scores below based on the measure proposed by Turney [2] to evaluate the co-occurrence score (WCS) between initial word problem (terms included in the Reformulated query) and related term candidate choice (candidate answer) by the following formula:

For the rest of this paper, we use the notation *hits(a)* to denote the number of search results that contain the query "a" made to a search engine. The concept candidate whose score is superior to an associated threshold Tsα is selected to be used in the context map construction. The threshold Tsα is based on the average of similarity values between terms of TS (that denotes the topic signature of query terms and obtained in the question analysis step) and the reformulated query RQ. For instance, the corresponding candidate values are selected based on web co-occurrence measures (superior to an threshold Tsα = 0.03.25) as follows: Score $(BMI, measure) = 0.28$; $Score(BMI, calculation) = 0.0373$; $Score(BMI, formula) = 0.37$.

The extracted answers candidates are used to construct new queries in order to collect new collection of snippets. New queries are automatically submitted to the search engine by extending the previous query as the following: BMI is a $< candidate_value >$, in order to collect the first β snippets. Then, a context map can be constructed from this collection and converted to an ontology fragment, as described in the following subsections.

3.3 Context map construction

The aim of this task is to construct context map that represents semantically the possible answers to user's information need, regarding its context. Note that the term "context map" refers to a network of domain terms and relationships extracted from textual passages. This task is mainly based on four operators: Concept identification (CIP), relation operators (RIP), Relation label and Concept learning (RLCP) operator and Concept and relation selection (CRS). The construction of context Map as shown by the figure 1, works as follows.

Concept identification (CIP) and relation operators (RIP). Domain concepts are identified by CIP by using particular typed dependencies which are detected by a syntactic parser. For each type of dependency, a set of transformation is defined, in order to identify domain concepts. Those rules are based on lexico-syntactic patterns. A subset of the following rules is detailed in [11]. The parser provides grammatically typed dependency networks. Then, these networks are mined by the RIP in order to transform automatically the grammatical representation into semantic ones. The semantic representations are then used to create the context map.

Relation label and Concept learning (RLCP). The (RLCP) operator use the constructed context map after the identification of basic domain concept and domain relations in order to discover others possible label of relations and concepts using snippets and lexico-syntatctic patterns. For example, to discover new label of the relation "IS-A" that relates the concept "*BMI*" and "*measure*", the following query: "*BMI* * *measure*" is made to a search engine in order to

import snippets that contain sentences regrouping this two terms in order to extract possible verbs relating them. According to the provided snippets, possible relation labels include the following verbs "*provide*", "*revert to*", "*give*", "*offer*" (figure 1).

On the other hand, new discovered labels are considered as new patterns for candidate concepts that can be related to domain concepts by means of these labels. Therefore, new queries are made to the search engine (such as "*BMI provides **"), which provide relevant sentences containing these patterns. Then, new domain candidates are discovered.

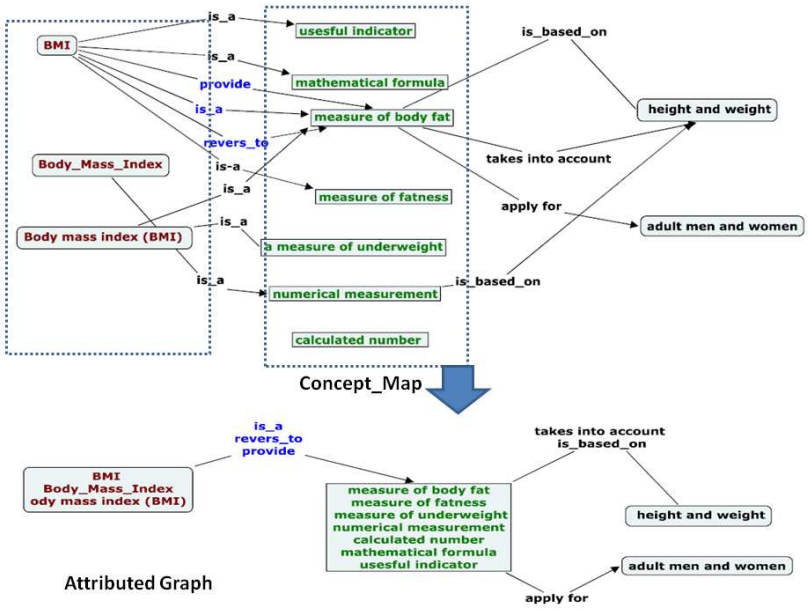

Fig. 1. Example of a context Map and extracted attributed graph related to BMI topic

Concept and relation selection (CRS). Applying the mentioned techniques does not mean that the extracted knowledge is enough strong to grant the definition of the module. This operators need to select the extractions that are sufficiently reliable. To perform this selection, we introduce a Web-based statistical analysis relying on co-occurrence measures computed directly from search engine. Co-occurrence measures are based on distributional hypothesis claiming that words that occur in the same context tend to have similar meanings. Several scores have been proposed in the past to compute Web-scale statistics, adapting the notion of co-occurrence and mutual information (computed as the probability of joint appearance of concepts in a corpus). Discovered candidate

concept is represented by a node and it is weighted using the score presented by the following formula, taking into account the appropriate pattern:

$$\text{Score}_{\text{pattern}},\ \text{(choice)} = \underset{i=1}{\overset{Patterns}{Max}} \left(\frac{hits(\text{"}pattern_i(\text{"}concept\text{"},\text{"}candidate\text{"}\ )}{hits(\text{"}candidate\text{"}\ )}\right)$$

This formula computes the maximum probability of finding any no taxonomic relation involving the candidate concept and the domain concept in the scope of web document containing that candidate. If this score is remarkably low than a threshold, the discovered concept or relation is rejected.

3.4 Ontology Module Representation

This step has as input the context map and as output an attributed graph. Since the OMs are supposed to be extracted from unstructured text, the discovered concepts and relations are not validated at one step. For this reason, we choose to rely on attributed graph because it is a powerfully enough to represent *OM* written in RDF, OWL or DAML+OIL. Besides, attributed graph is the model implemented in the ACG library, for graph transformation. Details about attributed graph are described in [9]. An attributed graph representation of the module $\mathcal{AG}_{\mathcal{M}}$ is a pair $(\mathcal{N}_G, \mathcal{E}_G)$, where $\mathcal{N}_G$ is a set of attributed nodes and $\mathcal{E}_G$ is a set of attributed edges.

An **attributed node** $\mathcal{N}_G = (T_N, AV_N)$ has a type T_N and a set of attribute values AV_N where Tn is the set of terms referring (eg. BMI, Body Mass Index) to a concept C and AV_N is the set of score's (WOC)values assigned to each of the terms belonging to CN.

An **attributed edge** $\mathcal{EG}_M = (T_E, \mathcal{RN}, AV_E, O_E, D_E)$ has a type T_E, a set of attribute values AV_E, an origin node O_E and a destination node D_E, where T_E denotes the type of a relation (hyponymy, meronymy, possession, verb label, etc.) and $\mathcal{RN}$ is a set of terms referring to the relation ($\mathcal{R}$).

An **attribute value** AV_E is a pair $(\mathcal{RN}, score)$ associating score's value to a term of ($\mathcal{RN}$). The figure 1 presents a reused module related to the disease subtopic.

4 Experimentation: Modular ontology learning for semantic search

In order to evaluate the feasibility of the proposed approach, we have tried to compare our approach with different related works on the modularization approaches. However, it has been difficult to compare the proposal with OM extraction approaches since the features and the usage of each input of those approaches are different. Therefore, it seems more logical to evaluate the present work according to OL creteria. We add that the evaluation of the proposed approach is based on two main criteria which are: (1) the comparison of OM learning process (figure 2) and (2) the impact of *OM* learning on the relevance of search results (figure 3).

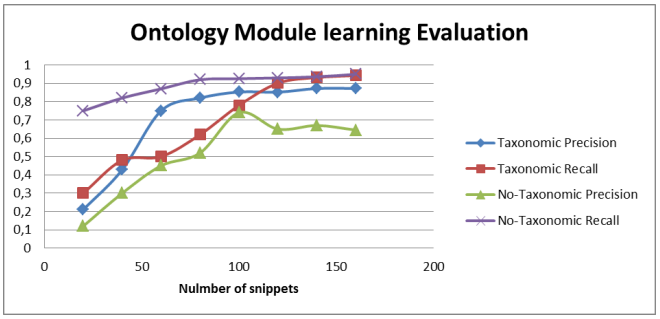

Fig. 2. ontology module learning evaluation

In figure 2, two ontologies are compared. One one hand, we use a Taxonomic Precision (TP) which is a similarity measure based on the notion of semantic cotopy sc. It is recently presented and analysed in [10]. The reason to choose this measure was to take advantage of its ability to compare ontologies as whole structures. The values of TP are from the range [0, 1]. We use the Mesh Ontology (MSO) as an ontology reference. 80 queries on the topic of animal diseases were collected manually by using 80 concepts of MSO. 80 ontology modules which make up a large ontology (RO) were constructed according to the proposed approach to be compared with (MSO).Semantic Cotopy sc(c,O) of a concept c from ontology O is a set containing c and all super and sub-concepts of c in O, excluding the concept root of (O). Then, TP(c, RO, MSO) of concept c and two ontologies RO and MSO where $c \in RO$ and $c \in MSO$ is defined as follows:

$$TP(\mathrm{C}, \mathrm{RO}, \mathrm{MSO}) = \frac{sc(\mathrm{C}, \mathrm{MSO}) \bigcap sc(\mathrm{C}, \mathrm{RO})}{sc(\mathrm{C}, \mathrm{RO})} \tag{2}$$

A Taxonomic Recall (TR) can be assessed as follows:

$$TP(\mathrm{C}, \mathrm{RO}, \mathrm{MSO}) = \frac{sc(\mathrm{C}, \mathrm{MSO}) \bigcap sc(\mathrm{C}, \mathrm{RO})}{sc(\mathrm{C}, \mathrm{MSO})} \tag{3}$$

Therefore, the global TP and TR are computed respectively by the following formulas:

$$GTP(\mathrm{RO}, \mathrm{MSO}) = \frac{1}{|\mathrm{RO}|} \sum_{\mathrm{c} \in \mathrm{RO}} TP(\mathrm{c}, \mathrm{RO}, \mathrm{MSO}) \tag{4}$$

$$GTR(\mathrm{RO}, \mathrm{MSO}) = \frac{1}{|\mathrm{MSO}|} \times \sum_{\mathrm{c} \in \mathrm{RO}} TP(\mathrm{c}, \mathrm{RO}, \mathrm{MSO}) \tag{5}$$

No taxonomic precision and recall are calculated according to the same formula by substituting the sc(c, O) by the set containing concept c and all concepts related to c by a no taxonomic relationship. The figure 2 shows the evolution of the precision of taxonomic and non taxonomic structure according to the number of snippets used in the ontology module leaning. On the other hand, in

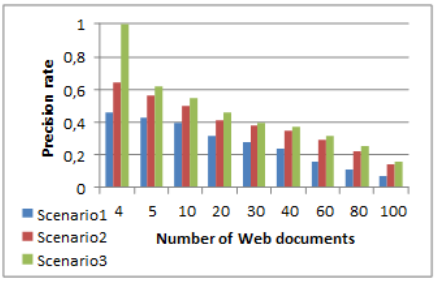

Fig. 3. Evaluation of result precision

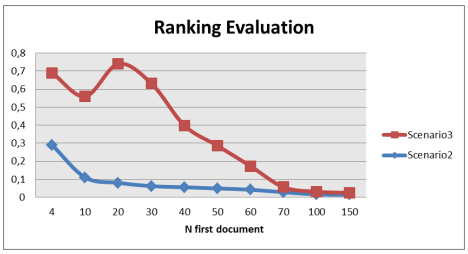

Fig. 4. Evaluation of results ranking

order to evaluate the approach presented in this paper, the impact of the use of OMs during query reformulation is also experimented. First, we have computed the precision of results retrieved by means of query reformulation using discovered modules. Evaluation results contained in Figure 3 represent the obtained precision according to the number of retrieved documents (from 4 to 100). The first scenario represents the initial search, which is a keyword search on Yahoo. The second scenario represents the situation where there are similar cases in the database. The search is based on using WorldNet to add synonyms. The third scenario represents the search for information based on the learned OMs by using 100 snippets for ontology module learning. The query reformulation is based on answers pattern extracted from constructed OMs. We have observe a significant improvement of the relevance of the retrieved information according to the amount of knowledge considered during query reformulation and *OM* creation. We have also noticed that this improvement is maintained as the number of documents increases, even though the quality of the retrieved document set decreases due to the higher amount of noisy and non-related documents retrieved. The results have revealed that: (1) Their accuracy was significantly improved by using modular ontologies; (2) Strongly, discovered ontology module are important to better contextualize users searches and (3) The relevance of documents are not based on the terms frequency but on the semantic relatedness between terms.

Second, in order to evaluate the ranking quality of results according to the formulated query, we used the well-known Normalized Discounted Cumulative Gain measure. While evaluating a ranking list, NDCG is computed according to the original paper [8], as follows:$NDGC(n) = Z_n \sum_{j=1}^{n} \frac{2^{r(j)}-1}{log(j+1)}$, Where $r(j)$ is the rating of the j-th document in the list, and the normalization constant Z_n is chosen so that the perfect list gets a NDCG score of 1. Figure 4 shows the evaluation results measured by the NDCG for the two scenarios previously described. The X-axis refers to the Web page rank. Again, it is shown that reformulated queries using pattern answers (extracted from obtained ontology modules) have contributed to improve significantly the document raking.

5 Conclusion and Future Work

In this paper, we have proposed a new approach of Ontology Module extraction from web snippets, and user's feedback (past user queries and selected documents). Unlike many previous modularization approaches, the originality of this work is that it has been designed in an automatic and domain-independent way, exploiting unsupervised techniques and the web as a large scale learning source. The contribution resides in the following techniques: Web-based co-occurrence measures for the assessment of extracted knowledge (concepts and relationships) and Unsupervised method for context map construction and Attributed graph representation for a multi-label representation of ontology module. The evaluation of the proposal is based on two criteria which are the comparison of OM extraction process and the impact of module-based query reformulation on the relevance of search results. The evaluation of question answering system has revealed that the accuracy of the results was significantly improved by using modular ontologies. Our ongoing work aims at exploring ontology module construction for social search systems.

References

1. Stuckenschmidt, H., Parent, C., Spaccapietra, S.: Modular Ontologies: Concepts, Theories and Techniques for Knowledge Modularization In, Springer-Verlag, Berlin, Heidelberg, (2009).
2. Turney, P.: Mining the web for synonyms: PMI-I versus LSA on TOA In ECML, (2001) 491-502.
3. Noy, N., Musen, M.: Specifying Ontology Views by Traversal, In Proc. of International Semantic Web Conference (ISWC), (2004).
4. D'Aquin, M., Sabou, M., Motta,E.: Modularization, a key for the Dynamic Selection of Relevant Knowledge Components. Proc. of the ISWC 2006 Workshop on Modular Ontologies, (2006).
5. Seidenberg J., Rector, A.: Web Ontology Segmentation: Analysis, Classification and Use In: Proc. of the World Wide Web Conference (WWW),(2006).
6. Sanchez, D., and Moreno, A.: Learning non-taxonomic relationships from web documents for domain ontology construction. DKE, 64(3):600-623, (2008).
7. Elloumi, M.,Ben-Mustapha, N., Baazaoui, H., Moreno, A., Sanchez, D.: Evolutive Content-Based Search System In KDIR, (2010).
8. Jarvelin, K., Kekalainen, J.: Cumulated gain-based evaluation of ir techniques. ACM Transactions on Information Systems, 20(4), (2002) 422-446.
9. Ehrig, H., Ehrig, K., Prange, U., Taentzer, G.: Fundamental theory for typed attributed graphs and graph transformation based on adhesive hlr categories. Fundam. Inf., 74:31:61, (2006)
10. Maedche, A., Staab, S.: Measuring similarity between ontologies. Proc. CIKM 2002. LNAI vol. 2473, (2002).
11. Schutz, A., Buitelaar, P.: Relext: A tool for relation extraction from text in ontology extension. In Y. Gil, E. Motta, V. R. Benjamins, and M. A. Musen, editors, ISWC, volume 3729 of LNCS, Springer, (2005) 593-606.