Mapping Ontologies Elements using Features in a Latent Space

Vassilis Spiliopoulos[1, 2], George A. Vouros[1], Vangelis Karkaletsis[2]

[1]*Dpt. of Information and Communication Systems Engineering,University of the Aegean, Greece*
{vspiliop | georgev}@aegean.gr
[2]*Inst. of Informatics and Telecommunications, NCSR "Demokritos", Greece*
vangelis@iit.demokritos.gr

Abstract

This paper proposes a method for the mapping of ontologies that, in a greater extent than other approaches, discovers and exploits sets of latent features for approximating the intended meaning of ontology elements. This is done by applying the reverse generative process of the Latent Dirichlet Allocation model. Similarity between element pairs is computed by means of the Kullback-Leibler divergence measure. Experimental results show the potential of the method.

1. Introduction

Existing approaches towards automating the ontology mapping task exploit various types of features (i.e., linguistic and/or structural/contextual) extracted from the vicinity of ontology elements (concepts and properties). State of the art approaches [1] compute mapping pairs by exploiting combinations of ontology features' types.

In contrast to these approaches and in a greater extent than similar approaches that necessitate the use of external resources, this paper proposes a method for computing mappings between ontology elements by exploiting *latent features* that reveal the intended meaning of ontology elements. This is done by applying the reverse generative process of the Latent Dirichlet Allocation (LDA) [2] model. Doing so, each ontology element is represented as a distribution over latent features, and similarities between elements' pairs are computed by means of the Kullback-Leibler divergence [3] measure.

The major advantages of this approach are as follows: The use of latent features helps to deal with problems of imprecise and vague ontology elements' descriptions, as well as with cases of polysemy and synonymy of words. The proposed approach does not presuppose the existence of any external resource, as it uses words appearing in the vicinity of ontology elements.

The rest of the paper is structured as follows: Section 2 states the problem, describes related approaches and motivates the proposed approach. Section 3 provides background information. Section 4 thoroughly describes the proposed method as a process of discrete steps, and section 5 presents evaluation results. Section 6 concludes the paper.

2. Problem Statement and Related Work

2.1. Problem Statement

An ontology is a pair *O=(S,A)*, where *S* is the ontological signature describing the vocabulary and *A* is a set of ontological axioms, restricting the intended interpretations of the words in the signature.

A mapping from an ontology $O_1 = (S_1, A_1)$ to an ontology $O_2 = (S_2, A_2)$ is a morphism $f:S_1 \rightarrow S_2$ of ontological signatures such that $A_2 \models f(A_1)$, i.e., all interpretations that satisfy O_2's axioms also satisfy O_1's translated axioms.

Given the above statement, the exact problem that this paper aims to solve is as follows:

Given two ontologies, $O_1 = (S_1, A_1)$ and $O_2 = (S_2, A_2)$, (a) find a latent space of features in terms of which the intended meaning of ontology elements can be expressed, (b) express the intended meaning of ontology elements using these features, and (c) compute a morphism $f:S_1 \rightarrow S_2$ of ontological signatures, by measuring the proximity of elements' intended meaning, such that $A_2 \models f(A_1)$.

2.2. Related Work

The approaches that are more closely related to the above stated problem are the Virtual Documents [4], the ASCO [5] and the HCONE [6] approaches.

Both ASCO and Virtual Documents approaches rely to the representation of ontology elements (i.e., concepts and properties) as vectors of weights. Each weight corresponds to a word and is being calculated using the TF/IDF measure. The similarity between two vectors is being computed by means of the cosine similarity measure. Given that the features in terms of which ontology elements are being represented are words in the vicinity of each element, these methods are prone to imprecise mappings due to words' polysemy and synonymy, to redundant repetitions of words, and to vague or imprecise descriptions of the ontology elements.

Instead of computing similarities between ontology elements in a direct way, the HCONE approach [6] maps ontology elements to WordNet synsets by applying the Latent Semantic Indexing (LSI) [6] method. The major drawback of this approach is its dependency from WordNet, leading to low recall values. Moreover, this approach, although highly precise, is prone to, what has been called, "semantic noise" [6]: This is due to the appearance of words that may distract the mapping process.

In this paper, to a greater extend than the above mentioned approaches, we investigate the problem of uncovering and exploiting *latent features* for representing ontology elements' intended meaning, aiming to compute ontology elements' mappings.

3. Background Knowledge

This section briefly provides basic information behind the use of Probabilistic Topic Models [2, 7] for uncovering latent features, as well as about the use of Kullback-Leibler information theoretic similarity measure [3].

Probabilistic Topic Models. A probabilistic topic model [7] specifies a certain generative process: Documents (assumed to be "bag of words") can be generated by a different mixture of latent variables (i.e., probability distributions over words). An instance of the generative process consists of: (a) latent variables specifying the probability of each word to appear in each latent variable, (b) the probability according to which each variable contributes to the generation of each document and (c) a number of different documents that are generated, each emphasizing on thematic topics, whose mixture is represented by a specific combination of the latent variables. It must be pointed out that the relation between words and latent variables is many-to-many. Furthermore, by emphasizing on latent variables rather than words, probabilistic topic models aim to capture the "significant" features (i.e., latent features) in terms of which different documents can be represented.

While the above concern the generation of documents by known mixtures of known latent variables, we are interested in the reverse process: Documents that express the meaning of ontology elements are known, and we need to infer the latent variables along with their mixture proportions for each document. To do this we use the Latent Dirichlet Allocation (LDA) model [2] and we utilize a specific form of a Markov Chain Monte Carlo (MCMC) process called Gibbs sampling[1] for reversing this process. The reader is referred to [7] for a detailed explanation.

Similarity Measure. To assess the similarity of ontology elements' meaning representations (expressed as multinomial distributions over the inferred latent features) we use the Kullback-Leibler (KL) [3] divergence measure: If p and q are two probability distributions (in our case p and q are the representations of ontology elements given T latent features), then the asymmetric KL-divergence between them is:

$$I(p,q) = \sum_{i=1}^{T} p_i \log_2\left(\frac{p_i}{q_i}\right)$$

The symmetric variant is defined as follows:

$$KL(p,q) = I(p,q) + I(q,p)$$

The highest similarity is represented by a KL-divergence equal to zero.

4. The Proposed Method

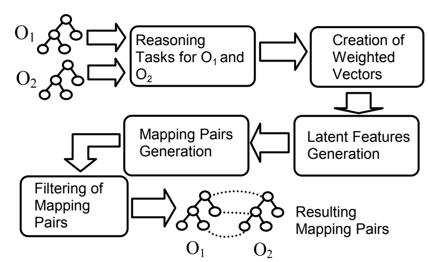

Figure 1. The mapping process

As figure 1 depicts, the input of the process consists of two ontologies, O_1 and O_2, formatted in any language of the OWL family of languages. The output consists of the mapped element pairs (concepts and properties) of the input ontologies. The paragraphs that follow explain each step of the method.

Reasoning Tasks: The method applies reasoning tasks for deriving subsumption, equivalence and disjoint relations that are not explicitly specified in the input ontologies.

[1] The LDA model implementation with Gibbs sampling that we have used is in http://www.arbylon.net/projects.

Creation of Weighted Vectors: Any element of O_1 or O_2 is represented as a vector $(f_1, f_2, \ldots, f_N)$, where f_i, $i=1, \ldots, N$ are the frequencies of the distinct N words extracted from both O_1 and O_2. To construct such a representation, the method extracts words occurring in the local name, label and comments of the ontology element and from all of its related elements. Finally, words from all the instances of concepts and properties are being extracted.

To further enrich the representation of an element, the following rules apply:

- The method extracts words from all the elements that are stated to be equivalent to this element.
- If the corresponding element is defined as the disjunction (conjunction) of other elements, then the method unions (respectively, intersects) the sets of words that have been extracted from all elements in the disjunction (respectively conjunction).
- If two elements are defined to be disjoint, then the method removes their common words from their "bags of words".
- The method extracts words from all direct super and sub-elements.

Optionally, if the local name or the label of an element has an entry in WordNet, then the words from all available synsets can also be extracted. The use of WordNet is not an essential part of the process.

During this step, tokenization, stemming and elimination of stop words is performed on the set of extracted words.

Latent Features Generation: In this step, the ontology elements' vectors are transformed to multinomial distributions over latent features. This is done by applying the reverse generative process of the LDA model. Concerning any element of O_1 or O_2, the resulting multinomial distribution over a given number of T latent features will be of the form $(lt_1, lt_2, \ldots, lt_T)$, where lt_i, $i=1,\ldots,T$ corresponds to the i-th latent feature and specifies the "contribution" of the corresponding feature in approximating the intended meaning of the ontology element.

Mapping Pairs Generation: Given two ontology elements q, p (represented as multinomial distributions over latent features) the similarity between these elements is defined as follows:

$$Sim(p,q) = 1 - \frac{KL(p,q)}{\underset{i,j}{Max}\{KL(i,j)\}}, \forall\, i \in O_1, j \in O_2$$

Sim is defined in [0,1], where "1" represents the best similarity and "0" the worst.

Two elements are considered to be equivalent if their *Sim* value is higher than a threshold value t. Through experimentation we specify the threshold t to be 0.75. It must be pointed out that although the method may locate mappings of 1:1 or 1:n cardinality, it is currently restricted to 1:1 mappings, by returning only the pairs with the highest *Sim* value.

Filtering of Mapping Pairs: All mapping pairs that introduce inconsistencies are filtered out. This assures that all interpretations that satisfy O_2's inclusion axioms satisfy O_1's translated inclusion axioms, given the mapping pairs computed in the previous step.

5. Experimental Results

We used the OAEI 2006 [8] dataset which comprises 51 distinct test cases (i.e., ontology pairs (O_1,O_2)) clustered in 5 distinct categories.

In our experiments, for each pair of ontologies we provide results in cases that Wordnet is being consulted (+WordNet cases) and in cases that it does not (-WordNet cases). Furthermore, the proposed method is being compared with the HCONE method and a term-based method (TERMS), where ontology elements are represented as vectors of words' frequencies. In the latter case, the similarity between two elements is being computed by means of vectors' Euclidean distance.

5.1. Results and Discussion

The results show the precision and recall of the proposed method for each pair of ontologies in the dataset. Precision measures the ratio (*#correct_pairs_computed/#pairs_computed*) and recall the ratio (*#correct_pairs_computed/ #correct_pairs*). We also present the weighted harmonic mean (H-mean) measure, where weights are the *#correct_pairs_computed*, following the OAEI competition specifications.

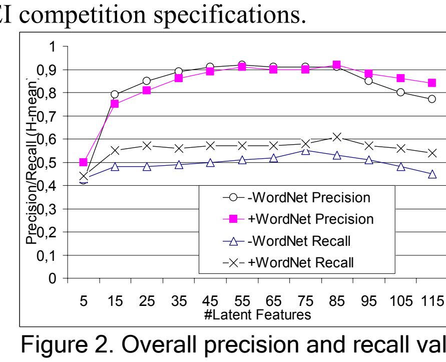

Figure 2. Overall precision and recall values

Overall performance. Figure 2 shows the H-mean for all test categories, for various numbers of latent features (x-axis).

As it can be pointed, the usage of WordNet leads to nearly the same overall precision (1% improvement) and to a small improvement in terms of recall (6%): This is very encouraging, showing the tolerance of this method in "semantic noise" (since all words from all WordNet senses are being used). Furthermore, this behavior shows that the existence of external resources can be of benefit to the proposed method, without being necessary.

Number of Latent Features. The number of latent features generated by the LDA model is a very important factor for the effectiveness of the method: This is clearly depicted in figure 2: When the number of latent features is getting smaller (respectively, larger), then they are too broad (respectively, narrow) in meaning, affecting the precision and recall of the proposed method. In our experiments we have specified *T*=85 for +WordNet cases and *T*=75 for –WordNet cases. Thorough study of the number of latent features is part of future work.

Comparison. As Figure 3 shows, in comparison to the HCONE method the proposed method achieves a "better balance" between precision and recall (this is also true in category 301-304, concerning mappings against real-world ontologies): Due to its dependency on the existence of the appropriate WordNet entries, the HCONE method achieves very high precision but its recall value is quite low.

It must be pointed that in cases (category 248-266) where ontology elements are not related with many words HCONE performs much more poorly than the proposed method.

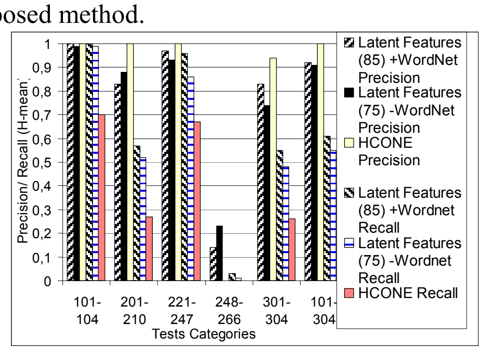

Figure 3. Precision and recall values in categories

Figure 4 depicts the precision and recall graphs given the average values of precision and recall for all categories. We observe that the +WordNet case has slightly better performance than the –WordNet case, followed by TERMS. The performance of TERMS is due to the phenomena of synonymy and polysemy of words. Generally, synonymy tends to decrease recall by introducing false extra mappings and polysemy tends to decrease precision as the assessed meaning of the elements is ambiguous. As the proposed method deals with these phenomena, it outperforms TERMS. HCONE performs poorly than the other methods, due to its dependency on WordNet.

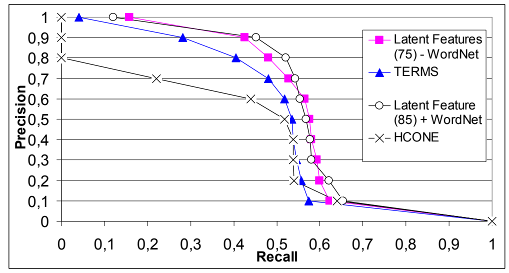

Figure 4. Precision and recall graphs

6. Conclusions

We proposed a method for the mapping of ontologies that emphasizes on the discovery and exploitation of latent features, whose combinations approximate the intended meaning of ontology elements. The overall experimental results are very encouraging, given that the method achieves high precision in quite high recall, it is tolerant to polysemy and synonymy phenomena, as well as to "semantic noise", and although it can benefit from external resources it can be applied without such a consultation.

Acknowledgements. This work was supported by the ONTOSUM project (www.ontosum.org).

7. References

[1] P. Shvaiko, J. Euzenat, "A Survey of Schema-based Matching Approaches", J. Data Semantics IV, 2005, pp. 146-171.

[2] Blei, D. M., Ng, A. Y., Jordan, M. I., "Latent Dirichlet Allocation", J. Machine Learning Research, 2003, 3, pp. 993-1022.

[3] Lin J., "Divergence measures based on Shannon entropy", IEEE Trans on Information Theory, 1991, 37(14), pp. 145-51.

[4] Y. Qu, W. Hu, G. Cheng, "Constructing Virtual Documents for Ontology Matching", In Proceedings of WWW, 2006.

[5] Le B.T., Dieng-Kuntz R., Gandon F., "On Ontology Matching Problems - for Building a Corporate Semantic Web in a Multi-Communities Organization". ICEIS, 4, 2004, pp. 236-243.

[6] Kotis, G. Vouros, K. Stergiou, "Towards Automatic Merging of Domain Ontologies: The HCONE-merge approach", J. Web Semantics (JWS), 2006, pp. 60-79.

[7] Griffiths, M. Steyvers, "A probabilistic approach to semantic representation", In Proceedings of CogSci, 2002.

[8] OAEI 2006, http://oaei.ontologymatching.org/2006.