Learning Subsumption Relations with *CSR*: A Classification-based Method for the Alignment of Ontologies[1]

Vassilis Spiliopoulos[1, 2], Alexandros G. Valarakos[1], George A. Vouros[1], and Vangelis Karkaletsis[2]

[1]AI Lab, Information and Communication Systems Engineering Department, University of the Aegean, Samos, 83 200, Greece
{vspiliop, alexv, georgev}@aegean.gr
[2]Institution of Informatics and Telecommunications, NCSR "Demokritos", Greece
vangelis@iit.demokritos.gr

Abstract. In this paper we propose the "Classification-Based Learning of Subsumption Relations for the Alignment of Ontologies" (*CSR*) method. Given a pair of concepts from two ontologies, the objective of *CSR* is to identify patterns of concepts' features (here, properties) that provide evidence for the subsumption relation among these concepts. This is achieved by means of a classification task using decision trees. For the learning of the decision trees, the proposed method generates training datasets from the source ontologies', considering each ontology in isolation. The paper describes thoroughly the method, provides experimental results for computing subsumption relations over an extended version of the OAEI 2006 benchmarking series and discusses the potential of the method.

Keywords: ontology alignment, subsumption, supervised learning, binary classification.

1 Introduction

Although many efforts [1] aim to the automatic discovery of equivalence relations between the elements of ontologies, in this paper we conjecture that this is not enough: To deal effectively with the ontologies' alignment problem, we have to deal with the discovery of subsumption relations among ontology elements. This is particularly true, when we deal with ontologies whose conceptualizations are at different "granularity levels": In these cases, elements (concepts and/or properties) of an ontology are more generic than the corresponding elements of another ontology. Although subsumption relations between the elements of two ontologies may be deduced by the equivalence relations of other elements, in extreme cases where no equivalence relations exist, this can not be done. In any case, we conjecture that the discovery of subsumption relations between elements of different ontologies may further facilitate the discovery/filtering of equivalence relations, and vise-versa, augmenting the effectiveness of our ontology alignment and merging methods [2].

This paper presents the "Classification-Based Learning of Subsumption Relations for the Alignment of Ontologies" (*CSR*) method. *CSR* computes subsumption relations between concept pairs of two distinct ontologies by means of a classification task, using decision trees, and by exploiting equivalences between properties. Given a pair of concepts, the supervised machine learning method "locates" a hypothesis concerning their relation in a space of hypotheses, which best fits (but not restricted) to the training examples [3], generalizing beyond them. Concept pairs are represented as feature vectors of length equal to the number of the *distinct* properties of source and target ontologies: Equivalent properties (i.e., properties with equivalent meaning) correspond to the same vector component. The training examples for the learning method are being generated from the target and source ontologies.

Although other features may be used, in this paper we study the importance of concepts' properties to assessing the subsumption between concepts: This is an important first step to assessing subsumption relations among concepts, since (a) it appeals to our intuition about the importance of properties as distinguishing characteristics of classes of entities, (b) it makes the least possible commitment to the precision of any method for the discovery of equivalence relations among ontology elements, (c) it provides a basic method that can be further enhanced with other concepts' distinguishing features (e.g., concepts in a given vicinity), and can be further combined with other

[1] This work is part of research project ONTOSUM (www.ontosum.org), implemented within the framework of the "Reinforcement Programme of Human Research Manpower" (PENED) and co-financed by E.U.-European Social Fund (75%) and the Greek Ministry of Development-GSRT (25%).

alignment methods: This paper studies the potential of *CSR*, while leaving further enhancements and combinations for future work.

The machine learning approach has been chosen since (a) there are no evident generic rules *directly* capturing the existence of a subsumption relation between ontology elements (e.g., by means of their surface appearance) and (b) concept pairs of the same ontology provide examples for the subsumption relation, making the method self-adapting to idiosyncrasies of specific domains, and non-dependant to external resources.

Decision trees are used widely in classification problems, since they are robust to noisy data, to missing attribute values, and they are capable of learning disjunctive expressions [3]: Features that match to the subsumption computation problem. Weka's j48 [3] is the implementation of the widely used and state of the art C4.5 [4] decision tree learning algorithm that we have used in this work. C4.5 suits perfectly to problems with the above characteristics.

2 Problem Statement and Related Work

An ontology is a pair $O=(S, A)$, where S is the ontological signature describing the vocabulary (i.e., the terms that lexicalize ontology elements) and A is a set of ontological axioms, restricting the intended meaning of the terms included in the signature [5]. Considering a partition of S let us introduce the sets S_p and S_c, denoting the sets of terms lexicalizing ontology properties and ontology concepts, respectively.

Ontology mapping from a source ontology $O_1=(S_1,A_1)$ to a target 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. However, instead of a function, we may articulate five different kinds of binary relations between the elements of source ontologies: Namely, equivalence ($\equiv$), subsumption (inclusion) ($\sqsupseteq$ or $\sqsubseteq$), mismatch ($\perp$) and overlapping ($\sqcap$). In this case, the ontology mapping problem is as follows: Classify any pair (C^1,C^2) of elements of the input ontologies, such that C^i is a term in S_i, $i=1,2$, to the above relations, consistently.

In this paper we deal with the *subsumption computation problem* which, given the above generic problem, is as follows: Given (a) a source ontology $O_1=(S_1, A_1)$ and a target ontology $O_2=(S_2, A_2)$ such that $S_1=S_{1c} \cup S_{1p}$ and $S_2=S_{2c} \cup S_{2p}$, and (b) a morphism $f:S_{1p} \rightarrow S_{2p}$ from the lexicalizations of the properties of the source ontology to the lexicalizations of the properties of the target ontology (computing properties' equivalences), classify each pair (C^1,C^2) of concepts, where C^1 is a term in S_{1c} and C^2 is a term in S_{2c}, to two distinct classes: To the "subsumption" ($\sqsubseteq$) class, or to the class "*R*". The class "*R"* denotes pairs of concepts that are not known to be related via the subsumption[2] relation, or that are known to be related via the equivalence, mismatch or overlapping relations.

Given the above stated problem, to the best of our knowledge only the *Semantic Matching* approach [6] deals with the computing of subsumption relations between concepts of ontologies. This method relies on codified knowledge contained in external dictionaries, and specifically in WordNet, transforming the available information into a propositional formula and solving a propositional satisfiability problem. Relations that do not satisfy the formula are filtered out and the remaining ones are returned in order of semantic strength.

In contrast to *Semantic Matching* method, *CSR* is a machine-learning based method that exploits the semantics of the input ontologies to assess the equivalence of properties and to generate the appropriate examples for the training of the classifier. This makes the proposed method independent from any third/external domain resource (lexicon or thesaurus).

3 The Classification-Based Learning of Subsumption Relations (*CSR*) Method

As it is shown in Fig 1, given a pair of ontologies $O_1=(S_1, A_1)$ and $O_2=(S_2, A_2)$, expressed in OWL-DL, the aim of the *CSR* method is to classify pairs of concepts (C^1,C^2), where C^1 is a term in S_{1C} and C^2 is a term in S_{2C}, either in the class "$\sqsubseteq$" – assessing the fact that the concept C^1 is subsumed by C^2 – or in the class "*R*".

[2] This means that a pair of concepts belonging to "*R*" may belong to the subsumption relation. In conjunction, "*R*" includes concept pairs that are not related via the subsumption relation (e.g., disjoint concepts).

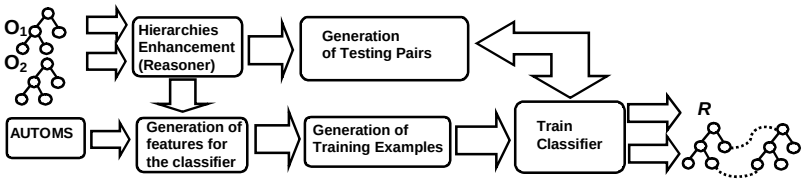

Fig. 1. Overview of the *CSR* method.

The discrete steps of the *CSR* method, as depicted in Fig. 1, are the following:

- Reasoning services are being used for inferring all facts according to ontologies' specification semantics [7]. This is a necessary step as it affects the generation of the training dataset.
- The generation of the features is performed by gathering all discrete properties from both ontologies:

Given properties' equivalencies computed by the AUTOMS mapping tool [8], each pair of concepts (C^1,C^2) is represented by a vector whose components range in {0, 1, 2, 3}. Specifically, the value of a feature is (a) "0", if the corresponding property does not appear neither in C^1 nor C^2, (b) "1", if the corresponding property appears only in C^1, (c) "2", if the corresponding property appears only in C^2, (c) "3", if the corresponding property appears in both C^1 and C^2.

It must be emphasized that (a) by "property appearance" we do not mean the occurrence of the property's lexicalization, but the occurrence of property's meaning as it is assessed by AUTOMS, and (b) feature vectors are not identical for symmetric pairs of concepts, allowing the computation of the direction of the subsumption relation.

- The sets of training examples are being generated according to the generic rules specified in the following. This step includes the balancing of the training dataset, as well.

Generating the training examples for the class "$\sqsubseteq$". This set of examples contains all the stated and inferred subsumption relations among the concepts in each of the source ontologies.

Generating the training examples for the class "***R***". According to the open world semantics, in case there is not an axiom that specifies the subsumption relation between a pair of concepts (or in case this relation can not be inferred by exploiting the semantics of the subsumption relation), then this pair does not belong to the subsumption class and it is included in the generic class "*R*". Training examples for the class "*R*" is further enriched by taking into account (a) the stated equivalence relations between concepts, and (b) by exploiting the union construct: Concepts defined as the union of others, can be substituted by any of their constituents.

The result of the above rule is the definition of four different training example categories for class "*R*", defined as follows: (a) *Concepts belonging to different hierarchies*, (b) *siblings at the same hierarchy level* for which no subsumption relation is defined or inferred, (c) *siblings at different hierarchy levels*: If any concept that is in a pair belonging in the "siblings of the same hierarchy level" category is substituted by any of its subsumees, then new pair examples are recursively generated, until the leaf concepts of the ontology are reached. Finally, (d) *concepts related via an object property* in case no subsumption relation is defined or inferred between them.

Creating a Balanced Training Dataset. It is very important for the performance of the classifier that the training examples are balanced: The number of training examples of the two classes must be equal, forcing both categories to be equally represented. This is referred as the *dataset imbalance* problem. Considering the various techniques the have been proposed towards its solution [9], we have adopted an under-sampling method:

1. All the generated training examples for the class "$\sqsubseteq$" are used.
2. Duplicate examples across different categories of class "*R*" are removed.
3. Select randomly *n/t* examples for each category of training examples of the class "*R*", where *n* is the number of examples of class "$\sqsubseteq$" and *t* is the number of different categories of class "*R*".

Given that examples are chosen randomly, the under-sampling method introduces non-determinism into the learning process. Furthermore, as shown in step 3 above, all the different types of example categories are equally present among the example pairs for class "*R*". This is of paramount importance, as a "good" classifier must learn to identify all the different types of examples.

Subsequent steps are as follows:

- The classifier is being trained using the training dataset.
- Concept pairs are being classified by the trained classifier, pruning the search space.

In order to prune the search space, the proposed algorithm firstly checks all the concepts from the first ontology with the root concepts (concepts with no subsumer) and unit concepts (root concepts

with no subsumees) of the second ontology. If a pair is not classified in the class "⊑", then the hierarchy rooted by the corresponding concept of the second ontology is not being examined by the classifier. If a pair is assessed to belong to the class "⊑", then the concept of the first ontology is recursively being tested with the direct subsumees of the corresponding concept in the second ontology, until either a pair is assessed to belong in the class "*R*", or until the leaf concepts are reached.

4 Experimental Results and Discussion

The testing dataset has been derived from the benchmarking series of the OAEI 2006 contest [10]. As our method exploits the properties of concepts, we do not include ontologies with no properties. Hence, the compiled corpus consists of 31 out of the 51 OAEI 2006 ontologies, and it is available at the URL http://www.icsd.aegean.gr/incosys/csr, together with the gold standard created. All benchmarks (101-304) except *R1-R4*, define the second ontology of each pair as an alteration of the same first. The benchmarks can be categorized based on their common features as follows: (a) in *A1-A5* (101-210, 237, 238 and 249) elements' lexicalizations of the target ontologies are altered in various ways (e.g., uppercasing, underscore, foreign language, synonyms and random strings), (b) in *A6-A7* (225 and 230) restrictions are removed and/or properties are modeled in more detail, (c) in *F1-F2* (222, 237, 251 and 258) the hierarchies are flattened and/or random lexicalizations of elements are introduced, (d) in *E1-E2* (223, 238, 252 and 259) the same as *F1-F2*, but the hierarchies are expanded and (e) in *R1-R4* (301-304) target ontologies are real world ontologies.

Due to the non-determinism introduced by the under-sampling method used, for each ontology pair the experimental results have been produced by applying the *CSR* method 20 times. The set of evaluation values produced during the experiments are visualized by using boxplots [11].

Results show the precision and recall of the proposed method as it is applied in the different types of ontology pairs. Precision is the ratio *#correct_pairs_computed*/ *#pairs_computed* and recall is the ratio *#correct_pairs_computed*/*#pairs_in_gold_standard*. *CSR* is compared with a baseline classifier which is based on the Boolean Existential Model (BEM), in order to show *CSR*'s ability to generalize successfully from the training examples. The baseline classifier consults the training examples of the class "⊑", testing whether each testing pair matches exactly to any of the training examples.

Fig. 2 and Fig. 3 depict the boxplots for the precision and recall of *CSR* for the various benchmark categories. The *CSR* method, while trying to generalize, takes into account the training examples of both classes. This, in conjunction to the fact that the feature vectors are not the optimum (due to errors in the mapping of properties), there are cases where feature vectors of examples for the class "*R*" are the same with examples for the class "⊑" (this happens for instance in cases A7, F1 and F2). This affects the discriminating ability of the classifier, resulting to low precision. These problems do not apply to the baseline classifier: in cases where the testing concept pairs are almost the same with the training pairs this classifier is quite effective. The above argument regarding the behavior of *CSR* is strengthened by the results of cases A4, A5 and A6 where the mapping of properties is more difficult than in A1 to A3 where the testing concept pairs are almost the same with the training pairs. In cases A4 to A6, the training and testing examples are not completely identical (due to the replacement of properties' and concepts' labels with synonyms and due to the absence of comments in some ontologies, which affects the mapping of properties): In these cases *CSR* outperforms the baseline classifier, as it manages to generalize successfully from the training examples.

Categories R1 to R4 include real world ontologies: These cases clearly show that the *CSR* method generalizes over the training examples. For example, in the test case R1 the baseline classifier fails completely in terms of both precision (0%) and recall (0%), while the *CSR* method achieves 25% precision and 89% recall. In category R3 the *CSR* method performs poorly, because properties are defined only for the root concepts. As a result, there are many training examples with 0's in their feature vectors that prevent the classifier for generalizing properly.

To further assess the quality of the classification method, we performed ROC analysis [16] (Fig. 6). In our case, ROC analysis considers the trade off between how "good" is the classifier in classifying testing examples in the distinct classes "⊑" and "*R*". By examining the ROC area under line values of the *CSR* method in all test cases it is obvious that the classifier is always above being "fair" and in the majority of the test cases (9/15) can be characterized as "excellent". It must be stated that these values depict that, although the performance of the classifier in the class "*R*" is of no evident interest for the ontology alignment problem, the *CSR* method performs even better there.

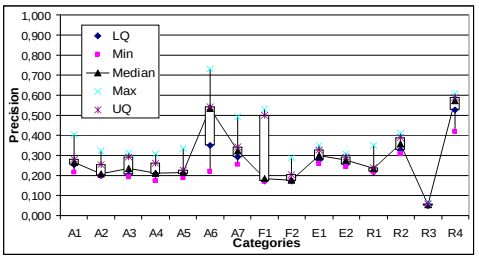

Fig. 2. Precision of *CSR* in various test cases.

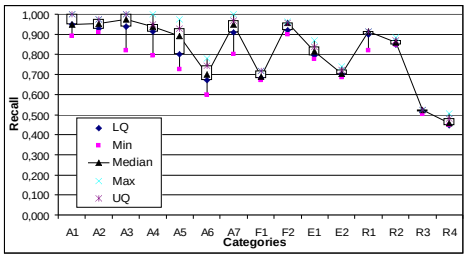

Fig. 3. Recall of *CSR* in various test cases.

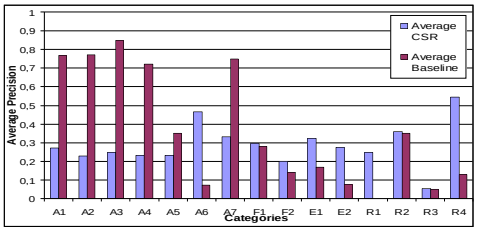

Fig. 4. Average precision of *CSR* and baseline classifier.

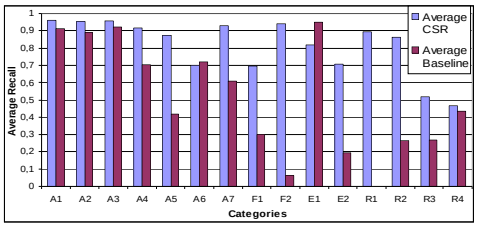

Fig. 5. Average recall of *CSR* and baseline classifier.

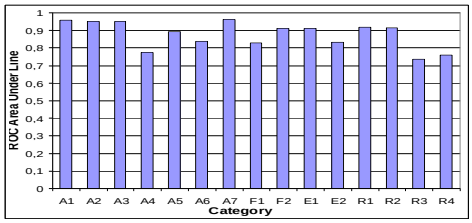

Fig. 6. ROC areas under line in all test cases.

5 Conclusions and Future Work

In this paper we propose the "Classification-Based Learning of Subsumption Relations for the Alignment of Ontologies" (*CSR*) method. Experimental results for computing subsumption relations over an extended version of the OAEI 2006 benchmarking series show the potential of the proposed method: *CSR* generalizes effectively over the training examples, showing (a) the importance of properties to assessing the subsumption relation between concepts of discrete ontologies (b) the importance of incorporating more precise property mapping methods into the process, (c) the potential to further improve the method via the incorporation of more types of features, and via its combination with other methods.

References

1. P. Shvaiko, J. Euzenat: A Survey of Schema-based Matching Approaches. Journal on Data Semantics, 2005.
2. O. Svab, V. Svatek, H. Stuckenschmidt: A Study in Empirical and 'Casuistic' Analysis of Ontology Mapping Results. In Proceedings of ESWC, 2007.
3. T. Mitchell: "Decision Tree Learning", in T. Mitchell, Machine Learning. The McGraw-Hill Companies, Inc., pp. 52-78, 1997.
4. Ross Quinlan: C4.5: Programs for Machine Learning. Morgan Kaufmann Publishers, San Mateo, CA (1993).
5. K. Kotis, G. Vouros, K. Stergiou: Towards Automatic Merging of Domain Ontologies: The HCONE-merge approach. Elsevier's Journal of Web Semantics (JWS). vol. 4:1, pp. 60-79 (2006), (Published Online First: 19 Okt. 2005).
6. F. Giunchiglia, M. Yatskevich, E. Giunchiglia: Efficient Semantic Matching. In Proceedings of ESWC, 2005.
7. Baader, F. Calvanese, D., McGuinness, D.L., Nardi, D., Patel-Schneider, P.F. (eds). Description Logics Handbook, Cambridge University Press, 2003.
8. K. Kotis, et al., AUTOMS: Automating Ontology Mapping through Synthesis of Methods, OAEI 2006 contest, Ontology Matching International Workshop, USA, 2006.
9. N. Japkowicz: The Class Imbalance Problem: Significance and Strategies. In Proceedings of IC-AI, 2000.
10. Ontology Alignment Evaluation Initiative. http://oaei.ontologymatching.org/2006/newindex.html.
11. J.W. Tukey. "Exploratory Data Analysis". Addison-Wesley, Reading, MA. 1977.