Towards a Framework for Ontology Mapping based on Description Logic Reasoning.

Quentin Reul, Jeff Z. Pan, and Derek Sleeman

University of Aberdeen, Aberdeen AB24 3FX, UK

Abstract. In this paper, we describe the Knowledge Organisation System Implicit Mapping (KOSIMap) framework, which differs from existing ontology mapping approaches by using description logic reasoning (i) to extract implicit information for every entity, and (ii) to remove inappropriate mappings from an alignment.

1 KOSIMap Framework

Ontology matching has been recognised as a means to achieve semantic interoperability by resolving lexical and structural heterogeneities between two ontologies. Given two ontologies $\mathcal{O}_1$ and $\mathcal{O}_2$, the task of mapping one ontology to another is that of finding an entity in $\mathcal{O}_1$ that matches an entity in $\mathcal{O}_2$ based on their intended meanings. As OWL ontologies are normally used with an inference engine, it is important to consider inference and reasoning as part of the ontology mapping process [2]. However, most approaches have to date disregarded the role of description logic reasoning because of efficiency reasons (e.g. [1]). While these approaches generally deliver good results, they are limited to the asserted axioms contained in the input ontologies. Therefore, the extraction of logical consequences embedded in the input ontologies may result in alignments containing less erroneous mappings.

In this paper, we describe the Knowledge Organisation System Implicit Mapping (KOSIMap) framework [3], which respects the *uniform comparison* principle [1] by restricting each comparison to entities (i.e. classes and properties) in the same category. KOSIMap consists of three main steps; namely *Pre-Processing*, *Matcher Execution*, and *Alignment Extraction*. The pre-processing step extracts logical consequences embedded in both ontologies using a DL reasoner (e.g. Pellet [4]). Note that we have enhanced its functionality by developing rules to extract further information about classes and properties. For example, we have devised several rules to extract the properties associated with a class. The pre-processing step also applies language-based techniques (e.g. tokenization, lemmatization) to lexical descriptions (i.e. labels). Next, KOSIMap applies three different types of matchers for every pair of entities (see §2). The final step extracts an alignment between two ontologies and consists of two phases. The first phase extracts a pre-alignment from the similarity matrix, by selecting the maximum similarity score above a threshold ζ for every row in the matrix. This pre-alignment is then passed through a refinement process, which eliminates erroneous mappings from the set of correspondences. This refinement approach differs from existing ones [5] by using the implicit knowledge extracted from the axioms in the first step.

2 Mapping Strategies

KOSIMap computes the lexical and structural similarity between pairs of entities based on three different matchers:

1. The *string-based* matcher assumes that domain experts share a common vocabulary to express the labels of entities. In KOSIMap, we compute the similarity between pairs of labels based on the SimMetrics library[1].
2. The *property-based* matcher first computes the overlap between two properties based on the set of properties (e.g. inferred super-properties) associated with them. It then calculates the overlap between two classes based on their respective sets of properties (e.g. inferred domains).
3. The *class-based* matcher first computes the similarity between two classes based on the set of classes (e.g. inferred super-classes) associated with them. It then calculates overlap between sets of binary relations[2] for each pair of object properties.

The property-based and class-based matchers rely on the *degree of commonality coefficient* to compute the overlap between two sets $\mathtt{S}_s$ and $\mathtt{S}_t$. The degree of commonality coefficient is defined is defined as the sum of the maximum similarity for each element in source set (i.e. $\mathtt{S}_s$). The aggregated scores from these matchers are then stored in a similarity matrix.

Acknowledgements: The IPAS project is co-funded by the Technology Strategy Board's Collaborative Research and Development programme and Rolls-Royce (project No. TP/2/IC/6/I/10292).

References

1. J. Euzenat, D. Loup, M. Touzani, and P. Valtchev. Ontology alignment with OLA. In *Proc. of the 3rd Evaluation of Ontology-based Tools Workshop (EON2004)*, 2004.
2. N. F. Noy. Semantic Integration: A Survey Of Ontology-Based Approaches. *SIGMOD Record*, 33(4):65–70, 2004.
3. Q. Reul and J. Pan. KOSIMap: Use of Description Logic Reasoning to Align Heterogeneous Ontologies. In *Proc. of the 23rd International Workshop on Description Logics (DL 2010)*, 2010.
4. E. Sirin, B. Parsia, B. C. Grau, A. Kalyanpur, and Y. Katz. Pellet: A practical OWL-DL reasoner. *Web Semantics: Science, Services and Agents on the World Wide Web*, 5(2):51–53, June 2007.
5. P. Wang and B. Xu. Debugging ontology mappings: A static approach. *Computing and Informatics*, 22:1001–1015, 2003.

[1] http://sourceforge.net/projects/simmetrics/

[2] The set of binary relation of an object property t, denoted $\mathtt{R}(t)$, is a collection of ordered pairs of classes extracted from axioms in an ontology.