Towards Explaining Semantic Matching

Deborah L. McGuinness[1] Pavel Shvaiko[2] Fausto Giunchiglia[2]
Paulo Pinheiro da Silva[1]

[1]Stanford University, Stanford, USA
{dlm,pp}@ksl.stanford.edu.
[2]University of Trento, Povo, Trento, Italy
{pavel,fausto}@dit.unitn.it

Abstract

Interoperability among systems using different term vocabularies requires mappings between them. Matching applications generate these mappings. When the matching process utilizes term meaning (instead of simply relying on syntax), we refer to the process as semantic matching. If users are to use the results of matching applications, they need information about the mappings. They need access to the sources that were used to determine relations between terms and potentially they need to understand how deductions are performed. In this paper, we discuss our approach to explaining semantic matching. Our initial work uses a satisfiability-based approach to determine subsumption and semantic matches and uses the Inference Web and its OWL encoding of the proof markup language to explain the mappings.

1 Semantic Matching

In this paper, we discuss semantic matching as introduced in [3], and implemented within the *S-Match* system [4]. We view information sources to be graph-like structures containing terms and their inter-relationships. The semantic matching distinguishes the following relations between terms: *equality* ($=$, mutual subsumption); *more general* ($\sqsupseteq$, subsumer); *less general* ($\sqsubseteq$, subsumee); *mismatch* ($\perp$, disjoint); *overlapping* ($\sqcap$, there may exist an instance of both classes). The semantic relations are calculated by mapping meaning which is codified in the element descriptions and the graphs in two steps: obtaining a representation of the node meaning and by determining the meaning of the node position in the graph. In order to obtain some information about the node labels, our initial implementation accesses WordNet. Extensions to the work would also take other DL representations of the classes as input such as full OWL ontologies. Semantic matching translates the matching problem into a validity check of the appropriate propositional formula. The algorithm then checks for sentence validity by proving that its negation is unsatisfiable. Our implementation uses the JSAT SAT reasoner.

2 Explaining Matching using Inference Web

Inference Web (IW) [6] enables applications to generate portable and distributed explanations for answers. In order to explain semantic matching and thereby increase the trust level of its users, we need to provide information about background theories (initially Wordnet), the JSAT manipulations of sentences, and the semantic matching translations of graphs into propositional sentences. The IW proof and explanation documents are represented in PML [1] and are composed of PML *node sets*. This representation could be viewed as the web-ized distributed OWL version of one author's previous work on explaining description logics [7].

Users may need different types of explanations. For example, if negotiating agents trust each other's information sources, explanations should focus on the *S-Match* manipulations. If on the other hand, the sources may be suspect, explanations should focus on meta information about sources. If a user wants an explanation of the inference engine(s) embedded in a matching system, a more complex explanations may be required, see [9] for details. Our current version of *S-Match* uses JSAT, and in particular the Davis-Putnam-Longemann-Loveland (DPLL) procedure [2].

3 Discussion

While there are a number of other efforts in semi-automated schema/ontology matching [8], we are not aware that any provide explanations. By extending *S-Match* to use the IW infrastructure, we demonstrate our approach for explaining matching systems that use background ontological information and reasoning engines[1]. The DPLL procedure explained in our approach, while unoptimized, includes the essence of the state of the art SAT engines. Thus, one could consider using another optimized SAT reasoner that may be chosen for particular matching problems and use the approach discussed for generating explanations. Future work includes using more expressive background ontologies and other SAT engines as well as other non-SAT DPLL-based inference engines, e.g., DLP, FaCT [5].

References

[1] P. Pinheiro da Silva, D. L. McGuinness, and R. Fikes. A proof markup language for semantic web services. TR KSL-04-01, Stanford University, 2004.

[2] M. Davis and H. Putnam. A computing procedure for quantification theory. In *Journal of the ACM*, number 7, pages 201–215, 1960.

[3] F. Giunchiglia and P. Shvaiko. Semantic matching. In *The Knowledge Engineering Review journal*, number 18(3), 2004. Also TR DIT-03-013.

[4] F. Giunchiglia, P. Shvaiko, and M. Yatskevich. S-match: an algorithm and an implementation of semantic matching. In *Proceedings of ESWS' 04*, 2004. Also TR DIT-04-015.

[5] I. Horrocks and P. F. Patel-Schneider. Fact and dlp. In *Automated Reasoning with Analytic Tableaux and Related Methods: Tableaux'98*, pages 27–30, 1998.

[6] D. L. McGuinness and P. Pinheiro da Silva. Infrastructure for web explanations. In *Proceedings of ISWC'03*, pages 113–129, 2003.

[7] D.L. McGuinness. *Explaining reasoning in description logics*. PhD thesis, Rutgers University, 1996.

[8] E. Rahm and P. Bernstein. A survey of approaches to automatic schema matching. In *VLDB Journal*, number 10(4), pages 334–350, 2001.

[9] P. Shvaiko, F. Giunchiglia, P. Pinheiro da Silva, and D. L. McGuinness. Web explanations for semantic heterogeneity discovery. TR KSL-04-02, Stanford University, 2004.

[1] Long version of this paper is available at http://www.dit.unitn.it/research/publications/techRep?id=549 as TR DIT-04-019 and at http://www.ksl.stanford.edu/people/dlm/papers/dl04long-abstract.html