Towards Ontology Mapping: DL View or Graph View?

Yongjian Huang, Nigel Shadbolt

Intelligence, Agents and Multimedia Group
School of Electronics and Computer Science
University of Southampton

November 27, 2005

Abstract

Ontology is important in sharing and reusing knowledge. It also plays a crucial role in the development of Semantic Web. The paper discusses the DL(Description Logic) and graph view on ontology. Different perspectives have different models and approaches on ontology mapping. The paper presents how the two different approaches handle ontology mapping, respectively. We argue that a combination of the two views can lead to a better solution in ontology mapping.

Keywords: *ontology, ontology mapping, description logic, graph, graph theory*

1 Introduction

Prior to ontology, knowledge is mainly stored in database, which makes it difficult to share and reuse. Thus, as one of the knowledge representation types, ontology is employed to represent knowledge. Neches et al. stated that "an ontology defines the basic terms and relations comprising the vocabulary of a topic area as well as the rules for combining terms and relations to define extensions to the vocabulary"[8]. Based on this definition, Gruber T. R. defined ontology as a formal, explicit specification of a shared conceptualisation"[4]. The definition is widely cited by ontology community. However, in practice, ontology is usually considered as a set of concepts and their interconnections, including some rules of inference. Sometimes it is considered as simply a taxonomy. Recently, it attracts more and more attention as it plays an important role in the development of Semantic Web[1]. Here is a section of family ontology written in OWL:

```
 <owl:Class rdf:ID="Woman">
   <owl:equivalentClass>
     <owl:Class>
       <owl:intersectionOf rdf:parseType="Collection">
         <owl:Class rdf:ID="Female"/>
         <owl:Class rdf:ID="Person"/>
       </owl:intersectionOf>
     </owl:Class>
   </owl:equivalentClass>
 </owl:Class>
<owl:Class rdf:ID="Mother">
   <owl:equivalentClass>
     <owl:Class>
```

```
      <owl:intersectionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#Parent"/>
        <owl:Class rdf:about="#Woman"/>
      </owl:intersectionOf>
    </owl:Class>
  </owl:equivalentClass>
</owl:Class>
<owl:Class rdf:about="#Parent">
  <owl:equivalentClass>
    <owl:Class>
      <owl:intersectionOf rdf:parseType="Collection">
        <owl:Restriction>
          <owl:onProperty>
            <owl:ObjectProperty rdf:ID="hasChild"/>
          </owl:onProperty>
          <owl:minCardinality rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
          >1</owl:minCardinality>
        </owl:Restriction>
        <owl:Class rdf:about="#Person"/>
      </owl:intersectionOf>
    </owl:Class>
  </owl:equivalentClass>
</owl:Class>
```

The ontology states that a *Woman* is a *Person* who is a *Female* and that a *Mother* is a *Female Parent*. *Parent* is a *Person* having at least one child.

2 Ontology Model

2.1 DL Model

Description logic is a family of formal languages developed for "representing knowledge". They are able to capture different kinds of relationships that can hold between concepts beyond is-a and part-whole relationships. The concepts in the above ontology can be described as follows:

$Woman \equiv Person \sqcap Female$
$Mother \equiv Parent \sqcap Female$
$Parent \equiv Person \sqcap \exists(\geq 1 hasChild) \cdot Person$

Another characteristic of DL is its inference mechanism, that is, to draw conclusions from the existing facts. For example, one might be interested in knowing whether *Mother* is a *Woman*, that is, $Mother \sqsubseteq Woman$? Because a *Mother* is a *Parent* and a *Parent* is a *Person*, we can infer that a *Mother* is a *Woman*. Such inference is called **subsumption**, which is the basic inference in DL reasoning. Many ontology languages such as OWL find their mathematical principles in DL.

2.2 Graph Model

The graph model we will present here is neither Existential Graphs nor Conceptual Graphs, but is one that is similar to semantic network[7]. The section of the family ontology is depicted in Figure 1:

The graph representation is often considered more attractive and effective from a practical viewpoint, because they can visually characterise the topological structure of an ontology. The well developed graph theory also provide a convenient mathematical vehicle for graph modelling.

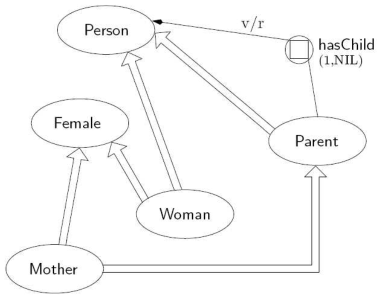

FIGURE 1: Semantic Network

3 Ontology Mapping

Due to popularity of ontologies, the number of ontologies that are made publicly available and accessible on the Web increases steadily[6]. Thus, the ontology community faces a challenge on how to manage the ontology. Researches in the name of ontology maintenance, ontology evolution, ontology mapping, ontology alignment etc. all address the ontology management problem. Particularly, significant research has been done on ontology mapping. Ontology mapping is the process whereby two ontologies are semantically related at conceptual level, and the source ontology instances are transformed into the target ontology entities according to those semantic relations. The major difficulty of ontology mapping lies on the lack of evaluation criteria. There has been an enormous amount of diverse work produced from different communities who claim some sort of solution to ontology mapping. The mathematical account of these methodologies lies either on logics or graph theory, (probably) equipped with statistics and probability approach. In our paper, we will present the two approaches towards ontology mapping by describing If-Map[5] and Similarity Network, which represent description logic and graph model, respectively.

3.1 IF-Map

The IF-Map refers to the Information-Flow-based method for ontology mapping. The basic idea is to map one ontology with another via ontology morphism, which can be defined as follows:

Given two ontologies $O = (C, R, \leq, \perp, |, \sigma)$ and $O' = (C', R', \leq', perp', |', \sigma')$, an ontology morphism $< f^*, g^* >$: $O \rightarrow O'$ is a pair of functions $f^* : C \rightarrow C'$ and $g^* : R \rightarrow R^*$, such that for all $c, d \in C, r \in R$, and $\sigma(r) =< C_1, ..., C_n >$,

1. if c $\leq$ d, then $f^*(c) \leq' f^*(d)$;
2. if c $\perp$ d, then $f^*(c) \perp' f^*(d)$;
3. if c $|$ d, then $f^*(c) |' f^*(d)$;
4. if $\sigma(g^*(r)) = < c'_1, ..., c'_n >$, then $c'_i \leq' f^*(c_i)$, for all i = 1,...,n.

Mapping the ontologies amounts to find an ontology morphism from O to O', which means that there must exist a logic infomorphism f = $< f^*, f_* >$. First, we will first look for an infomorphism between their respective classifications:

- A map of concepts $f^* :$ C → C'(concept-level);
- a map f_* from instances of C to instances of C'

Next, the ontological relations are used to guide the process that will result in the ontology mapping. Thus, we look for the infomorphism g: R → R' in a similar way stated above.

Finally, the ontology mapping is generated between local ontology and reference ontology. The algorithm can be implemented in a Prolog system.

3.2 Similarity Network

Ontology mapping can be understood as concept mapping, which seeks to find out the most similar concept in an ontology with respect to one concept in another ontology. Thus, it is important to calculate the similarity between the two concepts[2]. Similarity Network provides such a facility to carry out the computation. We can define the similarity function as follows[3]:

$$\text{sim(x,y)} = f(sim_1(x,y), ..., sim_k(x,y)), \text{ where sim(x,y)} \in [0,1]$$

Here, sim_i(x,y) refers to the similarities contributed by different factors and f is a combination function which integrates all these factors. Any concept consists of intentional and extensional parts. The intention of a concept is often characterised by a set of attributes whereas the extension of a concept is approximated by a set of instances. Thus, the sim(x,y) can be formulated as follows:

$$\text{sim(x,y)} = f(sim_{intension}(x,y), sim_{extension}(x,y))$$

If we simply use a linear combination function, then the concept similarity will be:

sim(x,y) = $w_1 sim_{intension}(x,y) + w_2 sim_{extension}(x,y))$, where w_i is the weight of the respective part.

Thus, we can compute the similarities of the concepts between the two ontologies. Based on the concept similarity, we can determine the most suitable way to map one ontology with another.

4 Further Work

In the paper, we have reviewed two approaches towards ontology mapping: DL approach and graph approach. While the DL approach can fully exploit the 'semantic meaning' of the ontology, it also exposed the disadvantage of exponential computational complexities. On the other hand, although the graph approach may not be able to calculate the complicated relationships between the ontologies, which can have complex relationship among concepts, it shows us a clear structure where mapping between two ontologies can be carried out in a statistical fashion. In the future, we will seek to combine the two approaches together. For example, the DL approach can be used to determine the relations between the intension parts of the concepts. Given the lack of objective criteria in ontology mapping and its difficulties, we also believe that the combination of different approaches should be tried to tackle this problem.

References

[1] Tim Berners-Lee, James Hendler, and Ora Lassila. The semantic web. *Scientific American*, May 2001.

[2] M. J. Egenhofer and M.A. Rodriguez. Determining semantic similarity among entity classes from different ontologies. *IEEE Transactions on Knowledge and Data Engineeering*, pages 108–113, 2005.

[3] M. Ehrig and Y. Sure. Ontology mapping - an integrated approach. *First European Semantic Web Symposium*, pages 76–91, 2004.

[4] Thomas R. Gruber. A translation approach to portable ontology specifications. *Knowledge Acquisition*, 5(2):199–220, June 1993.

[5] Yannis Kalfoglou and Marco Schorlemmert. If-map: an ontology mapping method based on information flow theory. *ISWC'03 Workshop on Semantic Integration*, 2003.

[6] Yannis Kalfoglou and Marco Schorlemmert. Ontology mapping: the state of the art. *The Knowledge Engineering Review*, 2003.

[7] Fritz Lehmann. Semantic networks in artificial intelligence. *Pergamon Press, Oxford*, 1992.

[8] Robert Neches, Richard Fikes, Tim Finin, Thomas Gruber, Ramesh Patil, Ted Senator, and William R. Swartout. Enabling Technology for Knowledge Sharing. *AI Magazine*, 12(3):36–56, Fall 1991.