

A Web Service for Efficient Ontology Comparison

James Z. Wang Farha Ali* Rashmy Appaneravanda
Department of Computer Science
Clemson University, Box 340974
Clemson, SC 29634-0974, USA
+1-864-656-7678
{jzwang, rappane}@cs.clemson.edu

Abstract

In this paper, we develop a web service for ontology comparison based on a novel senses refinement algorithm, which builds senses sets to represent the semantics of the input ontologies. The senses refinement algorithm converts the measurement of ontology difference into simple set operations based on set theory, thus ensures the efficiency and accuracy of the ontology comparison. We believe our web service is the first available online measurement tool for ontology comparison.

1. Introduction

Due to the heterogeneity and independency of the data sources and data repositories, measuring the semantic similarity of two different ontologies is critical in information retrieval, information integration and semantic web queries. Currently there is no such tool available on internet due to the complexity of existing ontology comparison algorithms and certain requirement of human involvement in these algorithms [1, 2, 3]. In this paper, we fill the void by developing a web service for ontology comparison based on a novel senses refinement algorithm, which builds senses sets to accurately represent the concepts and semantic constraints of the input ontologies.

2. Ontology Difference

We use senses set of an ontology to represent its semantics. A senses set of a class entity is a set of synonym words denoting the concept of the entity. A senses set of an ontology is obtained by extracting synonym words related to the ontology semantics from the senses sets of all concepts in the ontology. Assume the senses set of Target ontology is T and the senses set of Source ontology is S . The difference of set T from set S , denoted by $T-S$, is defined as $T-S = \{x | x \in T \wedge x \notin S\}$. Then the semantic difference between two ontologies can be calculated as $D(T, S) = \frac{|T-S|}{|T|}$, where $0 \leq D(T, S) \leq 1$.

* currently working as an instructor in Lander University.

3. Efficient Ontology Comparison

We propose a senses refinement algorithm that can efficiently and accurately measure the difference of two given ontologies.

3.1 Senses Refinement Algorithm

We propose the senses refinement algorithm in Figure 1. This algorithm automatically extracts the synonym words and relations in an ontology from the electronic lexical database WordNet [4] and refines the senses set of the ontology based on the semantic relationships between the parent concepts and the children concepts.

```
Algorithm SR(Ontology O)
begin
  Q = {};
  P = {p | p ∈ O && p is a parent in ontology O}
  for any p ∈ P
    P_flag = false;
    Sp = Senses set of p from WordNet;
    C = {c | c is a child of p in ontology O}
    for any c ∈ C
      C_flag = false;
      Sc = Senses set of c from WordNet;
      for any s ∈ Sc
        H = Hypernym set of s from WordNet;
        for any h ∈ H
          if (h ∈ Sp)
            C_flag = true; P_flag = true;
            if (h == p)
              x = c + "_is-a_" + p;
              Q = Q ∪ {x};
            else
              Q = Q ∪ {s};
          if(!C_flag)
            Q = Q ∪ {c};
        if(!P_flag)
          Q = Q ∪ {h};
        else
          Q = Q ∪ {p};
    return Q;
end
```

Figure 1: Senses Refinement Algorithm

3.2 Ontology Comparison Based on SR Algorithm

Using the proposed senses refinement algorithm, we design a simple ontology comparison algorithm in Figure 2. This algorithm takes two ontologies as the input parameters and returns their semantic difference in numeric value.

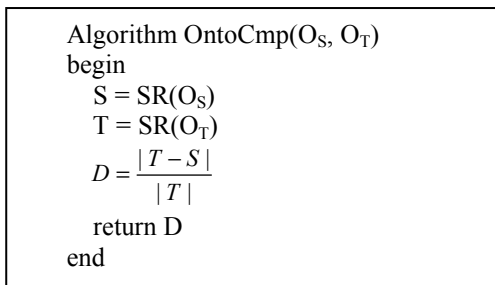


Figure 2: Ontology Comparison Algorithm

4. Web service for ontology comparison

To provide an online ontology comparison tool, we integrate our proposed ontology comparison algorithm into a web service [5], *OntoCmpService*, which accepts a pair of ontologies written in OWL and returns a numeric value to represent their semantic difference. The web service *OntoCmpService* makes use of the online electronic lexical database WordNet to generate senses sets for the input ontologies. We use J2EE SDK from Sun Microsystems as our development tool. *OntoCmpService* is implemented using JAX-RPC. It has a single interface class *OntoIF* that specifies the web service methods exposed to the public.

In *OntoCmpService*, the exposed method is *ontoCompare* which is implemented in the class *OntoImpl*. The *wscmpile* tool converts the Web Service interface to a WSDL file. We use Jena 2.1 Semantic web framework from HP labs to extract the concept labels and relationships from the input OWL files. The web service is deployed using Sun's Java systems Application Server. The interactions between the client and our *OntoCmpService* are depicted in Figure 3.

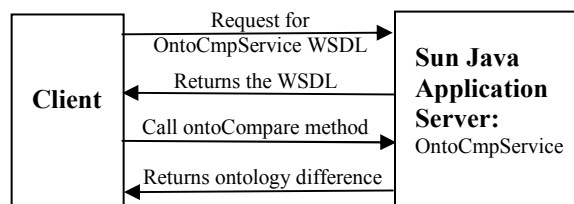


Figure 3: Interactions between client and *OntoCmpService*

When a client requests for the ontology comparison web service, a WSDL file for *OntoCmpService* is returned to the client. The client then uses the stub class generated from

WSDL file to call *ontoCompare* method for ontology comparison.

5. Conclusion and future studies

In this paper, we develop a web service for ontology comparison based on our proposed senses refinement algorithm, which builds senses sets to represent the semantics of input ontologies. The senses refinement algorithm automatically extracts senses from the electronic lexical database WordNet (locally installed or online), removes unnecessary senses based on the relationship among the class entities of the ontology, and specifies relations and constraints of the concepts in the refined senses set. The senses refinement converts the measurement of ontology semantic difference into simple set operations based on set theory, thus ensures the efficiency and accuracy of the ontology comparison.

The proposed senses refinement algorithm focuses on “is-a” relations of the class entities to discover the semantics of the ontology. Although “is-a” relation is the most common relation used in ontology, the “part-whole” relations [6], including “part-of”, “whole-of” and “has-a” relations, may be used to further define the ontology semantics. Furthermore, attributes, functions and parts may be used to denote detailed semantic features about the class entities in ontology. We are currently extending the senses refinement algorithm so that it can integrate the “part-whole” relations and semantic features of class entities into the senses set construction for ontology comparison.

6. References

- [1] A. Pease, I. Niles, and J. Li, The Suggested Upper Merged Ontology: A Large Ontology for the Semantic Web and its Applications. *In Working Notes of the AAAI-2002 Workshop on Ontologies and the Semantic Web*, Edmonton, Canada, July 28-August 1, 2002.
- [2] M. A. Rodriguez and M. J. Egenhofer, Determining semantic similarity among entity classes from different ontologies; *IEEE Transactions on Knowledge and Data Engineering*. 2003.
- [3] P. Weinstein and P. Birmingham, Comparing Concepts in Differentiated Ontologies, *in 12th Workshop on Knowledge Acquisition, Modeling, and Management*. 1999, Banff, Canada.
- [4] Christine Fellbaum (ed.), WordNet: An Electronic Lexical Database. The MIT Press, May 1998.
- [5] Online Ontology Comparison, <http://www.cs.clemson.edu/~jzwang/ontocomp.htm>
- [6] M. Winston, R. Chaffin, and D. Herrmann, A Taxonomy of Part-Whole Relations. *Cognitive Science*, Volume 11, pp. 417-444, 1987.