

## The application agents and web services based on ontology

<sup>a</sup>WANG Yi, <sup>b</sup>SUN Hui-juan

<sup>a</sup>*Department of Information Technology. Guangdong Textile Polytechnic Foshan City Guangdong Province, China*

<sup>b</sup>*Dept.of Information Engineering. Henan College of Finance & Taxation Zhengzhou, China*

---

### Abstract

In this information-exploding era, the user expects to spend short time retrieving really useful information rather than spending plenty of time and ending up with lots of garbage information. The purpose of this paper is firstly to highlight the problems that hamper the application of agents and web services separately. Our contribution is an overall solution based on a fully-fledged architecture and proof-of-concept implementation that serves as a meeting point for both service consumers and service providers through a simple web-based graphical ontology. The experiments show the algorithm could flexibly and dynamically discover resources and therefore provide a valuable addition to the field.

**Index Terms:** Ontology; Semantic web services; Knowledge management; Agent

© 2012 Published by MECS Publisher. Selection and/or peer review under responsibility of the International Conference on E-Business System and Education Technology

---

### 1. Introduction

At present, many data or service discovery processes use keyword-matching technique to find published information. This method often discontents requesters with so many unrelated results that lead to certain amount of manual work to choose the proper service according to its semantics. In order to solve this problem, semantic web technique – OWL and OWL-S, which are innovative for data discovery and service discovery respectively, have been adopted[1]. Both intelligent agent and semantic web service technologies are able to reach remarkable achievements and in some cases have overlapping functionalities.

Basically, a website model consists of a website profile for a website and a set of webpage profiles for the webpages contained in the website. Each webpage profile reflecting a webpage describes how the webpage is interpreted by the domain ontology, while a website profile describes how a website is interpreted by the semantics of the contained webpages. However, semantic web services need for an upper software entity able to deal with them and, on the other hand agent technology has historically suffered from a number of drawbacks that must be addressed. Integrating these two technologies in a joint environment can overcome their problems while strengthening their advantages. In this paper, the necessity for integrating these technologies and the potential benefits of their combination are analyzed.

Corresponding author:

E-mail address: <sup>a</sup>wangyi0705@163.com, <sup>b</sup>xhs\_js@sina.com

We presented a new aspect of ontology quality, developed a methodology for its assessment and showed its application. This initial study has demonstrated the need for further research and improvements to the methodology. Based on this study, we present SEMMAS, an ontology-based framework for seamlessly integrating Intelligent Agents and Semantic Web Services. We are currently working on applying the method to other upper-level and domain ontologies, and augmenting the method with other techniques from cognitive research. The basics of the framework are detailed and a proof-of-concept implementation described.

## 2. Semantic web services

### 2.1. Ontology

Ontology plays a pivotal role in the development of the Semantic Web. It can provide a representation of a shared conceptualization of a particular domain that can be communicated between people and applications. Compared with databases, ontology is more efficient and flexible in capturing and managing knowledge about concepts in the domain and relationships between these concepts. Web services represent a step forward in enabling collaborations between various entities on the web and in overcoming the interoperability problems that may appear. B2B (Business to Business) partners can benefit by allowing business entities to expose their capabilities and to make use of their functionality. Information systems based on a service-oriented architecture which are able to integrate different functionalities and to offer a virtual component model that abstracts from the peculiarity of specific implementations, seem to be a very appealing solution.

The W3C is currently examining various approaches with the purpose of reaching a standard for the SWS technology: OWL-S, WSMO, SWSF, WSDL-S, and SAWSDL. For this, it is necessary to link WSDL elements with semantic models, such as ontologies. SAWSDL is not attached to any semantic representation language, but it incorporates the mechanisms by means of which it is possible to reference semantic model concepts from WSDL documents.

### 2.2. Website Model

A webpage profile contains basic information, statistics information, and ontology information about a webpage. To produce the above information for website modeling, we use DocExtractor to extract primitive webpage information as well as to perform statistics.

Requester plans to search some interested resource, however, he didn't know which kind of resources he wants. According to the requester's resource requirement advertisement, unified discovery system locates it in the related friend groups through getting group location algorithm firstly; then it utilizes locating resource algorithm to start searching resource in related groups; and finally, the discovery results containing both data and services, return to the requester [2]. Fortunately, the customer service database also contains two remark fields. A symptom remark is associated with the symptom field and is used for describing the details about the problems or troubles experienced by the customer. A *process remark* is also available and is associated with both the *cause* and *process* fields. It is used to keep track of the service's progress in managing the problem.

This paper describes the various auction processes that could be entered into and, as an illustrative example, it describes how web services execution environment could be used to implement an auction process. Sample descriptions in WSML ontologies within the WSMX environment are also presented to exemplify the convenience of developing such auction processes.

### 2.3. Data analysis and View

We also propose an implicit webpage expansion mechanism oriented to the user interest to better capture the user intention. This user-oriented webpage expansion mechanism adds webpages related to the user interest for further retrieval into the website models. Here we exploit the outbound hyperlinks of the stored webpages in the website models. To be more precise, we are using Anchor Texts specified by the webpage designer for the hyperlinks, which contain terms that the designer believes are most suitable to describe the hyperlinked

webpages. It is these remark fields that constitute a significant tool for developing a better method and means for understanding what is really taking place. This is possible because these comments provide hidden information that can be used to ‘refurbish’ the database by re-designating the “*other*” values in the correct remark fields. To be more specific, the first thing about refurbishing a database is to replace the “*other*” values with proper terms extracted from the correspondent remark fields.

Users can put service input, output and QoS grading into that file. QoS grading is user’s requirement for service quality. After receiving query description from user, discovery system begins to do service matching process with service matching algorithm. URLs of the matched services will be passed on to requester.

With the use of the graphical tool, more and more data will be accumulated in the databases. The relations hidden in the data have high value. Data analysis is therefore implemented to mine the useful relations in the database. The ontology, which is good at the management and reuse of relations, is used to save the results of data analysis. Algorithm 1 gives the detailed steps to evaluate the structural validity of ontology, Algorithm 1:

```

Input: Resource  $r \in R$ , Leader  $l \in L(Fgs)$ , Threshold, thesaurus ontology Tho;
/*where R is a set of classes in thesaurus ontology;
L(Fgs) is a set of Friend Group leaders; Threshold is the threshold value which is given by discovery system.
*/
Output: ListOfNum list;
/* One list of groups’ numbers to which r belongs; */
step1:
BuildTree(Tho);
/* Build a tree structure for thesaurus ontology, and terms of the latter map to the nodes of the former; */
step2:
for (all  $l \in L(Fgs)$ ) do
if ( SimBetweenNodes (r, l) > Threshold)
then do
    number $n$   $\leftarrow$  the group number of l;
    list.add(n);
/*Add the selected Friend Group number to list. */
endif
endif

```

### 3. Research design and Implementation

#### 3.1. Query process

The classes of the ontology correspond to the tables of the database: the name of a class in the ontology is consistent with the name of the table in the database, and the slots of the class correspond to the column of the table. In the equation,  $N_{S,D}$  refers to the number of webpages on website  $S$  belonging to domain  $D$ ; and  $N_S$  stands for the number of webpages on website  $S$ . Here we need the parameter Domain\_Mark in the webpage profile to determine  $N_{S,D}$ . In short,  $R_{S,D}$  measures how strong a website is related to a domain[3]. Thus, the autonomous webpage expansion mechanism is also domain-oriented in nature.

$$R_{S,D} = \frac{N_{S,D}}{N_S} \quad (1)$$

Protégé is an open-source development environment for ontologies and knowledge-based systems. Protégé OWL Plugin has a user-friendly environment for editing and visualizing OWL classes and properties. It allows users to define logical class characteristics in OWL by a graphical user interface.

With the support of these ontologies, the symptom remark can be used to refurbish the *symptom* field, while the process remark can be used to refurbish the *cause* and *process* fields [4]. Now, we can discuss how Autonomous Website Evolver does domain-dependent website expansion. Autonomous Website Evolver employs a four-phase progressive strategy to autonomously expand the website models. The basic refurbishing process is outlined below, while a more detailed description is given in the next section. Note that all the operations are performed on the original database.

### 3.2. System evaluation

Data instances can be interpreted correctly only if they are semantically valid with respect to the ontology they comply with. The first experiment is to learn how well OntoClassifier works. We applied the feature selection program as described in ontology-reorganization to all collected webpages to select ontology features for each class. Table 1 shows the number of features for each class.

TABLE I. THE ONTOLOGY RESULT FOR DIFFERENT QUANTITY OF WEB PAGES

Stage	Stage 1	Stage 2	Stage 3
web pages	430	635	766
Data of ontology	557	63	64
Deep of ontology	232	73	35
D	563	63	73
W	56	53	35
T	783	73	74
s_e	34.3 %	45.5 %	46.6 %

Formal ontologies have two main purposes. They represent a specific domain and they are used as computational entities for reasoning purposes. In this paper we are concerned only with the representational aspect of an ontology.

Note that we propose to evaluate only one aspect of quality among many, other aspects of quality discussed in this section are of course important as well. Providing an integrated account of quality is beyond the scope of this paper.

In this paper, by introducing the semantic similarity of domain ontology and using a heuristic algorithm for finding neighbouring resource agents in a decentralization manner for the discovery of resources, without the overhead of negotiation, a resource agent only needs to know neighbour nodes he is connecting with and what the semantic descriptions of neighbouring resource agents are. The method of using semantic similarity provides a flexibility to allow individual resource agents to perform flexible semantic matches[5].

a) Find the equivalent concepts of  $c, eq(c)$ , and the superclass concepts of  $c, sup(c)$ . Essentially, we apply the Inter-concept Def-Def Dependency and Inter-isosem Dependency.

b) For each property  $p$  instantiated by  $i$ , find the equivalent properties of  $p, eq(p)$ , their inverse properties, denoted as  $inv(p)$ , and their superproperties,  $sup(p)$ . These properties may belong to the same concept as  $p$  or any concept in the set obtained in step 1 and 2. This is based on Intra-isosem Dependency and Inter-concept Des-Des Dependency.

### 3.3. Experimental design and procedure

To measure the performance of ontology and P2P based data and service unified discovery system and to verify the means of service matching with QoS presented in this paper, we have developed a prototype of the system. This framework is made up of modules of knowledge base, inference engine, explanation facility, user

interface, and developer interface. The knowledge base module is the core of the system; it consists of domain knowledge of financial statements and operational knowledge of analysts' decision rules. The user interface accepts the original facts from financial statements and invokes the inference engine to activate the decision rules in the operational knowledge base. The proposed system development framework is shown in Fig. 1.

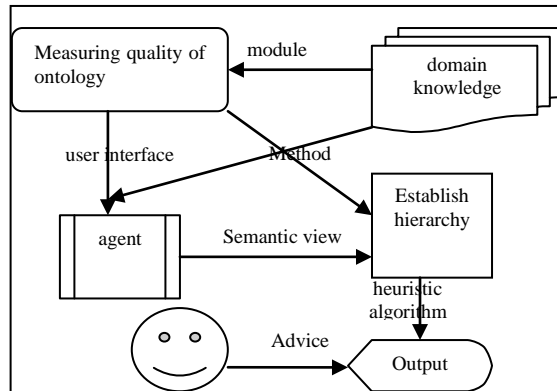


Figure 1. The system framework of ontology

Our system runs on a Windows XP-based PC with an Intel Pentium 4 CPU at 2.6 GHz and 512 MB RAM. The system was developed using Java on Java SDK 1.4.2 platform. The exemplified database is a real-life customer service database spanning from December 2003 to February 2004. The database originates from one of the major telecommunication companies in Taiwan. It contains three categorizing fields, namely, *symptom*, *cause*, and *process*, and two remark fields, one related to *symptom* and the other containing information about *cause* and *process* [6]. We separate the knowledge of financial statements analysis into domain knowledge and operational knowledge to develop an expert system for assessing financial quality of an enterprise. The former represents the well defined relationships of accounting items of financial statements, and the later represents the analytical processes of the analyst during the diagnostic process.

3.4. Experiment results

Though unified discovery system is more complicated than data or service discovery, it does not cost more search time than them. Fig.2 shows our second test result and it demonstrates that unified discovery costs less time than service discovery.

```

    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:someValuesFrom>
          <owl:Class rdf:about="# operation "/>
        </owl:someValuesFrom>
        <owl:onProperty>
          <owl:ObjectProperty rdf:ID=" resource "/>
        </owl:onProperty>
      </owl:Restriction>
    </rdfs:subClassOf>
  
```

Figure 2. The RDF format of resource ontology .

In terms of the above experimental analysis results, we can find that the task success probability is obviously influenced by the threshold of semantic similarity under different task complexity, task load and network scale.

However, the power of the test is lower than in the first experiment, due to the smaller sample size and there are a few more concept sets that show positive but not significant S2-S1-TD values which may become significant with a higher-powered test. Fig.3 shows the detailed comparison results of A and PA.

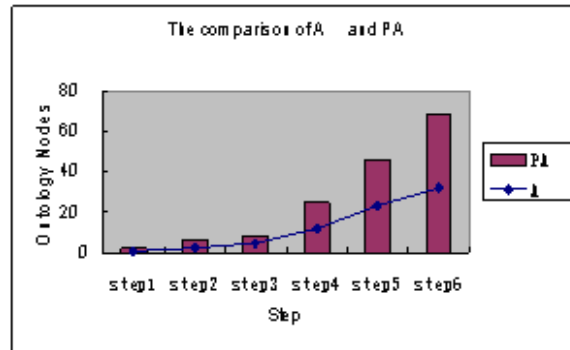


Figure 3. The compared result of PA and A

### 3.5. Experiment Discussion

Finally, we test the average cost time of distributed discovery system and centralized discovery system. Fig.4 verifies that efficiency of the former is much higher than that of the latter. In our distributed system, we utilize resource domain ontology to divide resources into different groups and searching bound declines with these different groups, so it raises efficiency.

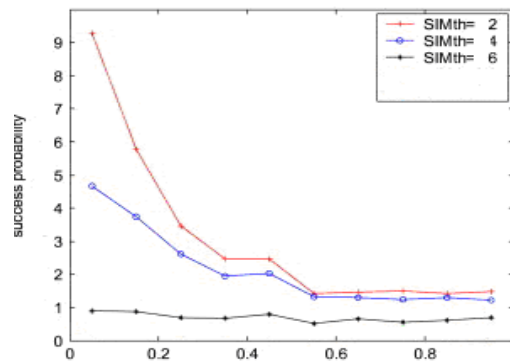


Figure 4. The compared result of similarity threshold settings

We first change the length of tasks, namely task complexity, which means resources required by tasks are different. We select different task lengths as 4, 8, 12 and 16. The task loading rate is represented by the number of tasks published or generated (the rates of task generation vary between [0.05, 1]). We calculate task success

probability with different similarity threshold settings (SIMth). The experimental results are shown in Fig.4. The algorithm can adapt its connectivity within the small world network structure. Similarly, the algorithm can also be applied to other network structures by the self-adapting policy.

When identifying the semantic view for a data instance, the concept of the data instance can be identified through API function *individual-direct-types*, its equivalent concepts and superclass concepts can be identified through API function *AR* and *PA*, respectively. Similarly, the concept as the range of an object property of the data instance can be identified along with its own equivalent concepts and superclass concepts. For each property of the data instance, the semantic view will also include its equivalent properties identified through *PA*, its super properties identified through *PA*, and its inverse property identified through *AR*. These concepts, the taxonomy/associations among them along with their restrictions and axioms are part of the semantic view.

#### 4. Conclusions and future directions

Agent technology promises to enable cost-effective, distributed systems that are powerful and flexible. However, several problems arose that have prevented multi-agent systems from being applicable in real-world settings. A major flaw of the technology is the use of proprietary protocols, what makes it difficult for agents that have not been designed to work together to interoperate. In this paper, we develop a novel decentralized heuristic algorithm by introducing semantic similarity of domain ontology and using a decentralized mechanism of finding neighbours. Our experiments demonstrated that this methodology allows more significant terms and more useful troubleshooting rules to be derived from the refurbished database and the total accuracy of the set of mined rules to be significantly increased.

This paper presented a method to study the cognitive quality of ontologies. Our findings suggest that none of three ontologies that we examined is a good specification of the conceptualization of the domain. Our findings suggest that both the terminology and the structure of the ontologies may benefit from improvement.

The future work will focus on how to adapt various strategies of the resource discovery under the circumstance of network node failures and how to introduce security aspects into the proposed model.

#### References

- [1] Berners-Lee T, James Hendler , Ora Lassila. The Semantic Web[J]. Scientific American,2001,5.
- [2] R. Ahmed and R. Boutaba, Distributed pattern matching: A key to flexible and efficient P2P search, *IEEE Journal on Selected Areas in Communications* 25 (1) (2007), pp. 73–83.
- [3] L. Qin and V. Atluri, Ontology-guided change detection to the Semantic Web data, in: 23rd International Conference on Conceptual Modeling (ER 2004), pp. 624–638.
- [4] C. Mastroiannia, D. Taliab and O. Vertab, A super-peer model for resource discovery services in large-scale grids, *Future Generation Computer Systems* 21 (8) (2005), pp. 1235–1248.
- [5] Missikoff M, Navigli R, Velardi P. Integrated Approach for Web Ontology Learning and Engineering[J]. *IEEE Computer*, 2002.35(11):60-63.
- [6] W.T. Tsai, Service-oriented system engineering: a new paradigm, in: Proceedings of IEEE International Workshop on Service-Oriented System Engineering (SOSE), October 2005, pp. 3–8.