Matching Algorithm of Web Services Based on Semantic Distance

Pengbin Fu1, Shanshan Liu2, Huirong Yang2, Liheng Gu2
1 Beijing university of technology, Beijing, China
Email: fupengbin@bjut.edu.cn
2 Beijing university of technology, Beijing, China
Email: {serina, yanghuirong, guliheng}@emails.bjut.edu.cn

Abstract—With the growing number of web services, the importance of matching and discovery web services is increasing. Using domain ontology to describe the semantic of web services and matching the web services on the semantic level is a hot research pot. In this paper, the problem of matching web services is transformed to the computation of semantic similarity between concepts in domain ontology. We propose the semantic similarity can be measured from the semantic distance and consider the factors of path length, depth, local density and number of downward direction in the algorithm. We establish that such a measure is an effective means of discriminating services at a level of granularity that is able to enhance the matching process in semantic web services.

Index Terms—domain ontology, matchmaking the Web Services, semantic similarity, semantic distance

I. INTRODUCTION

With the increasing number of available web service, discovery of correct web service for our needs from numerous web services becomes a key problem of web service system. Web Service Description Language (WSDL) [1] describes the way of invoking operation mainly, but it lacks of an explicit capability representation. Universal Description, Discovery and Integration (UDDI) [2] provides a web wide registry of web services but its syntax based search provided results that are coarse in nature. The precision and recall can’t satisfy our requirements. How to describe the semantic information of web services and matching semantic web services becomes the focus of the research.

In semantic web services matching, service providers can advertise their web services with a well-defined description language and ontology, such as DAML-S and OWL-S [3]. An ontology defines a conceptualization of a domain related to concepts, attributes, and relations. They are typically structured into a taxonomy tree where each node represents a concept and each concept has its parent as general concepts. The matching system then allows services requesters to upload their requests which are encoded in specific description language and ontology as well. From this point, the matching system determines the relationship between the requests and the registered services in an ontology.

II. RELATED WORK

Currently, there are several algorithms and architectures are proposed for semantic web service matching based domain ontology such as Massimo Paolucci et al. [4], Tang S et al. [5] and Hui Peng et al. [6]. In Paolucci’s model, semantic similarity is divided into Exact, Plug in, Subsumes, Fail based hyponymy (IS-A) of concepts which are corresponding to the Input and Output of web services. This becomes a classical algorithm in matching semantic web services, which is cited by other algorithms usually or is the base of comparing with different algorithms, but the differences between concepts is simple badly, it can’t fit to the instance of so many web services in network evidently.

Another solution proposed for matching is Tang-s’s architecture. Their design and implementation of a service matching prototype is based on DAML-S descriptions. The algorithm has four stages including input parameter matching, output parameter matching, profile matching and user-defined matching. There are nine degrees of similarity determined not only by the relation of subClassof between concepts , but also by the relation of subPropertyof. Degrees of the match are also organized in a discrete scale. There is no match value defined either.

In [6], they defined a piecewise function to calculate similarity between concepts in an ontology. The results compared with Paolucci’s model. The similarity of Exact, Fail, Plug in, Subsumes in their algorithm is respectively 1.0, a real number more than 0.5 and less than 1, a real number more than 0 and less than 0.5. They taked 0.5 as the separatrix which produced the results aren’t precise either.

According to above mentioned, aim at the problem of inexact algorithm of matching web services , This paper proposes that matching in semantic web services can be enhanced through the use of measures that quantify the ‘semantic distance’ between concepts in web services ontology. We demonstrate that a matching process based on Semantic Distance Measures will overcome the issues discussed above by refining and quantifying the degrees of the matching.

III. SEMANTIC DISTANCE MEASURES IN WEB SERVICES MATCHING

In this paper, we contend that semantic distance can represent in quantitative terms the degree of matching between a service request and a service advertisement. The Paolucci et
al proposed algorithm used discrete degrees to evaluate the similarity between concepts which there are certain limitations. We now illustrate how these degrees of similarity can be strengthened by precise quantitative measures of similarity.

The following formula can be used to have semantic distance measurement as well as satisfying all the requirements for matching web services.

A. Reference Model

As we can see from the example ontology in figure 1, the similarity or the distance between services implicitly exists. Therefore, there is a need to exploit this similarity or distance by the use of quantitative and explicit measures. By using semantic distance, the similarity or the relatedness between concepts can be measured. The concepts refer to a particular sense of words or services in term of web services. According to Budanitsky in [8], there are three principal approaches that can be used to measure the distance between concepts dictionary-based, Thesaurus-based, and semantic networks. According to Lee et al. in [9], semantic networks are defined as “any representation interlinking nodes with arcs, where the nodes are concepts and the links are various kinds of relationships between concepts.” Since this view of a semantic network has structured similarity to web services ontologies, we will focus the discussion only on this approach. In semantic networks, WordNet is widely used as the encoding of lexical knowledge.

There are several measures of Semantic Distance that have been developed. In this research, we modify the SDM proposed by Yuhua Li [10] to facilitate application in the context of semantic web services. The definition of semantic similarity is defined by the following formula:

\[
S(w_1, w_2) = e^{-\alpha t} \cdot \frac{e^{\beta h} - e^{-\beta h}}{e^{\beta h} + e^{-\beta h}}
\]

Where \( \alpha \geq 0 \) and \( \beta > 0 \) are parameters scaling the contribution of shortest path length and depth, respectively.

We now explain our proposed modification of the Semantic Distance formula in formula 1 proposed by Yuhua Li’s in [10]. In their paper, the formula is used to measure the relations between nouns in Word-Net. However, our SDM algorithm applies the measurement for services in web service ontology. The concepts of nouns in [10] is replaced with web services inputs and outputs parameters while the taxonomy tree of WordNet in [10] is replaced with the ontology of web services.

B. Our Proposed Model as Follows:

We propose that the similarity of SD(S₁, S₂) between S₁ and S₂ (S₁ is the requested service from user and S₂ is the advertised service.) is a function of the attributes PathLength, Depth, Local Density and NumberOfDownDirection as follows:

1) PathLength: Let PathLength be the shortest path length between S₁ and S₂ in ontology. The greater the PathLength, the lower the similarity, such as the relation between Vehicle and Sedan is closer than Vehicle and Lexus. S₁ and S₂ can be determined from one of two cases:
   - S₁ and S₂ are the same node in ontology, then PathLength = 0.
   - S₁ and S₂ are not the same node, let pathlength be the number of edges counted from S₁ and S₂ in the ontology. For example the service Lexus and Suv have the value of PathLength = 4.

2) Depth: Depth denotes the deep of the services in the ontology. Concepts which are located deeper in the tree tend to be more closely related to one another than those higher in the tree. Therefore, we consider assigning a different weight for different level in the ontology. For example in figure 1, the services which sell Lexus and Cadillac(level 5 in the ontology) should be closer to one another than the services which sell Sedan and Suv(level 3 in the ontology).
   - Let Lw(S) in (2) be the level weight for each path in ontology.
     \[
     L_w(S) = \frac{l}{n}
     \]
     Where \( l \) is the deep of services, \( n \) is the deepest in the ontology. We define the level of the first layer is 1/5, the level of the second layer is 2/5.
   - Let Lws in (3) is the level weight sum of the matching services.
     \[
     L_{ws} = L_w(S_1) + L_w(S_2)
     \]

3) Local Density: Local density is mean that the number of the sibling nodes of service. The two nodes that are in the same deep, which in the higher local density one’s semantic similarity is higher than the one in the lower local density, such as the relation between Buick Excelle and Buick Lacross is closer than Lexus and Cadillac. Let Den in (4) be the local density weight.

\[
Den = \gamma + (1 - \gamma) \times \frac{\bar{E}}{E(p)}
\]

Where \( \gamma \) (0 \( \leq \gamma \leq 1 \)) controls the local density of nodes in terms of the impact of the right. E(p) represents the number of the child nodes of p(the number of leaf/the number of branch). \( \bar{E} \) is the average density of the whole hierarchy. Take Family
for example, $E = 12/8 = 3/2$, $Den = (1 - \gamma) \times 3/2/3$, ($\gamma = 0.5$), $Den = 0.75$.

4) **NumberOfDownDirection:** Let NumberOfDownDirection be the number of edges counted between service $S_1$ and $S_2$ which direction is downward. For example, figure 1, Lexus and Suv have 1 value of NumberOfDownDirection, while Suv and Lexus have value of 3. This can satisfy that a requested service and an advertised service which has the child-parent relationship in the ontology should be higher in similarity than these services which have the parent-child relationship.

Finally, the interval of similarity should be $[0, 1]$. We use exponential-decay function and define our formula in (5).

$$D(S, S') = \begin{cases} 1 & \text{if \text{PathLength}=0} \\ e^{-\alpha -\beta \times \text{PathLength} - \text{Numbe} \text{rofDownDirection} - \text{Den}} & \text{if \text{S and S}' are sibling} \\ e^{-\alpha -\beta \times \text{PathLength} - \text{Depth} - \text{Den}} & \text{if S is S'} \text{ sibling} \end{cases}$$

(5)

where $\alpha \geq 0$, $\beta > 0$ are smoothing factors. They control PathLength and Depth in terms of the impact of the right.

IV. **PERFORMANCE ANALYSIS AND TEST**

The performance measuring of web service discovery doesn’t have a Unified Standard at present. Lots of definition method for web services and lacking of special web services testing sample sets that lead to the difficulty of web services discovery. In this paper, we use the distance education system constructed by our group as the testing sample sets. That contains the information about course, teaching affairs, exam database and so on. We developed many web services to operate the information in database such as search people, search subject, search college, browse exam database, input subject and so on. We created the education technology domain ontology and described the web services with owl-s. We published the profile file to JUDDI and requester search web service form JUDDI, then requester get the result from JUDDI.

We select course information process web services, teaching affairs information process web services and exam database process web services per 30. In this paper, we test using the matching algorithm in [4],[6] and this paper’s exact match, $D = 0.1$, $E = 1$. The precision of matching result is: more than 0.2, more than 0.4, more than 0.5, more than 0.6, more than 0.8 or 1.0. The result more than 0.2 or more than 0.4, In [4] we chose the subsumes as the result, the result more than 0.5,more than 0.6 and more than 0.8,in [4] we chose the plug in as the result. The test result as figure 2 and figure 3 shown(In figure 2, the x-axis is similarity and the y-axis is recall. In figure 3, the x-axis is similarity and the y-axis is precision).

Figure 2 The Recall of The Three Algorithms

Figure 3 The Precision of The Three Algorithms


V. **CONCLUSION AND FUTURE WORK**

In this paper, we have proposed a formula to measure the semantic distance between services through concepts specified in an ontology. In the formula, we considered the factors of semantic distance, node depth, local density. We established that Semantic Distance Measures are suitable for determining similarity between requested and advertised web services. In summary, the primary contribution of our research is that we have developed a Semantic Distance Measure to provide a quantitative similarity measures to support matching in semantic web services. Currently, our model does not support multiple inheritance ontologies. Therefore, the enhancement of our model to incorporate Semantic Distance Measurement between services in multiple inheritance ontologies is a proposed extension.

REFERENCES


[3] OWL-S Coalition. OWL-S 1.0 Release. Available at http://www.daml.org/services/owl-s/1.0/ (last accessed November 05, 2005).


