Association for Information Systems

AIS Electronic Library (AISeL)

ICEB 2010 Proceedings

International Conference on Electronic Business (ICEB)

Winter 12-1-2010

Semantic Information Based Web Service Discovery

Huiying Gao

Qian Zhu

Xiuxiu Chen

SEMANTIC INFORMATION BASED WEB SERVICE DISCOVERY

Huiying Gao, School of Management and Economics, Beijing Institute of Technology, Beijing, China, E-mail: huiying@bit.edu.cn
Qian Zhu, School of Management and Economics, Xinjiang Agricultural University, Urumqi, China, E-mail: zhuqian_1112@hotmail.com
Xiuxiu Chen, School of Management and Economics, Beijing Institute of Technology, Beijing, China, E-mail:xiuxiuchen77@163.com

Abstract

Web service discovery is an important part of Web service applications. With the increase of the Web services, there come the problems that we must face to. The technology of traditional Web service discovery is based on the matching foundation of key words by combining WSDL and UDDI standardization. However, it is short of the description of semantic information, which means a low level of intelligence. Thus, it leads to low exactness and completeness.

In the light of this issue, the paper draws the concept of semantic information into the research of Web service discovery. Through WordNet, it expands service description, raises semantic information, and makes semantic match possible. The paper puts forward a model based on semantic for Web service discovery, analyzes the function structure and the relationship between component parts of the model, moreover, it describes the work flow of the model. At the same time, the paper enlarges UDDI information and gives a matching algorithm on the similarity calculation through semantic analysis when finding the matched service.

Key Words: Web service discovery; Ontology; Service Description; Semantic expanding

I. Introduction

As a new Web application model, Web service is not only a new distributed computing model, but also an effective mechanism for data and information integration on the Web. With the rapid development of information technology, Web services has been widely used in variety fields, such as distributed computing, electronic commerce and so on.

Web service discovery is an important part of service oriented architecture. Finding services that meet users' demands is the prerequisite of service reuse and composition. The effect of service discovery directly relates to the quality of service reuse, the compatibility and substitution of service composition and also the ability to realize service "plug and play". So to find appropriate Web services initially is the key to Web service interoperation. Nowadays, UDDI based service matching framework is the most popular Web services matching technology. However, since lack of semantic information support, UDDI can only provide a basic description of services and frame-based matching mechanism, which makes the matching of the Web services only based on the foundation of key words and syntax. This way of matching can not distinguish different syntax but the same semantic information, but also different semantics but the same syntax information, and therefore, it can not provide service function based on the semantic matching [1].Meanwhile, this kind of Web service discovery is short of description of semantic information, which means it cannot understand user's messages intelligently. Thus, it leads to low rate in exactness and completeness, and it's pretty difficult to satisfied the increasingly needs of the modern society. All of these problems have restricted the accuracy and efficiency of Web service discovery.

In the light of this issue, the paper draws the concept of semantic information into the research of Web service discovery. The concept of semantic Web services is proposed by TimBemers-Lee firstly, and it is not a new Web form, but expansion and extension of the existing Web [2]. Web services based on the semantic can do: a clear connection and rules between description and reasoning affairs, a clear description of the tasks performed by Web services, so as to realize the automation of Web service discovery.

Semantic information description and matching of services has become a focus. In order to resolve the problems above, many scholars have done deep researches. Chakraborty presents a service discovery system with DAML as semantic Web service description language and Prolog as reasoning language [3]. Based on process ontology, Mark Klein and Abraham Bernstein put forward a service discovery method, which can achieve service discovery through semantic matching between the inquiry process ontology and service ontology with a process query language (PQL) search service [4]. In China, how to extend the semantic Web knowledge representation language and achieve the automatic search, discovery, matching and integration for Web services is studied in APEX Data and Knowledge Management of laboratory affiliated to Shanghai Jiaotong University. By introducing Web ontology, the management and implementation mechanism of the semantic intelligent Web services is discussed in the Knowledge Engineering Laboratory in computer Science Department of Tsinghua University [5]. Fengfeng Liang studied on distributed semantic Web service discovery based on ontology. [6]

However, constructing ontology is a fairly heavy engineering in semantic Web services. In this paper, through WordNet, we add semantic annotation to the WSDL service description, expand Web service description, map to UDDI, and then improve the rate in exactness and completeness. There are 5 sections in this paper. In section 2 a semantic model for Web service discovery is put forward and the work flow of the model is described exhaustively; after that in section 3 how to enlarge semantic information for Web descriptions is studied and a matching algorithm on the similarity calculation through semantic analysis when finding the match service is designed; then in section 4 the feasibility of the model is validated by a small example; finally, section 5 displays the conclusion and outlook.

II. Service Discovery Framework

Each module of the semantic Web service discovery model bears part of the system sub-functions, and they cannot complete service discovery without mutual cooperation [7]. Through the complete Web service discovery process, the whole system provides a range of services for service providers and service requesters. The framework of semantic Web service discovery is illustrated in Fig.1. It is composed of User Interaction Module, Search Matching Module, Semantic Annotation Module, User Information Database as well as UDDI Registry and WordNet[8].

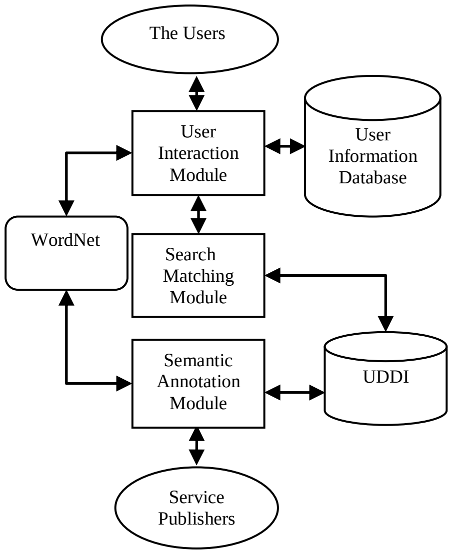

Figure 1.The framework of semantic Web service discovery

A User Interaction Module is necessary to provide a user communicate interface. A Search Matching Module with matching algorithm is required for the query process. Additionally, Semantic Annotation Module will add semantic information for a service description when service providers publish their services and this preparatory work improves the accuracy of the query. There are three roles (service provider, service registry and service requestor) and three operations (publish, find and binding) in the Web services architecture. Semantic expansion is necessary in the process of publish and find operations.

A. The Process of Publishing Services

The process of publishing services is illustrated in Fig.2.

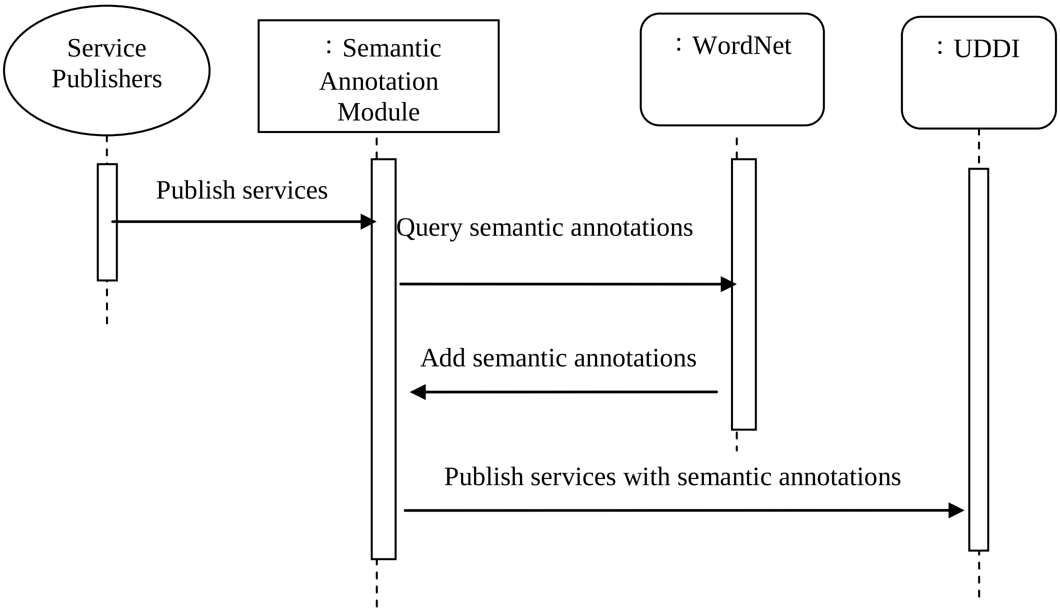

Figure 2 Sequence diagram of the publishing process

The main function of semantic annotation module which is the main part of the process of publishing services is to add semantic information for a service description when service providers publish their services. Web services cannot be found and used before they have been sent into UDDI registry. The technology of traditional Web service discovery is based on the matching foundation of key words. However, UDDI is short of description of semantic information, which means a low level of intelligence. The semantic annotation module is designed, which will add semantic information for a Web service description when service providers publish their services and this preparatory work improves the efficiency and accuracy of the search matching. Moreover, the semantic annotation module is directly connected with the WordNet which is an indispensable tool for many semantic judgments through the whole Web services discovery process. [8]

B. The Process of Services Discovery

The services retrieval process is shown in Fig.3.The User Interaction Module directly provides user with a user interface. Firstly, users input their login information and then put the request information into the user interface; meanwhile, User Interaction Module transforms the user service request in accordance with the knowledge and rules in the User Information Database. When there is polysemy in the user's request information that cannot judge according to user's background information, then it is necessary for the users to make independent determination on their demands. During this period, we cannot judge whether there is ambiguity in the user's input information without WordNet. At the same time synonym will be stored in a set. When Search Matching Module receives the request information, it will judge semantic information with WordNet and query the appropriate service in the UDDI Registry according to the matching algorithm. In section 3 a matching algorithm on the similarity calculation through semantic analysis is described exhaustively. Based on the priority level of matching results, Search Matching Module sends the information back to the User Interaction Module and then user interaction module delivers it to the users.

III. Semantic Matching of Web Service

In service-oriented architecture, Web service is considered as a convention between service provider and the requester. Although these conventions differ in the integrity and accuracy, their ultimate goals are to make requestors find needed services precisely. In fact, the service description not only just defines a simple series of available operations, but also describes desired behaviors and constraints in some special circumstances in order to know the combination and interaction of services during the follow-up search process. The semantic Web services description is the cornerstone of semantic Web service. Therefore many researches have been carried out, especially for building upper ontology for services [9]. However, constructing ontology is a fairly heavy engineering in semantic Web services, thus in this paper we will add semantic annotation to the Web service description through WordNet. It is mainly introduced in this section that how to add semantic annotations to user query information and WSDL descriptions with WordNet. Through semantic annotation sufficient preparations have been made for the follow-up matching work. As the technology of traditional Web service discovery is based on the matching foundation of key words, which means a low rate in exactness and completeness, thus we will propose a new matching Method in this section.

A. The Semantic Query Expansion

Only when users' needs have been understood correctly can we satisfy the demand of users, that is, service discovery system should accurately return Web services users have queried. The reason why the semantic query expansion and processing with WordNet is essential is that in query language there are several situations as follows. Firstly, there are many synonyms. It is quite common in natural language, such as E-business and E-commerce both mean e-commerce and so on. These are called synonyms in the WordNet. Secondly, there are the ambiguities of the concept. In many cases, the phenomenons of polysemy appear. Take the word "doctor" as an example, it may mean the person who is a licensed medical practitioner or Dr.. Thus, in order to eliminate the ambiguity, we should first clarify its specific implications. Thirdly, there are some words of superordinate concept and subordinate concept. In many cases, only through superordinate concepts or subordinate concepts can we retrieve the potential useful information, such as car and motorcar. [10]

In the semantic processing of user query, we concentrate on clarifying ambiguous concepts and the expansion of synonyms, while after semantic expansion with WordNet users' query intention can be better understood.

When the user enters the query information, we should determine whether the keyword in the query is a polysemy firstly. For example, as the word doctor has two meanings (Meaning 1：doctor, doc, physician, MD, Dr., medico -- a licensed medical practitioner; "I felt so bad I went to see my doctor"； Meaning 2：doctor, Dr. -- a person who holds Ph.D. degree (or the equivalent) from an academic institution; "she is a doctor of philosophy in physics"),so when a user enters doctor as keyword, we should determine the meaning queried according to the data in the user information database firstly, if a specialist input the word doctor, we can determine the practical significance of the word is meaning 1, because specialist is the subordinate concept of doctor, or specialist is one kind of doctor. Of course, when we cannot determine the exact meaning of a word with the background information in the user information database, several meanings of the word will be returned to the user and accurate choices should be made. Then we will go to the second step, which means to expand the input word to a synonym set. So if the word doctor has been clarified as meaning 1, these synonyms meanings "doctor, doc, physician, MD, Dr., Medico" will be delivered as a set to the Search and Matching Module.

B. The Semantic Annotation of Web Service Description

The purpose of semantic annotation is to retrieve the keywords of the WSDL service description with WordNet, which plays a vital role in implementing intelligent query of Web service. The technology of traditional documentation retrieval is based on the matching foundation of key words, which is short of description of semantic information. However, the same words often represent different meanings while different keywords may also be expressed the same in different contexts. Thus, the traditional way of retrieval leads to low rate in exactness and completeness. In the light of this issue, the paper draws the concept of semantic information into the research and pays more attention to distinguish Web services in the level of the concept, which is the same as superordinate concept and subordinate concept in WordNet. Through the semantic annotation of resources, it abstracts the true implication of the retrieval, and makes it possible that on the basis of semantic match, which improves the efficiency of retrieval.

Web service description language (WSDL) is a kind of XML language which is used to describe Web services as a set of network endpoints operating on the messages. In the model of this paper, the processing of WordNet based semantic annotation includes the scope determination of search keywords in the WSDL document, superordinate concept and subordinate concept annotation. In WSDL, major keywords are as follows: service name, port name, message name (input and output), operation name, and so on. Semantic annotation is the process of extracting the core vocabulary from the WSDL document and filling the corresponding superordinate concept and subordinate concept with WordNet. In general, we can use keywords as the document's semantic vector directly, but it is not concrete and substantial enough to establish semantic annotation of the WSDL document. So we should have the superordinate concept and subordinate concept as well as the concept itself in a semantic vector space.

Suppose D is the WSDL document set will be indexed, $d \in$ D, which is one document of the set; suppose C as a candidate concept set, and $c_i \in C$ is one candidate concept of the candidate concept set. And suppose that the superordinate concepts of c_i contained in WordNet are P_{i0}, $P_{i1} \cdots$, $P_{i(n-1)}$, n represents the total number of superordinate concepts of c_i. Therefore, the instance of candidate concept c_i in document d can be expressed as p_{i0}, p_{i1}, $\cdots$, $p_{i(n-1)}$, and then one of the semantic vectors of document d can be described by concept c_i and the corresponding superordinate concepts. In a word, one semantic vector can be expressed as (c_i, p_{i0}, $p_{i1} \cdots$, $p_{i(n-1)}$). Last but not least, WSDL documents with semantic annotations will be stored in the UDDI for retrieval according to the corresponding relation between WSDL and UDDI.

C. Mapping from WSDL to UDDI

A WSDL service consists of a set of abstract definition of operations and messages, a specific protocol bound to these operations and messages, and network endpoint specification. The UDDI provides a method of publishing and searching services description, the support of business and service information definition. Service description information defined in the WSDL is additional to the UDDI registry; many other different types of service descriptions are also available in UDDI. [11]

A complete WSDL service description includes a service interface and a service implementation document. Due to the reusable definition of services, the service interface was registered as a tModel in the UDDI. Service implementation describes instances of services, each of which is defined with WSDL service elements. These WSDL service elements in the service implementation document will be used to publish UDDI businessService. When publishing a WSDL service description, service implementation cannot be published as businessService until a service interface has be released as a tModel. The overview of mapping from WSDL to UDDI is illustrated in Fig.3.

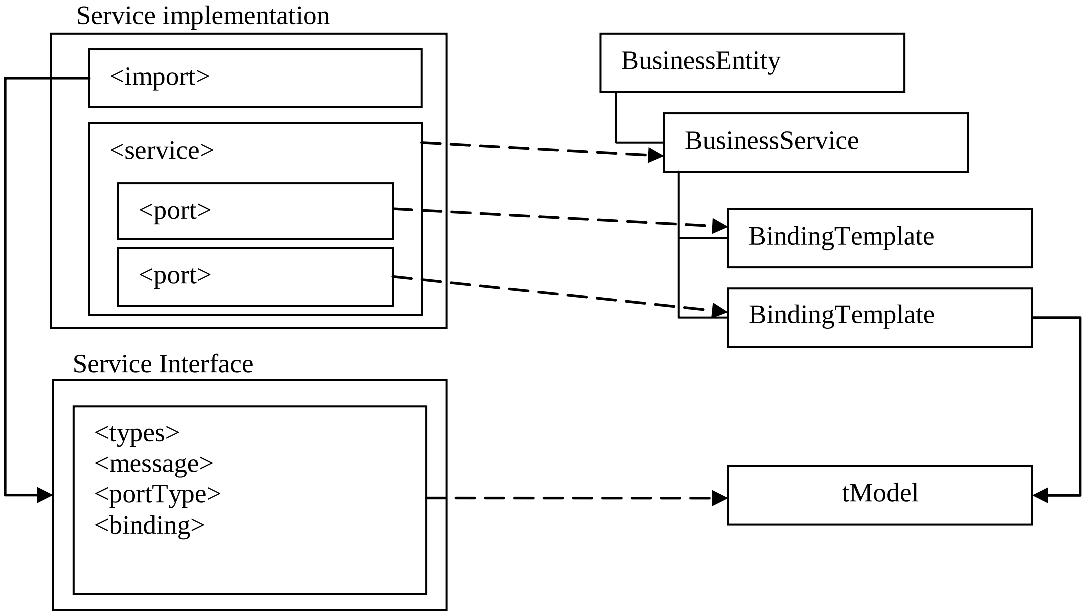

Figure 3 Overview of mapping from WSDL to UDDI

D. Semantic Matching Algorithm

There are two methods to convert the Web services into term vectors. One is manual method and the other is automatic method. Manual method means to convert the Web services into term vectors manually through making use of human knowledge. The automatic method is based on some natural language processing technology and some tools such as WordNet and UMLS.

According to the user's query and based on WordNet we can construct a semantic vector space with n dimension. The Multi steps of converting the Web service into term vectors are described as follows. [12]

(1) Compute the Comprehensive Frequency of Each Term in Every WSDL Document.

Let $q_i^p (i = 1, 2, ..., n)$ denote the comprehensive frequency of i^{th} term in p^{th} document and n is the total number of the terms.

In this step, we represent each Web services in a term vector of n dimension, expressed as $q^p = [q_1^p, q_2^p, ..., q_n^p]^T$.

As WSDL is an XML format for describing network services as a set of endpoints operating on messages containing either document-oriented or procedure-oriented information [13]. Usually the creator or the provider generates WSDL for Web services. Here we use WSDL to be a bridge to link the Web service and its semantic vector. Fig. 4 shows a WSDL specification metamodel as a UML class diagram [14].

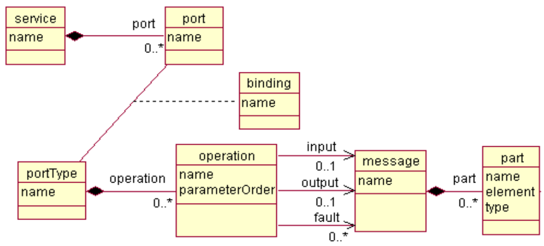

Figure 4 WSDL specification metamodel [14]

Based on WSDL document and the semantic vector space the domain experts have set up in the last step, we can compute the comprehensive frequency of each term in every WSDL. The terms appear in the service name, port name, the operation name, the message (input and output) name and the part name as shown in Fig.4 can reflect the semantic aspect of the services.

It is possible that one term appears in the different positions of the WSDL document will represent the Web service to some different degree. Thus when evaluating the important degree of one term in the WSDL document, we shouldn't just add up the frequency of the term in different position, but compute the comprehensive frequency of each term in the document based on the position weight.

In the semantic vector space the frequency of i^{th} term in the different positions the service name, port name, the operation name, the message (input and output) name and the part name(as shown in Fig.4) of p^{th} document can be represented respectively by $f_{i1}^p, f_{i2}^p, f_{i3}^p, f_{i4}^p$ and f_{i5}^p .Use b_1, b_2, b_3, b_4 and b_5 respectively represents the position weight of these areas. The weight coefficients are subjected to the following condition:

$$\sum_{j=1}^{5} b_j = 1 (0 < b_j < 1,\ j = 1,2,3,4,5) \qquad (1)$$

In this way to obtain the weight is also an evaluating process, so expert graded method can be adopted to compute the position weight. Then q_i^p can be computed by the following equation:

$$q_i^p = \sum_{j=1}^{5} b_j f_{ij}^{\,p} \,(i = 1,2,\dots n) \qquad (2)$$

Based on WSDL, if there are altogether d Web services the semantic content of every Web services can be represented by $q^p = [q_1^p, q_2^p, \dots, q_n^p]^T \ (p = 1,2,\dots,d)$.

The synonymies should be united and the different meaning of the term in the documents should be discriminated. Natural language processing technology can be used in this step. If the vocabulary and the knowledge warehouse are created, the computer can simulate human beings to determine the meaning of the term automatically. Some tools have been developed to process natural language, such as WordNet, UMLS and AWSC [15]. Here we use WordNet.

(2) Regress the Term Vector to [0, 1].

The inputs to a neural network must be regressed to be in a particular range, usually between 0 and 1. As $q_i^p \geq 1$, the term vector $q^p = [q_1^p, q_2^p, \dots, q_n^p]^T$ must be regressed to [0, 1], let

$$x_i^p = \frac{q_i^p - \min\limits_{1\le i\le n}\{q_i^p\}}{\max\limits_{1\le i\le n}\{q_i^p\} - \min\limits_{1\le i\le n}\{q_i^p\}} \qquad (3)$$

Obviously, $0 \leq x_i^p \leq 1$ come into existence. So each Web service can be represented by a term vector, such that:

$$X^p = [x_1^p, x_2^p, \dots, x_n^p] \ (p = 1,2,\dots,d).$$

If term i doesn't exist in WSDL document p, in another word, term i isn't important in WSDL document p, and then $x_i^p = 0$. If term i is very important in the document p, then $x_i^p = 1$. Values from 0 to 1 denote the importance increases. If the total number of the WSDL documents is d, d WSDL documents can be represented with a $d \times n$ vector matrix.

Suppose in a n dimension vector space, we have query's vector $X_q = [x_{q1}, x_{q2}, \cdots, x_{qn}]$ and document's vector $X_d = [x_{d1}, x_{d2}, \cdots, x_{dn}]^T$. The similarity between two vectors is generally measured as Euclidean distance between them, computing using the following equation:

$$SIM(X_q, X_d) = D = \|X_q - X_d\| = \sqrt{\sum_{i=1}^{n} (x_{qi} - x_{di})^2} \qquad (4)$$

Euclidean distance is the real distance between two points in n dimension, which is very usually adopted.

x_{qi} represents the weight of term k for query x_q and X_{di} represents the weight of term k for Web service X_d. Observe that $SIM(X_q, X_d) = 0$ means that X_d does not have any term that is in X_q, $SIM(X_q, X_d) = 1$, on the other hand means that X_d is most similar to x_q. Thus the value of the similarity from 0 to 1 indicates that the bigger the value is the more relevant the corresponding Web service is to the query demand.

Thus the relevancy-ranking algorithm of semantic algorithm can be implemented by measuring the similarity between the query and each candidate Web service using equation $SIM(X_q, X_d) = D = \|X_q - X_d\| = \sqrt{\sum_{i=1}^{n} (x_{qi} - x_{di})^2}$. Then the Web service can be ranked according to their similarity to the query.

IV. Example Analysis

In order to illustrate the process of system design and implementation, we will choose some hospital services as an example in this section.

A. User Query Information Expansion

In this paper, the model operated in .Net platforms is composed of three modules and the work flows have been given. Its procedure is written by Java, and various Java packages not only provide ready-made pieces of functions, but also make abstract interface for custom applications available.

When a user login this system, the User Interaction Module will return a query interface, and the interface diagram is shown in Figure 5.

Figure 5 User query interface

As shown above, users need to enter the following information: service name, port name, message name (input and output) information. Sometimes, more than one input and output message is necessary.

The user selecting interface mirrors in Fig.6.

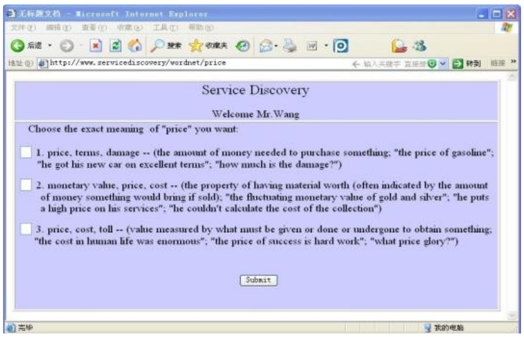

Figure 6 User selecting interface

When the user enters the query information, User Interface Module will determine whether the keyword in the query is a polysemy firstly. As is shown in Figure 5, the user enters the word "Price" in the place of Output. However, there are three different meanings for "Price" in the WordNet, which means we cannot define the user's exact intention. Therefore, User Interface Module will determine the exact meaning of a word with the background information in the user information database, otherwise several meanings of the word will be returned to the user and accurate choices should be made.

For example, when the second meaning is selected, User Interface Module will discard the other two meanings, at the same time, the second meaning "monetary value, price, cost" became the semantic query expansion of "Price", so with other key words. Then all the key words after processing will be delivered to Search Matching Module, without missing relevant information for matching.

Through semantic annotations, the query information can be represented as a five dimension vector space, which will be sent to Search Matching Module. Table 1 shows the vector space.

Table 1 Five dimension vector space after expansion

Service Name	**Port Name**	**Message Name**	**Input**	**Output**
OutpantinetCharge CheckService	OutpantinetCharge CheckPort	GetLastPrice	PatientID	Price, monetary value, cost

B. Semantic Annotation of Service Dispription

Take three of the Web services which are described with WSDL in the hospital as an example to illustrate the matching process. In order to simplify the process, these services will be described in the form of words in this section.

Service 1：Outpatient Charge Checking Service

ServiceName：OutpatinetChargeCheckingService

Port Name：OutpantinetChargeCheckingPort

Message Name：GetLastPrice

Input：PatientID

Output：Price

Service 2：Inpatient Charge Checking Service

Service Name：InpatientChargeCheckingService

Port Name：InpatientChargeCheckingPort

Message Name：GetLastPrice

Input：PatientID

Output：Cost

Service 3：Picture Checking Service

Service Name：PictureCheckingService

Port Name：PictureCheckingPort

Message Name：GetPicture

Input：PatientID

Output：Picture

Take Service 1 as an example, after semantic information extraction and annotation with WordNet:

Service Name：OutpatinetChargeCheckingService

Port Name：OutpantinetChargeCheckingPort

Message Name：GetLastPrice

Input：PatientID (badge)

Output：Price (value)

Then WSDL with semantic annotation will be mapped to UDDI according to section 3.

C. Matching Results

According to the matching algorithms, the query information can be represented as vector space, which will be useful for matching calculations of Service1, 2 and 3.The conclusion is shown in Figure 7:


Service Discovery

Welcome Mr.Wang

The result

Number	Service Name	Port Name	Message Name	Input	Output	Euclidean Distance
1	Outpantinet Charge Checking Service	Outpantinet Charge Checking Port	Get Last Price	PatientID	Price	0
2	Inpatient Charge Checking Service	Inpatient Charge Checking Port	Get Last Price	PatientID	Cost	1.414
3	Picture Checking Service	Picture Checking Port	Get Picture	PatientID	Picture	2

Figure 7 User query results

It is possible that one term appears in the different positions of the WSDL document will represent the Web service to some different degree. In the semantic vector space user query term in the different positions (Service Name, Port Name, Message Name, Input, and Output) of each service in UDDI can be represented respectively by following weight:

b_{11}=0.8, b_{12}=0.05, b_{13}=0.05, b_{14}=0.05, b_{15}=0.05; b_{21}=0.05, b_{22}=0.8, b_{23}=0.05, b_{24}=0.05, b_{25}=0.05; b_{31}=0.05, b_{32}=0.05, b_{33}=0.8, b_{34}=0.05, b_{35}=0.05; b_{41}=0.05, b_{42}=0.05, b_{43}=0.05, b_{44}=0.8, b_{45}=0.05; b_{51}=0.05, b_{52}=0.05, b_{53}=0.05, b_{54}=0.05, b_{55}=0.8.

Then can be computed by the equation (2):

q^1= (0.81, 0.81, 0.81, 0.81, 0.81) ^{T}

q^2= (02, 02, 0.82, 0.82, 0.82) ^{T}

q^3= (03, 03, 03, 0.83, 03) ^{T}

According to equation (3), regress the term vectors to [0, 1]:

X^1= (11, 11, 11, 11, 11) ^{T}

X^2= (02, 02, 12, 12, 12) ^{T}

X^3= (03, 03, 03, 13, 03) ^{T}

Calculate the Euclidean distance by the equation (4):

D^1=0

D^2=1.414

D^3=2

D. Analysis of the Results

The example above is simple that can not fully reflect all the semantic expansions and annotations in all situations, but it can basically validate the Web service discovery model based on semantic expanding. The paper draws the concept of semantic information into the research of Web service discovery, expands service description and improves the exactness. Due to the lack of a large number of samples, we cannot implement the quantitative analysis of the performance of service discovery. Therefore, the main work of the following phase is to collect samples of our services, conduct quantitative analysis on this model, and then validate discovery performance of the model.

V. Conclusion

The paper points to the lack of semantic information when use Web service inquiring, puts forward a model based on semantic expanding for Web service discovery, analyzes the function structure and the relationship between component parts of the model. Moreover, it adds semantic annotations to WSDL which is the Web service descriptions document with WordNet and maps it to UDDI exhaustively. Last but not least, the paper gives a matching algorithm on the semantic similarity calculation through semantic analysis when finding the match service, and improves the exactness. To some extent, it can realize the intelligence and semantics of Web service discovery. Further work should be done to conduct quantitative analysis based on a large size of Web service samples.

References

[1] Yinghui Huang,Guanyu Li. Semantic Web Service — — The Expansion of Web Services. Journal of Computer Applications, vol.28, 2008, pp.80-82.

[2] Xin Yang,Jianjing Shen. An Overview of Approaches to Semantic Web Service. Computer Applications and Software, vol.25, 2008, pp. 70-72.

[3] Jose Luis Ambite, Arogos. Dynamic Composition of Web Services for Goods Movement Analysis and Planning.

[4] Klein M, Bernstein A. Searching for Services on the Semantic Web using proeess ontologies. In Proe of the International Semantic Web Working Symposium, 2001, pp.159-172.

[5] Pengbo Xu.Research on P2P-based Semantic Web Service Discovery Model. Northwestern University, 2007. (In Chinese)

[6] Fengfeng Liang, Research on Distributed Semantic Web Service Discovery Based on Ontology. Beijing: Beijing University of Posts and Telecommunications, 2008. (In Chinese)

[7] Chunmei Li, Yunfeng Wu, Web Services Discovery Based on Ontology. Journal of Chuxiong Teachers College, 2006, pp. 10-17. (In Chinese)

[8] Huiying Gao, Qian Zhu. Semantic Web based Multi-agent Model for the Web Service Retrieval [J]. Computer Network and Multimedia Technology (CNMT2009), 2009, pp.897-900.

[9] Hongxing Liu.Research on Web Services Discovery and Selection Based on Agent. Dalian: Dalian Maritime University, 2006. (In Chinese)

[10] Zili Zhou.Research on Key Technologies of Ontology Construction Based on WordNet and Its Application in Security Domain. East China Normal University, 2009.

[11] W.H. Liu, J. Zhao.Ontology-based semantic matching method. Micro-Computer Application, vol.24, 2008, pp.5-8.

[12] Huiying Gao, Wolffried Stucky, Lei Liu. Web Services Classification Based on Intelligent Clustering Techniques. International Forum on Information Technology and Applications, 2009.5

[13] M.Brain Blake, Michael F.Nowlan," Taming Web services from the wild," Internent computing, Vol.12, 2008, pp.63-64.

[14] W3C. Web services description language (WSDL)
1.1 .http://www.w3.org/tr/wsdl,2001.3.

[15] C.Marco, Z.Alejandro, and C.Marco, "An approach to Web service classification based on machine learning techniques," Intelligencia artificial, Vol.12, 2008, pp.27-29.