11-28-2011

A Merging Cluster Algorithm for QoS-Oriented Supply and Demand

Xiuzhen Feng
Beijing University of Technology, xfeng@bjut.edu.cn

Gaofeng Wua
AVIC Information Technology Corporation Ltd., wgaofeng@163.com

芽|Sprouts Working Papers on Information Systems ISSN 1535-6078

A Merging Cluster Algorithm for QoS-Oriented Supply and Demand

Xiuzhen Feng
Beijing University of Technology, China

Gaofeng Wua
AVIC Information Technology Corporation Ltd., China

Abstract
Discovering service-on-demand for large numbers of functionality-similar web services is one of the key issues in services discovery study. According to the functionality-similar web services, a service-on-demand discovery process is proposed in this paper. To meet the target, FCM clustering is adopted for agglomerative clustering between the user's QoS demand information and the QoS information of Web Service resource. Then, the sequence could be determined by similarity computation in the same classification clustering. Lastly, the numerical example is presented to illustrate that the service-on-demand can be discovered efficiently to optimize network resources and improve the efficiency.

Keywords: service-on-demand, FCM cluster ,sequence, QoS.

Permanent URL: http://sprouts.aisnet.org/11-166

Reference: Feng, X., Wua, G.-F. (2011). "A Merging Cluster Algorithm for QoS-Oriented Supply and Demand," Proceedings > Proceedings of SIGSVC Workshop . *Sprouts: Working Papers on Information Systems*, 11(166). http://sprouts.aisnet.org/11-166

INTRODUCTION

Web service is a fast growing distributed computing model, which covers how execution programs can achieve interoperability on the web through a set of general standards on how the programs can be described, published, searched and called on. Leveraging the advantages of distributed computing, Grid computing, XML (Extensible Markup Language) and other related technologies, the Web service has a high degree of interoperability as well as being highly cross-platform compatible and loosely coupled, driven by its use of the WSDL (Web Services Description Language), OWL (Web Ontology Language), UDDI (Universal Description, Discovery and Integration) and SOAP (Simple Object Access Protocol) which are all XML-based. To date, Web service has become an effective mechanism for cross platform data and application integration on the internet. With the ever growing development and application of the Web Service in software structuring, to maximize the convenience of the customer execution programs based on browser applications, the web services are increasingly published by companies on the network, and discovered by users. It demonstrates that web service has become an effective solution to provide services.

With the continued development of web service applications, the information resources on the Internet have been getting abundant. In this case, users can be highly confused by many functionality-similar web services. One of the hot research topics is how to match the users' demands with these functionality-similar web services efficiently and effectively. Web service discovery analysis under the framework of UDDI is one of common approaches, but in this case, the user has to bear the work of determining and selecting web service according to their demands. This approach also leads to a less effective link-up between users' customized demand and service supplier's bespoke solutions. How to choose a service that meets their specific demands would become a key challenge for service users. In the web service architecture, QoS (Quality of Service, QoS) is used to describe the features and functionality of web service. Therefore, it can be used to identify and select the appropriate service. In other words, merging the QoS information of users' demands with that of service provided, the customized web service that meet the users' demands can be discovered by a clustering process.

The attributes of web service can be divided into two groups, e.g. functionality related versus non-functionality related. The functionality attributes describe the operations that a Web Service provides, and the non-functionality ones describe the characteristics of QoS which the Web Service provides. In the former case, the service specification can be designed in service registries for storing, managing and disclosing for service providers (Ferreira et al. 2010). The design and development of a service registry can be achieved through a prototype that is built on a "proof-of-concept" basis. The qualitative method, however, is difficult to measure the quality of service, such as the matching precision, and the availability of the service. In a quantitative method, the QoS attributes of service can be measured by software, such as HTTP watch. The existing keywords query method (Lingjuan et al. 2008) which is syntax based can find out the Web services with similar functionality in UDDI register center, but users still need to select the proper service from many functionality-similar service sets using static or dynamic method (Pathak et al. 2006; Bianchinia et al. 2006). In fact, users often have vague views on their demands so leading to relatively inaccurate selection process. Since the demands of users can be described by the QoS attributes of Web Service, it has become an important basis on which users could select service-on-demand (Changying et al. 2009). In this paper, we take an interval based approach to describe QoS attributes, and developed a service on demand discovery process based on the sequencing after clustering. In other words, Cluster analysis is performed between the QoS demand of users and the QoS of supplied Web Service resources. Then, the sequence could be determined by similarity computation within the classified clusters. Step by step, the un-matched service will be rejected during the process with the suitable, customized service based on demand information being filtered out.

THE ANALYSIS OF SERVICE DISCOVERY PROCESS

The web service cluster includes the QoS information of users' demand and the QoS information of web service resources. The web service resource information needs be acquired from the WSDL register information (Bin. 2006) of specific Web Service, and then form a resource database through a series of ETL (Extraction-Transformation-Loading) processes (Dezhi et al. 2006). This is the mechanism of web matching to build up the functionality-similar service pool. Based on this, the QoS demand of users and the QoS supply of Web Service resources in the same service fields will be clustered, and then being sequenced in the service cluster. Afterwards the decision on whether to negotiate or push the result to the user can be made based on sequencing results of QoS demand information. The service-on-demand discovering process can be illustrated in Figure 1.

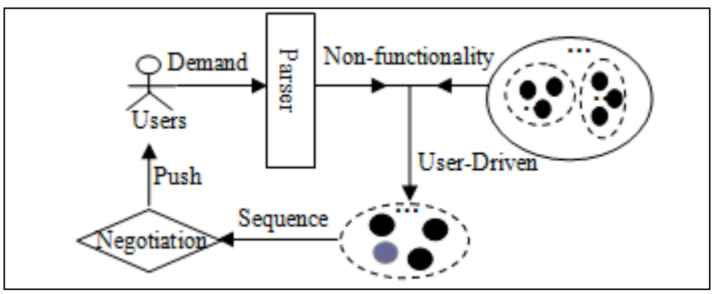

Figure 1. The service-on-demand discovering process

THE FCM BASED INTERVAL MERGING CLUSTER ALGORITHM

In the process of service discovery, due to the dynamics and uncertainty of network, and the vagueness and variety of users' demand, it is difficult to present the QoS information with exact numerical values between supply side and demand side, but using the interval values can be much more efficient. To address the intrinsic fuzziness of the QoS information, we adopted the Fuzzy C-Means (FCM) algorithm for user demand based service discovery. FCM is one type of cluster algorithms which are based on membership degree to determine whether the sample data belongs to a particular classification. The aim is to group the most similar objects under the same category, but the objects with low similarity into different categories (Lianghui et al. 2006). The algorithm was developed by Dunn, and is mainly used to solve the issues in information model recognition of exact value and fuzzy classification. The cluster study (Chunhai et al. 2004) based on interval value further contributed to the expansion of the application of this algorithm. It divides n sample vectors into m ambiguous / fuzzy groups. Through repeated modification of the clustering center and classification matrix, the clustering center of each categories and membership degree of each sample can be obtained and, hence, the division of fuzzy clustering can be realized.

Based on the service pool built (Xuanzhe et al. 2007), the FCM is adopted to study the merging cluster algorithm for QoS-oriented Supply and Demand. The result is that the web services with similar QoS information are grouped into the same category, while others could be formed into a second category. Under the rules of sequencing after clustering, a service in a category is deemed more suitable if it is ranked closer to the top. If there is at least one service in a category provides better service than that is required by the demand side, and then this service is deemed to be most suitable. Otherwise, the service can not be found in the service pool, and the negotiation algorithm (Xiuzhen et al. 2010) should be applied for optimization. The process of merging cluster based on FCM is shown in Figure 2.

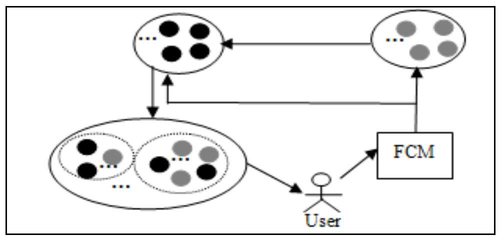

Figure 2. The merging cluster based on FCM

The merging cluster as multi-attribute decision-making problem where attributes are expressed by interval numbers can be described as: $\mathrm{Q}=\{\mathrm{Q}_1,\mathrm{Q}_2,\cdots \mathrm{Q}_\mathrm{p}\}$, $X=\{X_1, X_2, \cdots, X_{n/2}, \cdots, X_n\}$ is known. Q is the QoS information, and X is the sample of Web Service, where $\{X_1, X_2, \cdots, X_{n/2}\}$ is the resources samples in service pool, but $\{X_{n/2}, \cdots, X_n\}$ is the demand samples for users. $x_{ij}=[x_{ij}^-, x_{ij}^+]$ presents the interval value of Q_j for X_i, which $i \in 1...n$, $j \in 1...p$. In this paper, a cluster algorithm for QoS-Oriented supply and demand is characterized by the following steps:

Step 1: Quantification and Normalization. The QoS attributes of web services can be divided into qualitative and quantitative ones. For the qualitative attributes, the numerical value can be normalized into [0, 1] by means of expert investigation method and subjective experience; and for the quantitative attributes, statistical methods can be used to normalize the data (Jianghua et al. 2008). Among the QoS attributes, some are benefit-oriented index larger values correspond to better general performance, while some are cost-oriented where smaller values correspond to better general performance and these can be normalized with ‘weighting changing method’. For the convenience of calculation and analysis, the normalized processes of variables are kept in the interval [0, 1].

When the Q_i is the benefit-oriented index:

$$\begin{cases} x_{ij}^{'-} = x_{ij}^- / \sum_{i=1}^{n} x_{ij}^+ \\ x_{ij}^{'+} = x_{ij}^+ / \sum_{i=1}^{n} x_{ij}^- \end{cases} \quad i \in 1...n, \quad j \in 1...p \tag{1}$$

When the Q_i is the cost-oriented index:

$$\begin{cases} x_{ij}^{'-} = \dfrac{1}{x_{ij}^{+}} / \sum_{i=1}^{n} \dfrac{1}{x_{ij}^{-}} \\ x_{ij}^{'+} = \dfrac{1}{x_{ij}^{-}} / \sum_{i=1}^{n} \dfrac{1}{x_{ij}^{+}} \end{cases} \quad i \in 1...n, \quad j \in 1...p \tag{2}$$

and, $x_i^{'} = \{x_{i1}^{'}, x_{i2}^{'}, \cdots, x_{ip}^{'}\}$, $x_{ij}^{'} = [x_{ij}^{'-}, x_{ij}^{'+}]$

Step 2: Initialize Cluster Standard Set. For the pattern samples: $X = \{X_1, X_2, \cdots, X_n\}$, given the total number of their classes, c, and the fuzzy weighting factor, m (>0), and the iteration accuracy factor, ε (>0). We set the initialization cluster standard prototype $V^{(0)} = (v_1^{(0)}, v_2^{(0)}, \cdots, v_c^{(0)})^T$, which $v_k^{(0)} = (v_{k1}^{(0)}, v_{k2}^{(0)}, \cdots, v_{kp}^{(0)})^T$, and $v_{kj}^{(0)} = [v_{ki}^{(0)-}, v_{ki}^{(0)+}]$, setting the iteration number b=0.

Step3: Calculating the Membership Degree matrix $U^{(0)} = (\mu_{ki}^{(0)})_{c \times n}$.

□ If $\forall k, v_k^{(h)} \neq x_i^{'}$.The theorem below holds:

$$\mu_{ki}^{(0)} = 1 / \sum_{t=1}^{c} \left[\frac{D_{ki}^{(0)}}{D_{kt}^{(0)}} \right]^{2/(m-1)} \quad i \in 1...n, \quad k \in 1...c \tag{3}$$

Let $D_{ki}^{(0)} = \| v_k^{(0)} - x_i^{'} \|$ is the euclidean distance from $v_i^{(0)}$ to $x_i^{'}$.

□ If $\exists l \ (1 \le l \le c)$, which makes $v_k^{(0)} = x_i^{'}$. When $k = 1$, let $\mu_{ki}^{(0)} = 1$; when $k \neq 1$, let $\mu_{ki}^{(0)} = 0$.

Step4: calculating the latest cluster standard sets $V^{(b+1)} = (v_1^{(b+1)}, v_2^{(b+1)}, \cdots, v_c^{(b+1)})^T$.

$$v_k^{(b+1)-} = (\sum_{i=1}^{n} (\mu_{ki}^{(h)})^m x_i^{'-}) / \sum_{t=1}^{n} (\mu_{ki}^{(h)})^m \tag{4}$$

$$v_k^{(b+1)+} = (\sum_{i=1}^{n} (\mu_{ki}^{(h)})^m x_i^{'+}) / \sum_{t=1}^{n} (\mu_{ki}^{(h)})^m \tag{5}$$

and, $k = 1, 2, \cdots, c$, $i \in 1...n$, $j \in 1...p$, $x_i^{'-} = (x_{i1}^{'-}, x_{i2}^{'-}, \cdots, x_{ip}^{'-})^T$, $x_i^{'+} = (x_{i1}^{'+}, x_{i2}^{'+}, \cdots, x_{ip}^{'+})^T$,

$v_k^{(b+1)-} = (v_{k1}^{(b+1)-}, v_{k2}^{(b+1)-}, \cdots, v_{kp}^{(b+1)-})^T$, $v_k^{(b+1)+} = (v_{k1}^{(b+1)+}, v_{k2}^{(b+1)+}, \cdots, v_{kp}^{(b+1)+})^T$

Step5: Calculating $U^{(b+1)}$, and let

$$\mu_{ki}^{(b+1)} = 1 / \sum_{t=1}^{c} \left[\frac{D_{ki}^{(b+1)}}{D_{kt}^{(b+1)}} \right]^{2/(m-1)} \quad i \in 1...n, \quad k \in 1...c \tag{6}$$

Step6: Calculating clustering precision $J(U^{(b)}, V^{(b)})$ and $J(U^{(b+1)}, V^{(b+1)})$

$$J(U,V) = \sum_{k=1}^{c} \sum_{i=1}^{n} (\mu_{ki}^{(h)})^m \| v_k^{(h)} - x_i^{'} \|^2 \quad i \in 1...n, \quad k \in 1...c \tag{7}$$

Step7: Calculating convergence precision $\varepsilon_1 = J(U^{(b)}, V^{(b)}) - J(U^{(b+1)}, V^{(b+1)})$

When $\varepsilon_1 < \varepsilon$, the iteration will be stopped, and an optimized solution is achieved. In this case, $U^{(b+1)}$ is the optimal membership degree matrix, and the $V^{(b+1)}$ is the optimal cluster standard. Otherwise, repeat step 3-7.

For QoS, the data should be standardized into the [0, 1] range (step 1); an initial cluster set is regulated at first (step 2). By these, the membership degree matrix can be calculated (step 3). Comparing with $J(U^{(b)},V^{(b)})$ and $J(U^{(b+1)},V^{(b+1)})$ (step 6) by calculating $\mu_{ki}^{(b+1)}$ (step 5) and $V^{(b+1)}$ (step 4), a decision on whether the iteration should be continued can be determined with a comparison to standard value ε (step 7).

PRIORITY METHOD BASED ON INTERVAL SIMILARITY

There are three types of general relations among interval numbers: separation, intersection and inclusion. The separation is that the intersection region is not formed between intervals; the intersection is that a public region can be identified across intervals; the inclusion is that one of interval is contained by another interval. In this paper, the interval similarity (e.g. how one interval overlaps with another) can be used to evaluate the degree of coincidence among the intervals.

The hypothesis $a=[\underline{a},\overline{a}]$ is the QoS demand information, and $b=[\underline{b},\overline{b}]$ is the resources information, and $\underline{a},\overline{a},\underline{b},\overline{b}\in R$. The interval similarity can be formed as following:

$$sim(a,b)=\frac{L(a\cap b)}{(L(a)+L(b)-L(a\cap b))} \tag{8}$$

Where L is the interval length, $a\cap b$ presents the overlapping ranges from *a* to *b*. simultaneously, the metric can be defined and shown in Table 1.

Relations	Expression	Similarity
Separation	$\overline{b}\leqslant\underline{a}$	0
	$\overline{a}\leqslant\underline{b}$	1
Intersection	$\underline{b}<\underline{a}\leqslant\overline{b}<\overline{a}$	$(\overline{b}-\underline{a})/(\overline{a}-\underline{b})$
	$\underline{a}<\underline{b}\leqslant\overline{a}<\overline{b}$	$(\overline{a}-\underline{b})/(\overline{b}-\underline{a})$
Inclusion	$\underline{b}\leqslant\underline{a}<\overline{a}\leqslant\overline{b}$	1
	$\underline{a}\leqslant\underline{b}<\overline{b}\leqslant\overline{a}$	$(\overline{b}-\underline{b})/(\overline{a}-\underline{a})$

Table 1. Interval similarity metric

For the multi-attribute information, the similarity summary must be calculated to determine the priority in service pool.

$$sum=\sum_{i}^{n}\sum_{j}^{p} sim(x_{ij}^{'},y_{tj}^{'})\quad i\in 1\ldots n\,,\quad j\in 1\ldots p \tag{9}$$

Here: $sum=p$ means the demand has been satisfied completely, and the optimal service exists; if $0<sum<p$, the result represents a partial match of demand and it is necessary to call negotiation function to negotiate with un-matched metrics to meet the demand; if $sum=0$, the demands cannot be met, and the negotiation would be costly.

EXAMPLE

Xmethods introduction

The Xmethods is a web service proxy center, where users can browse the list of web services, access to specific web services, and understand the implementation details. Similar to UDDI, web service proxy center allows that service providers publish address and information of available web services; search based on their demand criteria; call the web service with specific functionality. Xmethods maintains a free list of web services, and services links with additional information on providers. The list contains information such as service name, style, description, and implement, covering as many as 400 registered services there. The specific service can be searched by users on a browser interface. And web service can be called through a series of applications. In addition, the site also provides the description on implementation of the web services, WSDL verification tools and guidance document to install and use a particular service. However, Xmethods does not use the same classification as that under UDDI, but uses the directory-based classification, where services can be searched by keywords method.

Weather Forecast Service Cluster based on FCM

The aim of this numerical example is to verify the feasibility of the service discovery algorithm we proposed. The metrics chosen includes qualitative, quantitative, benefit-oriented and cost-oriented metrics of QoS attributes. The interval value is used to present the data. By using the keywords method, a list of web services can be found matching users' demand. Taking 'weather forecast service' as an example, we can get 15 results. So the cluster sample can be shown in table 2, which includes 15 service resources and one service demand ('Weather forecast service').

NO.	Cost	Availability	Usability	NO.	Cost	Availability	Usability
S01	[18,20]	[5,6]	[0.6,0.8]	S09	[14,16]	[4,5]	[0.3, 0.3]
S02	[7,8]	[3,3]	[0.2, 0.4]	S10	[5,8]	[5,5]	[0.3, 0.4]
S03	[7,7]	[4,5]	[0.3, 0.5]	S11	[16,18]	[7,8]	[0.5, 0.5]
S04	[6,7]	[4,5]	[0.4, 0.4]	S12	[8,10]	[4,4]	[0.4, 0.5]
S05	[19,21]	[6,7]	[0.8, 0.9]	S13	[12,14]	[3,3]	[0.3, 0.4]
S06	[14,14]	[4,5]	[0.4, 0.6]	S14	[21,21]	[7,8]	[0.5, 0.6]
S07	[19,19]	[8,8]	[0.7, 0.8]	S15	[6,7]	[2,3]	[0.2, 0.3]
S08	[17,18]	[6,7]	[0.5, 0.6]	R00	[14,15]	[3,4]	[0.4, 0.4]

Table 2. Cluster object samples

We leverage a method developed by Xiuzhen, F., Kunliang, T. (2010) to get the relevant information, covering both qualitative and quantitative QoS metrics. For quantitative index, the dataset can be developed by using Httpwatch tools at the client terminal end to analyze service providers. For qualitative index, the data can be obtained by analysis on results of distributed questionnaires. There are 3 indices, where cost (the expenses that a user incurs to call the service) is the cost-oriented, availability (the quality aspect of whether the Web service is existent or ready for immediate use) and usability (measuring the capability of a service to be effectively understood, learned and used by a user.) is the benefit-oriented. Usability is the qualitative index, so the scale is 0.1(worst) to 1.0 (best). Due to space limitations, the normalized data have been left out in this paper. Assuming $c = 3$, m=2, $\varepsilon = 0.0005$, the initialization cluster standard is then shown in Table 3.

$v_1^{(0)}$	[0.025,0.053]	[0.055,0.136]	[0.061,0.121]
$v_2^{(0)}$	[0.042,0.064]	[0.034,0.082]	[0.015,0.082]
$v_3^{(0)}$	[0.058,0.110]	[0.021,0.074]	[0.036,0.071]

Table 3. Initialization cluster standard $V^{(0)}$

According to the clustering process, after 3 iterations of calculation, we got the result as $\varepsilon_3 = J(U^{(2)}, V^{(2)}) - J(U^{(3)}, V^{(3)}) = 0.000231 < \varepsilon$, and the optimal cluster standard $V^{(3)}$ (Table 4), and the optimal membership degree $U^{(3)}$ (Table 5). Per the maximum membership degree principle, the services are divided into 3 categories, which are "Good", "medium", and "poor" degrees. So the results clustered are $\{S01, S05, S07, S08, S14\}$, $\{S06, S09, S11, S13, R00\}$ and $\{S02, S03, S04, S10, S12, S15\}$.

$v_1^{(3)}$	[0.033,0.040]	[0.076,0.097]	[0.072,0.104]
$v_2^{(3)}$	[0.044,0.057]	[0.042,0.058]	[0.043,0.063]
$v_3^{(3)}$	[0.084,0.117]	[0.042,0.056]	[0.035,0.060]

Table 4. Optimal Cluster Standard $V^{(3)}$

To further select the optimal service, range similarity based services sorting method is used to select the service that matches the users' demand most closely. Obviously, $R00$ is located in the second group. According to Eq. (9) and Table 1, the similarity between $R00$ and ($S06, S09, S11, S13$) are 3, 1.5, 2 and 1. The sequence is therefore $S06 > S11 > S09 > S13$. In this case, only $S06$ is the optimal service after comparing the similarity with others. Accordingly, no further negotiation process is needed.

	S01	**S02**	**S03**	**S04**	**S05**	**S06**	**S07**	**S08**
$\mu_1^{(3)}$	0.813	0.033	0.035	0.022	0.839	0.564	0.894	0.800
$\mu_2^{(3)}$	0.075	0.199	0.185	0.147	0.092	0.834	0.070	0.036
$\mu_3^{(3)}$	0.042	0.867	0.880	0.931	0.045	0.052	0.027	0.092
	S09	**S10**	**S11**	**S12**	**S13**	**S14**	**S15**	**R00**
$\mu_1^{(3)}$	0.534	0.057	0.597	0.092	0.538	0.834	0.040	0.558
$\mu_2^{(3)}$	0.816	0.106	0.857	0.372	0.880	0.089	0.199	0.879
$\mu_3^{(3)}$	0.030	0.838	0.077	0.536	0.042	0.034	0.861	0.063

Table 5. Optimal Member Degree $U^{(3)}$

CONCLUSIONS

A large number of functionality-similar web services represent tough challenges for users to discover the service-on-demand. Although users could search the Web Service in UDDI with traditional methods, it is inefficient, especially in finding solutions for customized demand. A QoS based merging cluster algorithm has been presented in this paper to solve the service discovery issue. In this process, the FCM is used to cluster the QoS demand and QoS supply information. Then, the sequence of objects can be determined to evaluate the Web Service and whether or not further negotiation is needed. To address the issue of many functionality-similar web services provided by multiple service providers, the method of sequence-after-cluster is adopted to optimize the service discovery research. Among functionality-similar web services, FCM clustering is adopted for agglomerative clustering of the user's QoS demand information and the QoS information of Web Service resource. Thereafter, the sequence could be determined by similarity computation within the same classification clustering. The decision of further negotiation can be made according to sequencing results to find out the service closest to the demand of users (or lack of it). This method can simplify the service discovery process and maximize the likelihood of identifying the services that meet customers' demands.

The QoS based research of supply and demand service discovery process uses the dynamic network environment as the medium, focuses on studying the non-functional properties of Web services, and represents the web services performance in the form of the interval values. On the basis of matching the function attributes, it optimizes the network service resources to achieve "on-demand, customized" solutions to meet specific demand by users and to enhance the efficiency of Web service discovery process.

Opportunities of future work on this topic exist and are derived from the limitations of the work presented in this paper. Specifically, they include:

- The evaluation on index; the performance of service is measured by index, so it is necessary to choose suitable index to determine whether a service is good or poor. Traditionally, the index can be categorized into qualitative and quantitative, and cost-oriented and benefit-oriented ones. However it is difficult to determine which index is more important in measuring the performance, and whether or not a common set of indices can be used to measure a wide range of services. In this type of research, the choice of index is the basis to derive the conclusion. If the index cannot be effectively determined, the validity of research results may be undermined.
- The evaluation method; as we know, different evaluation methods lead to different results. With developing of web service technology, new evaluation methods that better fit with human's thought process are needed to improve the efficiency and effectiveness of service discovery.
- The evaluation technology; the efficiency of algorithm is closely linked to the evaluation technology. In this paper, we proposed an idea to match the users' demand information to that of resources' supply by a merging cluster algorithm and subsequent sequencing process. However, the efficiency of the algorithm is not the focus of this paper and may warrant further research to optimize the calculation performance.

REFERENCE

Ferreira, P., and Van, O. (2010) "Service Registry Design: An Information Service Approach", *International Journal of Information System in the Service Sector*2(4),pp 1-21

Lingjuan, H., and Lianchen L.(2008) "A Modified Operation Similarity Measure Method Based on WSDL Description", *Chinese Journal of Computers*31(8), pp 1331-1339

Pathak, J., Koil, N., and Caragea, D. (2006) "A framework for Semantic Web Services Discover, " in *proceedings of the 7th* ACM *International Workshop on Web Information and Data Management.* ACM Press

Bianchinia, D., and Antonellis, V. D. (2006) "Ontology based methodology for e-service discovery, " *Information System* (31). pp 361-380

Changing, D., Mingkun, D., and Huijuan Z. (2009) "Expanded Augment UDDI system to support QoS of web service discovery model", *Computer Engineering and Design* 30(2), pp 358-361

Bin, X. (2006) "Web Services Search Method Based on Domain", *Computer Engineering* 32(20), pp 33-35

Dezhi, X., Chunhui, Z. (2006) "Passi K Concept semantic similarity research based on SUMO ", Journal *of Computer Applications* 26(1), pp 180-183

Lianghui, H., Changqing, C. Ting, Z. (2006) "A Combinational Model of Evaluation Results Based on the Idea of Fuzzy Clustering", *in the proceedings of 2006 Chinese Control and Decision Conference*, pp 1251-1255

Chunhai, Y., Zhiping. F. (2004) "A FCM clustering algorithm for multiple attribute information with interval numbers", *Journal of Systems Engineering* 19(4), pp 387-393

Xuanzhe, L., Zheng, H. (2007) "Consumer-Centric Service Aggregation: Method and Its Supporting Framework", *Journal of Software* 18(8), pp 1883-189.

Xiuzhen, F., Gaofeng, W. (2010) "A Study on Interval Negotiation Algorithm for Service-oriented QoS", in *proceedings of the International Conference on Computational Intelligence and Software Engineering*, pp 1-4

Jianghua, L., Shuting, S. (2008) "Research on Approaches of Acquiring and Processing QoS Information of Web Services", *Journal of Jiangxi University of Science and Technology* 29(06), pp 24-28

Xiuzhen, F., Kunliang, T. (2010) "A user preference-oriented model for selecting information resource service", *Information Science* 28(9), pp 1397-1408

芽|Sprouts
Working Papers on Information Systems | ISSN 1535-6078

Office:
Sprouts
University of Amsterdam
Roetersstraat 11, Room E 2.74
1018 WB Amsterdam, Netherlands
Email: admin@sprouts.aisnet.org