A NOVEL FRAMEWORK FOR DYAMIC BUSINESS SEMANTIC WEB SERVICE CLUSTERING

Manish Kumar Mehrotra[1], Suvendu Kanungo[2]

[1]*Department of Computer Sc. & Engg., Birla Institute of Technology,*
Mesra, Allahabad Campus, (India)

[2]*Department of Computer Sc. & Engg., Birla Institute of Technology,*
*Mesra, Allahabad Campus, (*India)

ABSTRACT

Now a days, There is growing demand for well Structured repositories for web services in order to facilitate efficient discovery of semantic web services in the area of service oriented Computing. In this paper we have proposed a clustering based novel approach for inducing improvement in the searching mechanism of semantic web service. In view of categorization of services, we have employed I/O matrix for both advertised and enquiry service using ontology for I/O Parameters. The Efficacy of our proposed approach is based on real data sets which is available in the data world.com and is evident with higher precision and recall rate of clustered services in comparison with other similar approaches.

Keywords: Business semantic, SOA, Clique, advertised services, Inquired services, clustering

I. INTRODUCTION

Web service technology bound to be effective in their available capacity which uses their dynamic discovery and compositions of advertised Web services. A Web service is an interface, which describes a collection of comprehensive programs which has other links of sub programs or linked programs that are used by network to access through standardized operation. At present, the business architecture of web services are based on the interactions between three entity i.e. service provider, service registry and service requester. The interactions among them involve publish, find and bind operations. Clustering Mechanism based on business semantic is one of the core area in service computing on which several research have been performed .In order to perform discovery of web services, researchers generally focus only on the advertised services. However, they have not considered for similar importance for both advertised services and inquiry services.

In service oriented Architecture(SOA), published services are stored in repositories called UDDI, and searching process essentially based on either keyword oriented or browning oriented. However, the process is time consuming due to inefficient service categorization. In this paper, we have proposed clustering based framework for efficient service categorization by simplifying semantic of input/output matrix and providing equal importance to both advertised and inquiry services. We primarily focus on efficient service clustering mechanism that facilitate discovery of semantically relevant web services. The remainder of the paper is organized as follows. In section 2, we throw light on different discovery process for web services. In section 3, we discuss our proposed framework for service clustering. Then, we examine the performance of our proposed algorithm in section 4. Finally ,section 5 concludes the paper and section 6 provide discussion on future research direction

II. RELATED WORK

Discovering services in [3] is used novel method to calculate the fuzziness of similar degree of the required service and the advertised service. Technique used in [4] is to calculate the maximal similar degree and the mean similar degree. Comparison have been performed using calculative method in [5] is to know the profile of the required service with that of the advertised service. There are also some services discovery methods which are based on the organizing advertised services.

Classification of service domain is used in [6] to locate the right domain and then search service one by one in that is available in selected domain. But how to classify services into. different domains is not mentioned. Ontology method used in [8], to describe semantic of services, and a class's graph which represents ontology is used to organizing advertised services. Searching the required service is by utilizing the inherit mechanism of classes. How to form the class's graph is not mentioned in the article.

In this paper, we improve the performance of Web service clustering by introducing a novel frame work of advertised and inquired service with predefined presumption . We have compared our algorithm with tag and non tag clustering algorithm of other clustered services.algorithms.

III.PROPOSED CLUSTERING FRAMEWORK

We assume that there are four main sub services of sales and distribution service.

W1: Inquiry :<Company{name, registered address, validity of registration}>,< Product {Availability, Type, Model}>,<Price{basic, tax. discount , other privilege} <Delivery{ date , Address, Product}>

W2: Order:<Product {Availability, Type , Model}> ,<Price {Basic, Tax. Discount , Other privilege},Location{ address of client, distance,}>

W3 : Delivery<{Address of client, Quantity, Type of product, ID}>

W4: Billing <Product, Price, Client>

Above service are advertised on UDDI cluster from where this have share the some attribute with other services

Presumption : Software oriented architecture is one basic framework on which modified framework is continued developing with new assumption. As we know that advertised and inquiry services follow the same semantic to search the new cluster of service for business strategies. Here we make assumption that semantic of advertised and inquiry follow the same expression of semantic. Here we consider that www standard follow the every advertised and inquired services. If pattern of advertised service has to change with new standards then inquiry services follows the same pattern of changes. Role of agent service has been considered as mute.

Here Wi1,Wi2…WiN are one the advertised services with constant nature during operation of cluster e . Number of companies is constant during process of clustering.

3.1. Advertised Service Matrix

Table 1: Advertised Services

COMPANY/SERVICE	(W1) SERVICE *i* *(Representing in probabilistic manner) with reference to column vector)*	(W2)SERVICE *o* *(Representing in probabilistic manner) with reference to column vector)*	(W3)SERVICE *d* *(Representing in probabilistic manner) with reference to column vector)*	(W4)SERVICE *b* *(Representing in probabilistic manner) with reference to column vector)*	Number of service
A	$TI1 = \frac{Ni1of\ Wi1}{Ni1 + Ni2 + \cdots N}$	$TO1 = \frac{No1of\ Wo1}{Ni1 + Ni2 + \cdots N}$	$TD1 = \frac{Nd1of\ Wd1}{Ni1 + Ni2 + \cdots NiN}$	$TB1 = \frac{Nb1of\ W}{Ni1 + Ni2 +}$	M1
B	$TI2 = \frac{Ni2of\ Wi2}{Ni1 + Ni2 + \cdots N}$	$TO2 = \frac{No2of\ Wo2}{Ni1 + Ni2 + \cdots N}$	$TD2 = \frac{Nd2of\ Wd2}{Ni1 + Ni2 + \cdots NiN}$	$TB2 = \frac{Nb2of\ W}{Ni1 + Ni2 +}$	M2
C	$TI3 = \frac{Ni3of\ Wi3}{Ni1 + Ni2 + \cdots N}$	$TO3 = \frac{No3of\ Wo3}{Ni1 + Ni2 + \cdots N}$	$TD3 = \frac{Nd3of\ Wd3}{Ni1 + Ni2 + \cdots NiN}$	$TB3 = \frac{Nb3of\ W}{Ni1 + Ni2 +}$	M3
D	$TI1 = \frac{Ni4of\ Wi4}{Ni1 + Ni2 + \cdots N}$	$TO4 = \frac{No4of\ Wo4}{Ni1 + Ni2 + \cdots N}$	$TD4 = \frac{Nd4of\ Wd4}{Ni1 + Ni2 + \cdots NiN}$	$TB4 = \frac{Nb4of\ W}{Ni1 + Ni2 +}$	M4

Average probability of service

1. Average probability of service $AVTI = (TI1 + TI2 + TI3 + TI4)/Ni$. *Ni* is the number of company in column vector.
2. Average probability of servic $AVTO = (TO1 + TO2 + TO3 + TO4)/No$ e. *No* is the number of company in column vector.
3. Average probability of service $AVTD = (TD1 + TD2 + TD3 + TD4)/Nd$. *Nd* is the number of company in column vector/

4. Average probability of service $AVTB = (TB1 + TB2 + TB3 + TB4)/Nb$. *Nb* is the number of company in column vector

3.2 Inquiry Service Matrix

Table 2: Inquiry Services

CLIENT/SERVICE	(W1)SERVICE *i* *(Representing in probabilistic manner) with reference to column vector*	(W2)SERVICE **Inquiry Service** *o* *(Representing in probabilistic manner) with reference to column vector)*	(W3)SERVICE *d* *(Representing in probabilistic manner) with reference to column vector)*	(W4)SERVICE *b* *(Representing in probabilistic manner) with reference to column vector)*	Number of service
Ac	$T_cI_c1 = \frac{Ni_c1of\,Wi_c1}{Ni_c1 + Ni_c2 + Ni}$	$T_cO_c1 = \frac{No_c1of\,Wo_c1}{No_c1 + No_c2 + No}$	$T_cD_c1 = \frac{Nd_c1of}{Nd_c1 + Nd_c}$	$T_cB_c1 = \frac{Nb_c1of\,Wb}{Nb_c1 + Nb_c2 +}$	Mc1
Bc	$T_cI_c2 = \frac{Ni_c2of\,Wi_c2}{Ni_c1 + Ni_c2 + Ni}$	$T_cO_c2 = \frac{No_c2of\,Wo_c2}{No_c1 + No_c2 + No}$	$T_cD_c2 = \frac{Nd_c2of}{Nd_c1 + Nd_c}$	$T_cB_c2 = \frac{Nb_c2of\,Wb}{Nb_c1 + Nb_c2 +}$	Mc2
Cc	$T_cI_c3 = \frac{Ni_c3of\,Wi_c3}{Ni_c1 + Ni_c2 + Ni}$	$T_cO_c3 = \frac{No_c3of\,Wo_c3}{No_c1 + No_c2 + No}$	$T_cD_c3 = \frac{Nd_c3of}{Nd_c1 + Nd_c}$	$T_cB_c3 = \frac{Nb_c3of\,Wb}{Nb_c1 + Nb_c2 +}$	Mc3
Dc	$T_cI_c4 = \frac{Ni_c3of\,Wi_c3}{Ni_c1 + Ni_c2 + Ni}$	$T_cO_c4 = \frac{No_c4of\,Wo_c4}{No_c1 + No_c2 + No}$	$T_cD_c4 = \frac{Nd_c4of}{Nd_c1 + Nd_c}$	$T_cB_c4 = \frac{Nb_c4of\,Wb}{Nb_c1 + Nb_c2 +}$	Mc4

Average probability of service

5. Average probability of service $AVTI_c = (TI_c1 + TI_c2 + TI_c3 + TI_c4)/Ni_c$. Ni_c is the number of company in column vector.

6. Average probability of servic$AVTO_c = (TO_c1 + TO_c2 + TO_c3 + TO_c4)/No_c . No_c$ is the number of company in column vector.
7. Average probability of service$AVTD_c = (TD_c1 + TD_c2 + TD_c3 + TD_c4)/Nd_c . Nd_c$ is the number of company in column vector/
8. Average probability of service $AVTB = (TB_c1 + TB_c2 + TB_c3 + TB_c4)/NN_c . Nb_c$ is the number of company in column vector

In the making of I/O Matrix, we will put1in advertised service matrix in Table1in corresponding services cell if $AVTI > AVTI_c$otherwise 0. This will follow the same procedure for order, delivery and billing services also.

3.3 Calculation used in service clustering

Vectors formation in matrix are having row vector and column vector. To Transform into I/O parameters set of a service with rows vector and columns we have to ensure that similar services corresponding to the same vector. It can also simplify the expression of the I/O parameters set. Using entities 'attribute ratio in sharing ontology to represent the characters of vector will bring about high dimension of these vectors. but vectors calculation is still simple for the elements in vector is duality

Similarity measurement is one the basic method to cluster same object in one group. Researchers have been used several definitions for the function of similarity in [10]. $S(X, X') = X^T X'/(\|X\| \|X'\|)$ is using here to measure similarity in reference to I/O parametric sets . In the function, X and X' represent the N-dimension vectors of two services respectively ,and $X^T X'$ is the number of common attributes of X and X' and$\|X\| \|X'\|$= $(X^T X X'^T X')1/2$ is the arithmetic mean of the respective number of attributes of X and X' . I.e., this function is used to calculate the comparatively proportion between common attributes and all attributes of X and X'

Distance function formula $D(X, X') = 1/S(X, X') - 1$ used in this paper for I/O Parameter set . $D(X, X')$ and $S(X, X')$ is inverse proportion. It means that the bigger the similar degree of two vectors is, the smaller distance of two vectors is. If two vectors are equal, the distance of them is 0; if two vectors are different completely, the distance of them is infinite. Given a set of services, the distance matrix of the set can be acquired by calculating the distance of two random services in the set. DM is symmetry and all elements in the diagonal is 0 because $D(X, X') = D(X', X)$ and $D(X, X) = 0$

. $J = \frac{1}{2}\sum_{i=1}^{c} n_i \bar{s}_i$ (Variance Function) (1)

And $\bar{s}_i = MAX_{X, X' \in D_i}\{\|X - X'\|\}$; Max distance in i_{th} class. (2)

3.4. Algorithm for Service clustering

Complete graphs used to be constructed in the process of clustering. Nodes are named as service utility and the distances between each two pair of services are edges in the graph. Then the biggest complete sub-graphs will be abstracted from the graph one by one. For extracting completer graph, we are using clique method A clique is a subset of vertices of an undirected graph such that every two distinct vertices in the clique are adjacent; that is, its induced sub graph is complete. Each sub-graph is a services cluster. Before reaching to biggest complete sub-graph, we are deleting those edge which value is not smaller than the current distance threshold value. In this way, services can be classified into several clusters. In the next clustering iteration, distance threshold needs to be calculated according to the new services clusters. Such cluster will be divided, which contains a distance which is not small than the new distance threshold. In order to find out such clusters, the current clusters need to be grouped in two sets. Clusters in one set needs to be re-clustered and the other needs not When distance threshold is not bigger than the given value (called the terminal distance threshold) in equation (3) , the clustering process terminates. According to the formula of distance, it is not vary equably to the change of similar degree. In fact, it increases rapidly when similar degree approaches to 0 but it decreases slowly when the degree approaches 1. Therefore the distance threshold changes according to the result of clustering. It can be calculated by formula, which is followed in next line.

$$t = \left(MAX_{i=1,2\cdots c}\{\bar{s_i}\} + MIN_{i=1,2\cdots c}\{\bar{s_i}\} + \ 2\ \ MAX_{i=1,2\cdots c}\{\bar{s_i}\}.MIN_{i=1,2\cdots c}\{\bar{s_i}\}\right)/2$$

(3)

IV.PERFORMANCE OF WEB SERVICE CLUSTERING

In this section, we compare the performance of our Web service clustering approach with other four clustering approaches including two state-of-art clustering approaches and two versions of the proposed WT-LDA approach{10]. The details of these algorithms are given below

1.W Cluster: In this approach, semantic of WSDL-level similarity is used to cluster data which are the key feature of WSDL documents.

2. WT Cluster. In this approach, author focus on WSDL documents data and the Tagging data, which are used to clustered the Web services according to the composite semantic similarity.

3. W-LDA. In this approach, feature words from WSDL documents and cluster Web service data are the key data which are used for clustering here we are not considering any other additional information which are used in traditional LDA approach.

4. WT-LDA. In this approach, WSDL documents and the user-contributed tagging data are key focusing strategy to cluster Web services using WT-LDA.

Table 3:Performance Analysis

Clustering Approach	**Precision**	**Recall**
W Cluster[10]	0.4219	0.4378
WT Cluster[10]	0.4387	0.4553
W – LDA[10]	0.4350	0.5017
WT – LDA[10]	0.5966	0.5919
CLUSTERING BASED ON COMPLETE SUB GRAPH	0.7555	0.7555

In exhibited Table 3, we are comparing our result with other four clustering services algorithms, which are used in [10].

V.CONCLUSION

Web service dynamic nature is one the current hot topic in researcher communities. Nature of advertised services and inquiry services is to be changed as per the semantics of services. We have proposed novel technique which has been based on the complete sub graph clustering method..having with equal importance of advertised and inquiry services. We have compared our method with other methods [10], in which tags or without tags method have been used to cluster the data of different nature. We have tested our conceptualization with datasets which is available in data.world.com. We have found that higher precision and recalling rate in comparison with other clustering methods which are available in [10].

VI. FUTURE SCOPE

Web community is facing the great challenges to enable the success of future of Web-based applications which should be effective to handle of interoperability demands of web services. With appropriateness Service-Oriented Architectures along with Web Services technologies, researcher must considered those affordable solution to promote interoperability, by applying strategies like Service Composition. The role of agents during service composition should be considered in new researches.

REFERENCES

[1] T.Berners-Lee, J.Hendler, and O.Lassila, "The Semantic Web," Scientific Am.2001, vol. 284(no. 5),pp. 34-43.

[2] Gruber T R, "A Translation Approach to Portable Ontology Specifications," Knowledge Acquistion, 5(2), 1993, pp: 199-220.

[3] Lei Liu, Zhijian Wang, Jianhong Chen, "Fuzzy Ontology Model and Its Application in Semantic Web Service Description and iscovery," proceedings of ICICIC'08. 3rd International Conference on Innovative Computing Information and Control. Publication Date:18-20 June, 2008,pp.551-554.

[4] Ning Zhou, ZiLin Song, WeiHua Ai, et al. "Application of an Improved Hungarian Algorithm in Semantic WEB Service Discovery," proceedings of World Congress on Intelligent Controland Automation,2008(WCICA 2008), Publication Date:25-27 June.2008, pp.5422-542.

[5] Ren,K., Jinjun Chen, Tao Chen, et al. "Grid-based Semantic Web Service Discovery Model with QoS Constraints," proceedings of Third International Conference on Semantics, Knowledge and Grid,2007, Publication Date:29-31 Oct. 2007,pp.479-482.

[6] Zhenqi Wang, Yuanyuan Hu, "An Approach for Semantic Web Service Discovery Based on P2P Network," proceedings ofWiCOM'08.4th International Conference on Wireless Communications, Networking and Mobile Computing,2008,Publication Date:12-14 Oct. 2008, pp.1-4.

[7] Chen Wu, Chang, E., Aitken,A, "An empirical approach for semanticWeb services discovery," proceedings of ASWEC 2008.19thAustralian Conference on Software Engineering,2008, PublicationDate:26-28 March. 2008, pp.412-421.

[8] Qi Yong, Qi Saiyu, Zhu Pu, et al. "Context-Awre Semantic Web Service Discovery," proceedings of Third Internatioanl Conference on Semantics, Knowledge and Grid, Publication Date:29-31 Oct.2007,pp.499-502.

[9] Jiangang Ma, Yanchun Zhang, Minglu Li, "OMWSC- An Ontology-Based Model for Web Services Composition," Proceedings of the Fifth International Conference on Quality Software (QSIC'05),Publication Date: 19-20 Sept. 2005, pp. 464 – 469.

[10] L. Chen, Y. Wang, Q. Yu, Z. Zheng, and J. Wu, "Wt-lda: user taggingaugmented lda for web service clustering," in International onference on Service-Oriented Computing. Springer, 2013, pp. 162–176.