http://ansinet.com/itj

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSInet
Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 13 (9): 1668-1672, 2014
ISSN 1812-5638 / DOI: 10.3923/itj.2014.1668.1672
© 2014 Asian Network for Scientific Information

Research on Web Service Clustering Based on Feature Model

Zhigang Zhang
School of Computer Science and Technology, Hubei Polytechnic University, Huangshi, China

Abstract: Web service clustering helps to enhance the efficiency of service discovery and the accuracy of service clustering will influence the service discovery efficiency directly. Web service clustering is an important research direction in the area of service computing. In order to solve the problem of low accuracy of service clustering methods, this study proposes a web service clustering approach using feature model. This approach considers the features of users and web services, the corresponding feature model is extracted. It uses the traditional clustering algorithm to realize web service clustering and the clustering accuracy will be enhanced. The accuracy of service finding will be also enhanced in further. The experiments are used to validate the effectiveness of the proposed theory.

Key words: Feature model, web service, clustering

INTRODUCTION

Service computing (Zhang *et al*., 2007) includes web service selection, discovery, clustering, composition, management, evolution and some other technologies. Researchers have done more and more research work about these technologies and obtained a lot of achievement. Service clustering refers to aggregate services which meet user's specific requests and form different service clusters. It can help to enhance service discovery efficiency and accuracy.

At present, there exist some methods and technologies about web service clustering (Papazoglou, 2003). It mainly includes the following research work: Some methods use the traditional clustering algorithms to cluster the services that realize specific function. This kind of approach considers the clustering from the semantic and grammar level. Other methods cluster services form the service function, execution process, users and some other aspects. These research methods cluster services from different levels and using different mechanism. It helps to enhance service discovery efficiency. But these approaches do not consider the user and service features and the feature model is not used to realize service clustering. This will influence the service clustering accuracy and the service discovery accuracy will be influenced in further.

The feature model is introduced in section 2. In section 3, it discusses how to use feature model to cluster Web services and the overall architecture and the concrete algorithms are introduced. The related work is introduced in section 4 and the experiment is given to validate the proposed approach in section 5. Finally, the conclusion and the next step research work are given.

FEATURE MODEL

Feature model (Zhang and Mei, 2003) is a feature-oriented domain requirements specification model and it can reflect the relatively stable characteristics of the area as well as the relationship between these characteristics. There are two types of features: Common features and changing features. Common features exist in the field of each member with the system and changing features are only presented in certain areas of the members. Variability can reflect the member characteristics on the basis of the common feature. Through the optionality and variability mechanism, some part of features can embody the variability of specific object. The constraint relationship (Xing, 2007) between features mainly include dependence, mutual exclusion, alternative and so on. We can use different modes to express the feature relations, as shown in Fig. 1.

In the above Figure, fea_1~fea_{14} denote the different features in feature model. There exist different relations between these features and we can get the feature

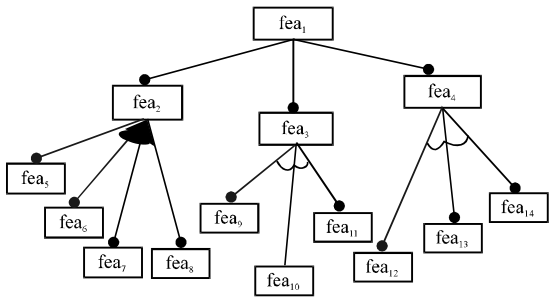

Fig. 1: Feature model

relations as follows: FM = {<fea_1, Mandatory, fea_2>, <fea_1, Mandatory, fea_3>, <fea_2, Optional, fea_4>, <fea_2, OR, fea_5>, <fea_2, OR, fea_6>, <fea_3, Alternative, fea_{10}>, <fea_3, Alternative, fea_{11}>, <fea_4, Mandatory, fea_{12}>…}.

WEB SERVICE CLUSTERING USING FEATURE MODEL

Overall architecture: This study proposes a web service clustering approach based on feature model and the overall architecture of the proposed method is shown in Fig. 2.

In the Fig. 2 it firstly extracts the features of web services and users on the internet, then the specific feature extraction method is used to get features and the corresponding feature model will be constructed. The web service clustering method based on feature mode is used to cluster web services to form different service clusters. The service clustering accuracy will be enhanced and the service discovery efficiency will be improved in further.

Web service feature extraction algorithm::

Definition 1: Web Service (ws): ws = {Name, Input, Output, Operation} and:

- Name refers to the name of ws
- Input = {IN_i,∈Ini∈Entity, i∈1, 2...n}, Input refers to the input set of ws
- Output = {OUT_j, OUT_j∈Entity, j∈1, 2...m}, Output refers to the output set of ws
- Operation = {OP_k, OP_k∈Entity, k∈1, 2...l}, Operation refers to the operation set of ws

Definition 2: Feature Model(FM): FM = {Fea, Rel} and:

- Fea = {fea_i, fea_i∈Entity, i∈1, 2...n}, Fea refers to the features in FM
- Rel = {<fea_i, r_k, fea_j>, r_k∈{Mandatory, Optional, Alternative and, OR}, fea_i, fea_i∈Fea, i, j∈1, 2...n, k∈1, 2...m}, Rel refers to the feature relations in FM

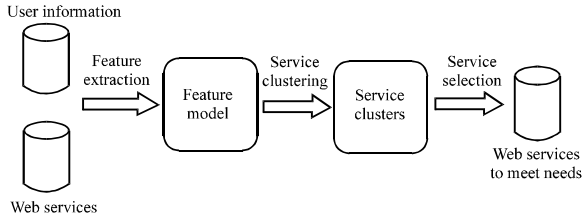

Fig. 2: Web service clustering using feature model

On the basis of the definition of web service and feature model, the process of extracting web service and user's features is shown in the following Algorithm 1.

Algorithm 1: Web service feature extraction algorithm

Input: *WS* = {ws_i, *I* = 1,2,….*snum*}, *User*, *DM*
Output: *FM*1. *FM*←∅
2. foreach *web service* ws_i in *WS*
3. foreach *input* IN_i in ws_i.*Input*
4. *FM.Fea*←*FM.Fea*∪IN_i
5. endfch
6. Using step 3-5 to add features of *Name*, *Output* and *Operation*
7. endfch
8. add user's goal in *User* to *FM*
9. foreach *feature* fea_i in *FM.Fea*
10. foreach *feature* fea_j in *FM.Fea*
11. use the domain model in *DM* to determine the relation r_k between fea_i and fea_j
12. *FM.Rel*←*FM.Rea*U<fea_i, r_k, fea_j>
13.endfch
14. return *FM*

In the above algorithm, DM is the domain model of specific area and it is generated through domain modeling towards specific title. This algorithm is used to get the feature model of web service to be clustered and the corresponding users. Step 2-7 is used to get Name, Input, Output and Operation features of services in WS. Step 8 is used to get the user features. Step 9-12 is used to get feature relations through domain model in FM. Finally, return FM.

Web service clustering algorithm: The process of clustering Web services using feature model is shown in the following Algorithm 2.

Algorithm 2: Web service clustering algorithm

Input: *WS* = {ws_i, *I* = 1,2,….*snum*}, *FM*
Output: *cluster*
1. *cluster*←∅
2. Initializing each cluster *cluster*[*i*]←ws_i
3. foreach *feature* fea_i in *FM.Fea*
4. foreach *triple* <fea_m, r_k, fea_n> in *FM.Rel*
5. if(fea_i==fea_m||fea_i==fea_n) then
6. foreach *web service* ws_i in *WS*
7. find the *Input*/*Output*/*Operation*/*Name* equals to fea_m
8. endfch
9. foreach *web service* ws_j in *WS*
10. find the *Input*/*Output*/*Operation*/*Name* equals to fea_n
11. endfch
12. if(r_k==*Mandatory* || r_k==*And*) then
13. *cluster*[*i*]?*cluster*[*i*]?ws_j
14. end if
15. endfch
16. endfch
17. return *cluster*

The above algorithm gives the process of how to cluster the web services based on the feature model of

services. Step 1-2 is used to do initialization of cluster. It uses step 3-16 to cluster web services. Towards each feature fea_i in FM, it finds the feature fea_j which has relation with fea_i. Then the services which are corresponding with the features are clustered according to the different feature relations. And the all the services are clustered and return finally.

RELATED WORK

There exists a lot of research work about web service clustering and some research methods use the traditional clustering algorithm to cluster web services using the basic information of web services, such as k-means, k-medoids, hierarchical and so on. Other research methods about web service clustering mainly consider the service function, execution process, users and other factors to cluster web services. This kind of method realizes web service clustering from different levels.

Sudha *et al*. (2006) have proposed a clustering based approach for facilitating semantic web service discovery. The approach has several appealing features: Minimizing the requirement of prior knowledge from both service consumers and publishers; avoids exploiting domain dependent ontologies; visualizing the semantic relationships among web services. Elgazzar *et al*. (2010) have proposed a novel technique to mine Web Service Description Language (WSDL) documents and cluster them into functionally similar web service groups. The approach has shown good performance for clustering Web services based on function similarity. Liu *et al*. (2009) have proposed an efficient approach to facilitating the service consumer on discovering Web services. The homogeneous service community contains two types of discovery: The search of similar operations and that of composible operations. A similarity measurement model is described for Web services by leveraging the metadata from WSDL and a graph-based algorithm to support both of the two discovery types is designed. Skoutas *et al*. (2010) have proposed a methodology for ranking the relevant services for a given request, the objective measures based on dominance relationships defined among the services are introduced. They investigate methods for clustering the relevant services in a way that reveals and reflects the different trade-offs between the matched parameters. The effectiveness and the efficiency of the proposed techniques and algorithms are validated through the experiment. Sun and Jiang (2008) proposes a mechanism to support semantic web service discovery. Petri net is adopted as a modeling language for the specification of service process model. The service clustering methodology groups similar services according to the functional similarity and process similarity. The utilization of service clustering can potentially enable service matchmaker to significantly reduce the overhead. For a service request, a service matchmaker compares the functionality compatibility and process consistency with the candidate services. Dong *et al*. (2004) have proposed a similarity search method for web services. They described the algorithms underlying the Woogle search engine for web services. The engine supports similarity search for web services, such as finding similar web service operations and finding operations that compose with a given one.

There exists some research work about feature model and web services. Lamia *et al*. (2009) provide an ontology framework for feature modeling which consists of an ontology that formally provides a specification for feature models. They also provide means to integrate segmented feature models and provide a rule based model consistency check and conflict detection. SWRL rules are used to implement the rules and a DL reasoner is used to evaluate the rules and infer extra interesting information. Jens and Andreas (2007) have considered realistic web service features for semi-automatic composition. They present two slight changes in the working of the composition unit that were needed to allow for some of the named realistic features. A greater flexibility for the application of our composer is also provided.

In the above research work, the different approaches are used to cluster web services from different levels and using different mechanisms. But these methods do not consider the features of web services and the corresponding users. Thus the service clustering accuracy will be influenced. This study constructs the feature model based on the extracting web service and user's features. Then it uses web service clustering method based on feature model to cluster services. The accuracy of service clustering and service selection will be enhanced in further.

EXPERIMENT

Experiment dataset: We use the publicly available service retrieval test collection OWLS-TC v4 (http://projects.semwebcentral.org/projects/owls-tc/). It contains real-world Web service descriptions and it provides 1083 semantic Web services from nine different domains (education, medical care, food, travel, communication, economy, weapons, geography and simulation). It provides a set of 42 test queries which are associated with relevance sets to conduct performance evaluation experiments. The ontology set includes the including ontology concepts and the semantic relations between

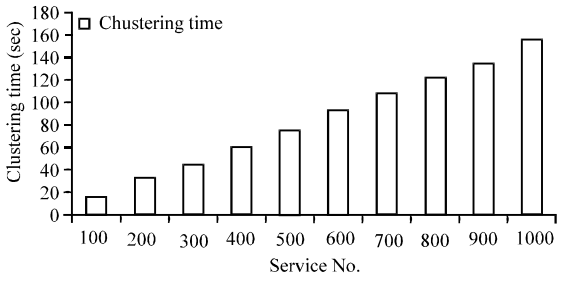

Fig. 3: Web service clustering time comparison

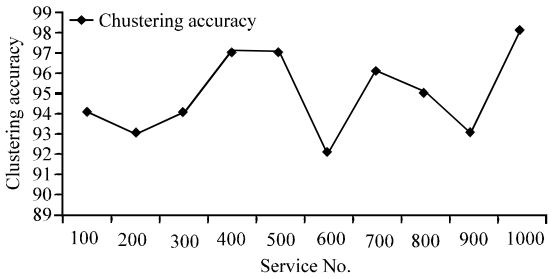

Fig. 4: Web service clustering accuracy comparison

Table 1: Comparison of feature model extraction time of different service No.

Service No.	100	200	300	400	500	600	700	800	900	1000
Time (sec)	32	61	97	118	158	177	214	236	271	331

concepts. It helps to get the domain model of different areas and thus the corresponding feature models can be got.

Experiment analysis: This thesis proposes a web service clustering method using feature model. This section designs experiments to compare the time and accuracy of this approach with other methods. We mainly consider the Name, Input and Output information of services and the information are extracted to construct feature model.

Experiment 1: Comparison of feature model extraction time under different service numbers. This experiment is used to count the time of extracting feature model towards the different service numbers and corresponding ontology. In the condition of different service numbers: 100, 200, 300, 400, 500, 600, 700, 800, 900 and 1000, it extracts the feature models and gets the corresponding time. The experiment result is shown in Table 1.

From Table 1, the time of extracting feature model is different under different service numbers. As the service number increases, the using time is increased. We can see the time of extracting feature model of one service is about 0.3s. The time is not too much under different service numbers.

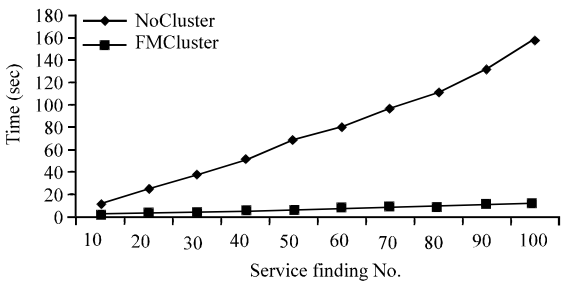

Fig. 5: Comparison of service finding time

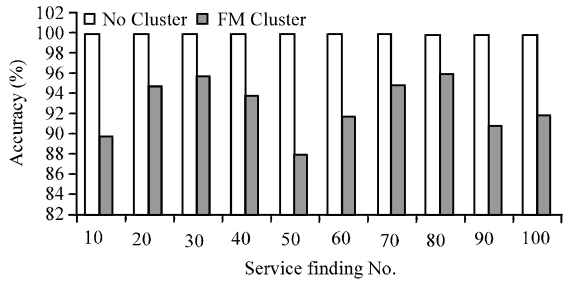

Fig. 6: Comparison of service finding accuracy

Experiment 2: Comparison of service clustering time and accuracy. On the basis on the constructed feature model, this experiment compares the time and accuracy in the condition of different service numbers: 100, 200, 300, 400, 500, 600, 700, 800, 900 and 1000. The experiment result is shown in Fig. 3 and 4.

From the above Fig. we can see the service clustering time is different in the condition of different service numbers. As the service number increased, the time of clustering is becoming larger. In addition, the service clustering accuracy is more than 90%. It can cluster the most services and this embodies the effectiveness of the proposed methods.

Experiment 3: Comparison of the service finding time and accuracy. The service search space will be reduced largely after the services are clustered using the feature model clustering method and it helps to enhance service discovery efficiency. We denote the clustering method in the study as FMCluster and use NoCluster to express the approach which is not using the clustering method. In the condition of 1000 service numbers, we use the above two methods and thus to find the needed services under different service request numbers: 10, 20, 30, 40, 50, 60, 70, 80, 90 and 100. We compare the service finding time and accuracy, the result is shown in Fig. 5 and 6.

From the above Fig. 6, we can the service finding efficiency using FMCluster method is apparently more

than NoCluster method. The time that is used by the former method is less than the latter. This is because FMCluster method can reduce the service searching space and enhance service discovery efficiency. But the service finding accuracy of FMCluster method is less than NoCluster, it's between 88 and 100%. This is related to the accuracy of FMCluster method.

CONCLUSION

How to find the services to meet user's requests quickly, accurately and efficiently is a key issue needs to be solved in modern world. In order to solve the problem of low clustering and service finding accuracy, this study has proposed a new web service clustering method using feature model. The feature extraction algorithm is used to get the features of users and web services. Then is uses feature model based algorithm to realize web service clustering and thus to form different service clusters. The service clustering and discovery accuracy will be enhanced in further.

The next step research work includes combining the ontology and feature model to enhance service clustering accuracy in further. We will consider the user and web service features comprehensively and the service QoS information will also be used to build feature model. Services will be clustered from the aspects of function and non-function.

ACKNOWLEDGMENT

This research project was supported by the Hubei Province Young College Teachers in-depth Business Action Plan Project under Grant No. XD2012495, the Research on Teaching Reform of Computer Major in Local Universities Experiment Project under grant No. 2013B15.

REFERENCES

Dong, X., A. Halevy, J. Madhavan, E. Nemes and J. Zhang, 2004. Similarity search for web services. Proceedings of the of 30th International Conference Very Large Data Bases (VLDB), August 2004, Toronto, Canada, pp: 372-383.

Elgazzar, K., A.E. Hassan and P. Martin, 2010. Clustering WSDL documents to bootstrap the discovery of web services. Proceedings of the 2010 IEEE International Conference on Web Services, July 5-10, 2010, Miami, FL., pp: 147-154.

Jens, L. and F. Andreas, 2007. Considering realistic web service features for semi-automatic composition. Proceedings of the 3rd South-East European Workshop on Formal Methods, Novembre 30- Dicembre 1, 2007, Thessaloniki, Greece.

Lamia, A.Z., K. Frederic and D.T. Olga, 2009. Applying semantic web technology to feature modeling. Proceedings of the 24th Annual ACM Symposium on Applied Computing (SAC), March 8-12. 2009, Waikiki Beach, Honolulu, Hawaii, USA.

Liu, X., G. Huang and H. Mei, 2009. Discovering homogeneous web service community in the user-centric web environment. IEEE Trans. Serv. Comput., 2: 167-181.

Papazoglou, M.P., 2003. Service-oriented computing: Concepts, characteristics and directions. Proceedings of 4th International Conference on Web Information Systems, December 10-12, 2003, IEEE, Piscataway, USA., pp: 3-12.

Skoutas, D., D. Sacharidis, A. Simitsis and T. Sellis, 2010. Ranking and clustering web services using multicriteria dominance relationships. IEEE Trans. Serv. Comput., 3: 163-177.

Sudha, R., H. Yousub and Z. Huimin, 2006. A clustering based approach for facilitating semantic Web service discovery. Proceedings of the 15th Annual Workshop on Information Technologies and Systems, March 2006, Las Vegas, USA, pp: 1-6.

Sun, P. and C.J. Jiang, 2008. Using service clustering to facilitate process-oriented semantic web service discovery. Chinese J. Comput., 31: 1340-1353.

Xing, Y., 2007. Feature model driven web services composition approach and its support tool. Chin. J. Software, 18: 1582-1591.

Zhang, L.J., J. Zhang and H. Cai, 2007. Services Computing. Tsinghua University Press, Beijing, China, ISBN-13: 9783540382843, pp: 1-80.

Zhang, W., H. Mei, 2003. A feature-oriented domain model and its modeling process. Chinese J. Software, 14: 1345-1356.