

Research Article

Collaborative Filtering Based Hybrid Approach for Web Service Recommendations

¹G. Vadivelou and ²E. Ilavarasan

¹Department of Computer Science and Engineering, Bharathiar University, Coimbatore, Tamilnadu, India

²Department of Computer Science and Engineering, Pondicherry Engineering College, Pondicherry, India

Abstract: Now a days, Web services are becoming the primary source for constructing software system over Internet. The quality of whole system greatly depends on the QoS of single web service, so QoS information is an important indicator for service selection. But in reality, QoSs of some Web Services may be unavailable for users. How to predicate the missing QoS value of Web service through fully using the existing information is a difficult problem. This study first proposes a novel method for clustering similar web services using semantic approach in order to overcome the limitations of pattern-matching approach and finally proposes a cluster based approach using Slope One Collaborative filtering method to predict the QoS values for similar web services for similar users. Proposed approaches are applied on a dataset consisting of WSDL files and QoS values for various QoS parameters which are collected from the Internet and the proposed approaches shows better qualities with respect to clusters formed and the QoS values predicted.

Keywords: Clustering, collaborative filtering, slope one, web service, WSDL

INTRODUCTION

A Web service is a software system designed to support interoperable machine-to-machine interaction over a network. It has an interface described in a machine-processable format specifically Web Service Description Language (WSDL). Other systems interact with the Web service in a manner prescribed by its description using Simple Object Access Protocol (SOAP) messages, typically conveyed using HTTP with an XML serialization in conjunction with other Web-related standards. Web services are Internet-based application components published using standard interface description languages and universally available via uniform communication protocols.

As an ever-increasing number of Web services published and deployed on the Internet, it is critical for service users to discover desired services that match their requirements. Web service discovery is the process of locating effective approaches to find, match and access the Web services published in a local repository or located across the Internet.

To effectively discover and match Web services, it is necessary to establish some kinds of the correlations between a user and potential services available, which can be achieved through two steps. On the client side, a service user can express his/her requirements described in the form of natural language and then use a service search engine to interact with a set of potential Web services. On the side of services

providers, on the other hand, they advertise services' capabilities through some descriptions such as the Web services' names, the operations' descriptions and the operations' names described by WSDL. In this situation, they also assume that clients would agree on the words used to describe the Web services. However, the problem of how to deal with the agreement and how to associate the users' requirements to the advertisements of web services would have a critical impact on discovering web services. Therefore, locating desired services might be difficult.

In addition, participants who request web services through the web in distributed collaboration need to apply appropriate approaches to access provided web services that meet their requirements. Therefore, efficient collaboration in distributed context is achieved by automatic web services discovery and binding web services in order to satisfy their requirements.

Recommender systems have been a popular topic of research ever since the ubiquity of the web made it clear that people of hugely varying backgrounds would be able to access and query the same underlying data. Recommendation agents need to employ efficient prediction algorithms so as to provide accurate recommendations to users. There are generally two methods to formulate recommendations both depending on the type of items to be recommended, as well as, on the way that user models are constructed. The two different approaches are content-based and

Corresponding Author: G. Vadivelou, Department of Computer Science and Engineering, Bharathiar University, Coimbatore, Tamilnadu, India

This work is licensed under a Creative Commons Attribution 4.0 International License (URL: <http://creativecommons.org/licenses/by/4.0/>).

collaborative filtering, while additional hybrid techniques have been proposed as well.

Content based algorithms are principally used when documents are to be recommended, such as web pages, publications, jokes or news. The agent maintains information about user preferences either by initial input about user's interests during the registration process or by rating documents. Recommendations are formed by taking into account the content of documents and by filtering in the ones that better match the user's preferences and logged profile. The documents considered in this study for recommendations are WSDL documents.

The Vector Space Model (VSM) is one of the simplest methods is based on the exact matching of terms that can be found in documents. It converts texts to n-dimensional vectors for measuring distances among them. As a distance measure, the cosine of the angle between two vectors is used in most cases. The result is a value of similarity ranging from 0 to 1, where 1 indicates an exact/high match between terms and 0 indicates that there is no match. This means the higher the value of the cosine, the higher the likelihood that two terms are equal.

However, the fact that the method is solely based on the exact matching of words raises problems, such as synonymy and polysemy. Synonymy deals with different words having the same meaning. For example, car and automobile are synonyms, which are not considered to be equal in the VSM. This can lead to poor recall, meaning that not all relevant information sources are discovered. Polysemy refers to words having more than one distinct meaning. Another problem is that common words, such as-the and -is and correlating high similarity measures result in a high match, which does not represent the actual desired result.

To alleviate the drawbacks of the VSM, presented Latent Semantic Analysis (LSA) (Papadimitriou *et al.*, 2000) a statistical, corpus-based text comparison method. Throughout in this study, it is sometimes also referred to as Latent Semantic Indexing (LSI), which is primarily used in the field of information retrieval, whereas LSA is used in other application areas. The process of learning words that are related to each other is based on their statistical co-occurrence together in a context.

Also, in the presence of multiple Web services with identical or similar functionalities, Quality of Service (QoS) provides non-functional Web service characteristics for the optimal Web service selection. Since the service providers may not deliver the QoS it declared and some QoS properties (e.g., network latency, invocation failure-rate, etc.) are highly related to the locations and network conditions of the service users, Web service evaluation by the service users can obtain more accurate results on whether the demanded Web services fit the functional and non-functional requirements (Wu *et al.*, 2007; Zeng *et al.*, 2004; Zheng

and Lyu, 2008a). However, evaluation from the service user's perspective has the following drawbacks.

Firstly, it requires service invocations and imposes costs for the service users. At the same time, it consumes resources of the service providers. Secondly, there may be too many service candidates to be evaluated and some suitable Web services may not be discovered by the service users. Finally, most of the service users are not experts on the Web service evaluation and the common time-to-market constraints limits an in-depth evaluation of the target Web services.

Memory-based Collaborative filtering is Classified into User-based (UPCC) and Item-based (IPCC). In user based approaches, the value of ratings user u gives to item i is calculated as an aggregation of some similar users rating to the item. The user-based algorithm calculates the similarity between two users, produces a prediction for the user taking the weighted average of all the ratings. Multiple mechanisms such as Pearson Correlation Coefficient and Vector Cosine based are used for similarity computation. Pearson Correlation Coefficient (PCC) has been introduced in a number of recommender systems for similarity computation, since it can be easily implemented and can achieve high accuracy.

The Pearson correlation similarity of two users a and b is defined as:

$$sim(a, b) = \frac{\sum_{p \in P} (r_{a,p} - \bar{r}_a)(r_{b,p} - \bar{r}_b)}{\left(\sqrt{\sum_{p \in P} (r_{a,p} - \bar{r}_a)^2} \sqrt{\sum_{p \in P} (r_{b,p} - \bar{r}_b)^2} \right)} \quad (1)$$

where,

P = The set of all items

$r_{a,p}$ = The rating of user a on item p

\bar{r}_a = The average rating of user a

Now, the rating for item i for user a can be predicted using a simple weighted average, as in:

$$pred(a, p) = \bar{r}_a + \frac{\sum_{b \in N} sim(a, b) * (r_{b,p} - \bar{r}_b)}{\sum_{b \in N} sim(a, b)} \quad (2)$$

When applied to millions of users and items, conventional neighborhood-based CF algorithms do not scale well, because of the computational complexity of the search for similar users. As an alternative, item-to-item Collaborative Filtering (Deshpande and Karypis, 2004; Sarwar *et al.*, 2001) where rather than matching similar users, they match a user's rated items to similar items. In practice, this approach leads to faster online systems and often results in improved recommendations.

In this approach, similarities between pairs of items i and j are computed offline using Pearson correlation, given by:

$$sim(i, j) = \frac{\sum_{u \in U} (r_{u,i} - \bar{r}_i)(r_{u,j} - \bar{r}_j)}{\left(\sqrt{\sum_{u \in U} (r_{u,i} - \bar{r}_i)^2} \sqrt{\sum_{u \in U} (r_{u,j} - \bar{r}_j)^2} \right)} \quad (3)$$

where,

U = The set of all users who have rated both items i and j

$r_{u,i}$ = The rating of user u on item i

\bar{r}_i = The average rating of the ith item across users

An enhanced PCC for the similarity computation between different service users is defined as:

$$sim'(a, u) = \frac{2 * |I_a \cap I_u|}{|I_a| + |I_u|} \quad (4)$$

Invoked by user a and user u, respectively.

Now, the rating for item i for user a can be predicted using a simple weighted average, as in:

$$pred(u, i) = \bar{r}_i + \frac{\sum_{ik \in S(i)} sim(ik, i) * (r_{u,ik} - \bar{r}_k)}{\sum_{ik \in S(i)} sim(ik, i)} \quad (5)$$

where, K is the neighborhood set of the k items rated by a that are most similar to i, For item-based Collaborative Filtering too, one may use alternative similarities metrics such as adjusted cosine similarity.

Slope one algorithms are easy to implement, efficient to query, reasonably accurate and they support both online queries and dynamic updates, which makes them good candidates for real-world systems. An online rating-based Collaborative Filtering CF query consists of an array of (item, rating) pairs from a single user. The response to that query is an array of predicted (item, rating) pairs for those items the user has not yet rated. This study aims to provide robust CF schemes that are.

Easy to implement and maintain: All aggregated data should be easily interpreted by the average engineer and algorithms should be easy to implement and test.

Updateable on the fly: The addition of a new rating should change all predictions instantaneously.

Efficient at query time: Queries should be fast, possibly at the expense of storage.

Expect little from first visitors: A user ith few ratings should receive valid recommendations.

Accurate within reason: The schemes should be competitive with the most accurate schemes, but a minor gain in accuracy is not always worth a major sacrifice in simplicity or scalability.

The Slope one schemes used in this study fulfills all the five goals. This study uses Slope One scheme

and its variants Weighted Slope One for predicting QoS values.

To overcome the drawbacks described above, the proposed approach in this study employs an effective and novel hybrid collaborative filtering algorithm, for Web service selection and QoS values prediction. Collaborative filtering methods can automatically predict the QoS performance of a Web service for an active user by employing historical QoS information from other similar service users, who have similar historical QoS experience on the same set of commonly-invoked Web services. Hybrid approach combines both collaborative and content based recommendation algorithms. By the proposed hybrid collaborative filtering method, the QoS performance of Web services can be predicted for active service users without requiring the service users to conduct Web service evaluation and to find out a list of service candidates themselves.

The main objectives of the proposed work are as follows:

- To use more semantics in the representation
- To reduce the dimensionality of the feature vectors formed
- To improve the cluster quality
- To retrieve relevant services satisfying the user request
- To predict the QoS of web services for recommending web services to the active user with minimum error rate

LITERATURE REVIEW

A number of clustering techniques for forming service community of functionally equivalent web services exist. Research in web mining has recently gained much attention due to the popularity of web services and the potential benefits that can be achieved from mining Web services description files. Non-semantic web services are described by WSDL documents. Non-semantic web services are more popular and supported by both the industry and development tools. The discovery process is quite different according to the web services description method. Semantic web services are discovered by high level match-making approaches, whereas non-semantic Web services discovery uses information retrieval techniques.

Liu *et al.* (2005) have compared Document Frequency (DF), Term Contribution (TC), Term Variance (TV) and Term Variance Quality (TVQ) as unsupervised feature selection on document clustering. No predefined label on document clustering is the reason of using unsupervised feature selection. The experiment shows that the unsupervised feature

selection can improve the accuracy of document clustering.

Nayak and Lee (2007) propose a method to improve the Web service discovery process using the Jaccard coefficient to calculate the similarity between Web services. He provides the user with related search terms based on other users' experiences with similar queries. If the query comes at first time, then users' experience doesn't work.

Wei and Wilson (2009) applies a text mining techniques to extract features such as service content, context, host name and name, from Web service description files in order to cluster web services. They propose an integrated feature mining and clustering approach for web services as a predecessor to discovery, hoping to help in building a search engine to crawl and cluster non-semantic web services.

QoS based approaches for Web services selection have been discussed in a number of recent literature (Cardellini *et al.*, 2007; El Haddad *et al.*, 2008; Ma *et al.*, 2007; Zheng and Lyu, 2008b) which enables optimal Web services to be identified from a set of candidates according to the QoS performance of the candidates and the preference of the service users. Our work is quite different from these approaches since we employ the information of similar service users as well as similar Web service items to predict the QoS performance of Web services. Our method requires no Web service invocation, which will save a lot of resource and time.

There is limited work in the literature employs collaborative filtering methods for Web service recommendation, since there is no large-scale Web service QoS datasets available for studying the QoS value prediction results. Without convincing and sufficient real-world Web service QoS data, the characteristics of Web service QoS information cannot be fully mined and the proposed recommendations algorithms will become merely are development of the traditional movie recommendation algorithms, which may not be applicable to the Web service recommendation. Work (Karta, 2005; Sreenath and Singh, 2003) mentions the idea of applying collaborative filtering methods to Web service recommendation and employs the Movie Lens dataset for experimental studies, which is not convincing enough. Work (Shao *et al.*, 2007) proposes a user-based PCC method for the Web service QoS value prediction, however the performance of UPCC is not good when the given number is small.

Pearson correlation-based algorithms are the mainstream strategies to treat such problem at current stage. Shao *et al.* (2007) firstly attempted to use Pearson similarity-based collaborative filtering to provide the QoS value of a specific Web service. But their experiments are performed on a dataset in small scale and the error analysis is not so sufficient. Subsequent, Zheng *et al.* (2011) first collected plenty of QoS records

from different service users via a monitoring platform Planet-lab. Then, they combined user-based and item-based CF together to form a comprehensive algorithm (i.e., WsRec) for service's QoS prediction. Their WsRec exhibits better performance than the single user-based or item-based prediction algorithm.

Recently, some improvements on Pearson correlation-based algorithm have been proposed. Liu's research group presented a Personalized Hybrid Collaborative Filtering (PHCF) algorithm by considering the personal information about service user (Jiang *et al.*, 2011). However, it is not so easy to obtain such personal information, so the application of their method is limited. Sun *et al.* (2011) adopted an improved similarity measure for Web service similarity computation and the corresponding Normal Recovery Collaborative Filtering (NRCF) was proposed for personalized Web service recommendation. In essence, it is only a minor modify on the similarity measure for the WsRec prediction framework. In addition, Shi *et al.* (2011) presented a linear regression prediction algorithm for Web service's QoS based on clustering user in respect to location and network condition. It is not hard to find that the distance between users plays a significant role for prediction precision, however, which is not easily measured in practice.

Of course, there are also some Slope One-based methods for service's QoS prediction. Xie *et al.* (2010) presented a personalized context-aware QoS prediction method based on the Slope One approach. In this study, the basic Slope One algorithm is used for prediction, but it has been validated to be not very precise in our experiments. Then, Li *et al.* (2012) utilized an enhanced Slope One method called Bi-Polar Slope One to predict the ratings of Web services. On the one hand, their approach mainly aim sat the rating prediction problem. On the other hand, Bi-Polar phenomenon may be exists in the dataset in rating style, but not obvious in QoS data (i.e., the continuous data type).

METHODOLOGY

In order to recommend web services to the active users based on query string which contains both functional and non-functional parts, first the functional part of the query is considered to select the similar web services and then the QoS attribute values of non-functional parameters are considered to further find top k similar web services to give recommendations to the active user.

Considering the functional part of the query string, in order to overcome the limitations of keyword based approaches used by previous authors, semantic approaches be applied for clustering similar web services to reduce the search space to find the relevant web services corresponding to user's requirements. In this study, the keyword based approach is considered as the baseline and it is compared with LSA based approach.

Keyword based service discovery: Keyword based approaches are widely used in traditional information retrieval systems. An information requester submits the system with a query that consists of a number of keywords in order to retrieve the desired documents. The retrieval system returns stored documents in answer to the information requester based on the similarity between the query and the stored documents. Here similarity means that the documents contain particular keywords from the requester’s query or those documents prove similar enough to the corresponding the query and those documents are returned to the information requester.

Currently, keywords based mechanism is one of the techniques for Web services discovery and matching. The proposed work includes keyword based approach in order to cluster similar services as a separate module. The work has been used to compare with semantic approaches.

Steps for using Keyword-based Approach:

- Collect a set of WSDL documents used for Feature Extraction (service name, operation name, message name, port name and port type)
- Text Preprocessing (tokenize, compact cleavage, Stop Words removal, Stemming (Porter Stemmer))
- TF-IDF matrix computation
- Similarity between the documents
- Clustering documents using K-means
- Pass query and then find similarity between cluster and query
- Finally retrieves the relevant document from the dataset

Proposed LSA based service discovery: Semantic approach using LSA is dependent on the combination of the keyword technique and the semantics extracted from the service descriptions. The objectives of LSA approach are to handle the poor scalability in the Web environment and the issue of lacking semantics. To realize these goals, a large service collection is first partitioned into a set of smaller clusters by using k-means clustering algorithm. After finding the right cluster related to a query, the SVD technique is applied to the cluster so that service matching against the query can be carried out at the concept level.

The LSA approach is based on the assumption that the efficiency of finding services can be improved if relevant service cluster can be located before the extracting semantics algorithm is implemented. To begin with, given a query, the proposed approach retrieves a set of samples of Web services from a source of Web services to form an initial data set. Instead of directly applying the SVD to this large initial data set, the proposed work partitions it into a set of smaller clusters by using a k-means clustering algorithm, aiming to reduce the number of services retrieved. This phase focuses on analyzing the syntactical correlation between the query

Table 1: A simple example to illustrate the slope one approach

	Item I	Item J
User A	2	3
User B	4	? = 4 + (3 - 2)

and service descriptions. As a next step, the proposed work filter out those Web services whose contents are not compatible with a user’s query via finding relevant cluster.

Finally, SVD technique is applied to the relevant clusters via computing the similarity between a query and the centroid of each cluster to capture semantic concepts hidden behind the words in a query and the advertisements in services, so that services matching against the query are expected to be carried out at an advanced concept level.

The pseudo code for LSI based approach algorithm is given as following:

1. Retrieving initial service collection
2. Partitioning the collection into a set of clusters
3. Finding relevant cluster
4. Applying SVD to the cluster
5. Semantic Matching service against query
6. if the results match the query then go to step 10
7. else choosing next cluster
8. go to step 4
9. end if
10. end

Proposed cluster-based slope one approach: The slope one algorithm for predicting rating is based on the intuitive principle of a “popular differential” (deviation) between items for users (Sarwar *et al.*, 2001). For a simple example, consider Table 1, item I gets 2 from user A, while item J gets 3 and user B gives a rating of 4 to item I, then what’s the rating of user B for item J? By the slope one approach, this study can calculate the deviation of item J respect to item I from user A: $dev_{ji} = 3 - 2 = 1$. Then this approach can infer the rating of user B for item J: $4 + dev_{ji} = 5$.

Formally, the slope one algorithm exploits the predictor of the form $f(x) = x + b$ to indicate the relation of the item pair, where b is a constant deviation and x is a variable representing rating value. So the formal definition of dev_{ij} can be given. Given two items i and j , ($i \neq j$), the deviation of item i respect to j is as follow:

$$dev_{ij} = \frac{\sum_{u \in S_{ij}} (r_{ui} - r_{uj})}{|S_{ij}|} \tag{6}$$

Here S_{ij} denotes the set of users who both rate item i and item j and $|S_{ij}|$ is the number of the users in the set. All deviations construct a deviation matrix which is a skew-symmetric matrix. With the deviation matrix constructed, it is easy to make predictions by using the

slope one algorithm. Regarding $r_{uj} + dev_{ij}$ as a prediction for rating r_{ui} given r_{uj} , a more reasonable predictor might be the average of all such predictions:

$$P_{ui} = \frac{\sum_{j \in S(u)-\{i\}} (r_{uj} + dev_{ij})}{|S(u) - \{i\}|} \quad (7)$$

Here $S(u) - \{i\}$ represents the set of all items rated by user u except item i .

In the above approach, this study average the predictions calculated from each individual item to generate a final prediction. Actually they have different contributions to the final prediction. Simply using the average is not a good solution. Thus, a better predictor (i.e., the weighted slope one approach) was proposed as follow:

$$P_{ui} = \frac{\sum_{j \in S(u)-\{i\}} (r_{uj} + dev_{ij}) * c_{ij}}{\sum_{j \in S(u)-\{i\}} c_{ij}} \quad (8)$$

where, c_{ij} in the equation is the number of users who both rate item i and item j . The bigger the c_{ij} , the more reliable the corresponding prediction (i.e., $r_{uj} + dev_{ij}$).

Now considering the non-functional parts of the query string, firstly, items were clustered using K-Means algorithm based on the QoS values. The algorithm uses the QoS values in the class to calculate the deviation between the target items and other items, which means using fewer, higher quality ratings to help the current active users to predict the ratings on target items. This means the algorithm cannot only solve the data sparsity problem of collaborative filtering, but also improve the prediction accuracy. The dataset released by Al-Masri and Mahmoud (2007a) is used for simulation.

First, a cluster is selected based on functional part of the query and then the proposed cluster slope one approaches are used to predict the QoS values for the web services which are not invoked by any user. The cluster based approach is used in order to reduce the dimensionality of the user-item matrix. After predicting the QoS values, the web services are ranked and top k web services are recommended to the active by sending the corresponding WSDL file of the top k web services.

OBSERVED RESULTS AND DISCUSSION

The keyword based and LSA based approaches in the proposed work are implemented using RapidMiner 5.3.00, Java, Net Beans IDE 6.1 and mySql under Windows platform. The dataset which consists of 228 WSDL are gathered for clustering from various online Web service repositories such as WebserviceX.Net, SOATrader.com and seekda.com. The observed results are given below.

Table 2: Average recall, precision and F-measure values

Approaches	Avg. recall	Avg. precision	Avg. F-measure
Keyword-based	36.854	48.248	40.438
LSA-based	54.534	63.904	56.732

Avg.: Average

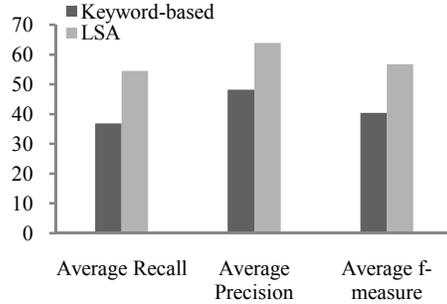


Fig. 1: Average recall, precision and F-measure values

Comparison between keyword-based and LSA approaches: Table 2 and Fig. 1 show the results of average recall, precision and F-Measure value for keyword-based and documents.

Predicted QoS values by the proposed cluster based slope one method: The dataset released by Al-Masri and Mahmoud (2007a, b), is considered for simulation purpose. The Slope One method and its variant Weighted Slope One are implemented and the results using the clustered approach shows better prediction accuracy when compared to memory based approaches. The subset of the dataset which is considered consists of 180×1028 user-item matrix. QoS attributes considered are response time, availability, success ability, reliability and latency. The normalized data is considered for this study.

Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) metrics are used to measure the prediction quality of our method in comparison with other collaborative filtering methods. MAE is defined as:

$$MAE = \frac{\sum_{(u,i) \in T} |r_{ui} - p_{ui}|}{|T|} \quad (9)$$

And RMSE is defined as:

$$RMSE = \sqrt{\frac{\sum_{(u,i) \in T} (r_{ui} + p_{ui})}{|T|}} \quad (10)$$

where, r_{ui} is the expected QoS value of Web service i observed by user u , p_{ui} is the predicted QoS value, T is the testing set and $|T|$ is the number of elements in T . The smaller the MAE and RMSE values mean the better prediction accuracy. Table 3 given below shows

Table 3: Average MAE and RMSE values

	Training matrix density = 90%		Training matrix density = 80%	
	MAE	RMSE	MAE	RMSE
User-based (UPCC)	0.2271	0.5800	0.2384	0.5918
Item-based (IPCC)	0.1741	0.4984	0.2278	0.5850
Slope one	0.1422	0.4437	0.2149	0.5613
Weighted slope one	0.1359	0.4220	0.1288	0.4081
Clustering with slope one	0.1345	0.4290	0.1526	0.4618
Clustering with weighted slope one	0.1288	0.4081	0.1459	0.4392

the average of MAE and RMSE results considering the above mentioned QoS values of different prediction methods.

CONCLUSION

In this study, a hybrid approach using content based and collaborative filtering approach is proposed to predict the QoS values of non-invoked web service and to recommend top k web services to the active users. Clustering Web Services Documents (WSDL documents) into functional similar groups can greatly reduce the search space of a service discovery task. Therefore, it can be seen as a predecessor of web service discovery or an important functionality provided by future service search engines. However, very few researchers have looked into this area. In this study, two approaches for clustering WSDL documents namely Keyword-based approach (non-semantic) and LSA based (semantic) are implemented and the result shows better accuracy with respect cluster quality using semantic method.

In the past collaborative filtering algorithm, as user-item matrix is usually very sparse, many algorithms need to predict a large number of empty values before recommending, increasing the complexity of the algorithm itself. In this study, the improved Slope One algorithm don't have to predict empty values to recommend, but use K-means algorithm to remove noise ratings. It improves predicting accuracy and well solves higher commendation quality under data sparsity. Experiments show that, the improved Slope One algorithm selecting the appropriate value of k, compared with Slope One algorithm and other collaborative filtering algorithms, improves the accuracy. After predicting the QoS values, web services are ranked based on QoS values and the top-k ranked web services are recommended to the active user.

As only numeric QoS values are considered in this study, in future it is decided to test the proposed method on interval data on more dataset and to use probabilistic approach with LSA method for clustering WSDL documents.

REFERENCES

Al-Masri, E. and Q.H. Mahmoud, 2007a. Discovering the best web service. Proceeding of the 16th International Conference on World Wide Web (WWW), pp: 1257-1258.

Al-Masri, E. and Q.H. Mahmoud, 2007b. QoS-based discovery and ranking of web services. Proceeding of the IEEE 16th International Conference on Computer Communications and Networks (ICCCN, 2007), pp: 529-534.

Cardellini, V., E. Casalicchio, V. Grassi and F.L. Presti, 2007. Flow-based service selection for Web service composition supporting multiple QoS classes. Proceeding of the IEEE International Conference on Web Services (ICWS, 2007), pp: 743-750.

Deshpande, M. and G. Karypis, 2004. Item based topn recommendation. ACM T. Inform. Syst., 22(1): 143-177.

El Haddad, J., M. Manouvrier, G. Ramirez and M. Rukoz, 2008. QoS-driven selection of web services for transactional composition. Proceeding of the IEEE International Conference on Web Services (ICWS, 2008), pp: 653-660.

Jiang, Y., J. Liu, M. Tang and X. Liu, 2011. An effective web service recommendation method based on personalized collaborative filtering. Proceeding of the IEEE International Conference on Web Services (ICWS'11), pp: 211-218.

Karta, K., 2005. An investigation on personalized collaborative filtering for web service selection. Honours Thesis, University of Western Australia, Brisbane.

Li, J., L. Sun and J. Wang, 2012. A slope one collaborative filtering recommendation algorithm using uncertain neighbors optimizing. Proceeding of the 2011 International Conference on Web-Age (WAIM, 2011). Springer-Verlag, Berlin, Heidelberg, pp: 160-166.

Liu, L., J. Kang, J. Yu and Z. Wang, 2005. A comparative study on unsupervised feature selection methods for text clustering. Proceeding of the 2005 IEEE International Conference on Natural Language Processing and Knowledge Engineering (IEEE NLP-KE, 2005), pp: 597-601.

Ma, H., I. King and M.R. Lyu, 2007. Effective missing data prediction for collaborative filtering. Proceeding of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR, 2007), pp: 39-46.

Nayak, R. and B. Lee, 2007. Web service discovery with additional semantics and clustering. Proceedings of the IEEE/WIC/ACM International Conference Web Intelligence.

- Papadimitriou, C.H., P. Raghavan, H. Tamaki and S. Vempala, 2000. Latent semantic indexing: A probabilistic analysis. *J. Comput. Syst. Sci.*, 61(2):217-235.
- Sarwar, B., G. Karypis, J. Konstan and J. Riedl, 2001. Item-based collaborative filtering recommendation algorithms. *Proceeding of the 10th International Conference on World Wide Web (WWW'01)*, pp: 285-295.
- Shao, L., J. Zhang, Y. Wei, J. Zhao, B. Xie and H. Mei, 2007. Personalized QoS prediction for web services via collaborative filtering. *Proceeding of the IEEE International Conference on Web Services (ICWS'07)*, pp: 439-446.
- Shi, Y., K. Zhang, B. Liu and L. Cui, 2011. A new QoS prediction approach based on user clustering and regression algorithms. *Proceeding of the IEEE International Conference on Web Services (ICWS'11)*, pp: 726-727.
- Sreenath, R.M. and M.P. Singh, 2003. Agent-based service selection. *J. Web Semant.*, 1(3): 261-279.
- Sun, H., Z. Zheng, J. Chen and M.R. Lyu, 2011. NRCF: A novel collaborative filtering method for service recommendation. *Proceeding of the IEEE International Conference on Web Services (ICWS'11)*, pp: 702-703.
- Wei, L. and W. Wilson, 2009. Web service clustering using text mining techniques. *Int. J. Agent-Oriented Softw. Eng.*, 3(1): 6-26.
- Wu, G., J. Wei, X. Qiao and L. Li, 2007. A bayesian network based QoS assessment model for web services. *Proceeding of the IEEE International Conference on Services Computing (SCC, 2007)*, pp: 498-505.
- Xie, Q., K. Wu, J. Xu, P. He and M. Chen, 2010. Personalized context-aware QoS prediction for web services based on collaborative filtering. *Proceeding of the 6th International Conference on Advanced Data Mining and Applications (ADMA'10). Part II, Springer-Verlag*, pp: 368-375.
- Zeng, L., B. Benatallah, A.H. Ngu, M. Dumas, J. Kalagnanam and H. Chang, 2004. QoS-aware middle ware for web services composition. *IEEE T. Softw. Eng.*, 30(5): 311-327.
- Zheng, Z. and M.R. Lyu, 2008a. WS-DREAM: A distributed reliability assessment mechanism for web services. *Proceeding of the International Conference on Dependable Systems and Networks With FTCS and DCC (DSN, 2008)*, pp: 392-397.
- Zheng, Z. and M.R. Lyu, 2008b. A QoS-aware middleware for fault tolerant web services. *Proceeding of the 19th International Symposium on Software Reliability Engineering (ISSRE, 2008). Seattle, WA*, pp: 97-106.
- Zheng, Z., H. Ma, M.R. Lyu and I. King, 2011. QoS-aware web service recommendation by collaborative filtering. *IEEE T. Serv. Comput.*, 4(2): 140-152.