Middle-East Journal of Scientific Research 24 (S2): 243-247, 2016
ISSN 1990-9233

DOI: 10.5829/idosi.mejsr.2016.24.S2.158

An Effective Approach for Web Services Recommendation Using Adaptive Weighting Algorithm with Temporal Features

[1]Balika J. Chelliah, [2]K. Vivekanandan and [3]N. Saraswathi

[1]Department of Computer Science and Engineering, SRM Univerisity, India
[2]Department of Computer Science and Engineering,
Pondicherry Engineering College, Pondicherry, India
[3]Department of Computer Science, SRM University, Chennai, India

Abstract: A Web service is a method of communication can taken place between two electronic devices over a network.Service Oriented applications are frequently constructed by adapting web services. Web services are recommended based on Quality of Service (QoS) factors that usually depends on network, location of user and web services. In the existing system, Collaborative Filtering (CF) is the one of the common methodemployed for making web services recommendation. However the users ratings are notreflect onservice recommendation. In this paper, we proposed an adaptive weighting algorithm with temporal features for web service recommendation. This method influence the information of user ratings and profile content of web services, a set of features will be designed and dynamically recommended by adaptive weighting method. Dynamic features are extracted using Time Series Analysis (TSA) algorithm. For similar neighbor selection and data modeling, k-means clustering algorithm is to be used.

Key words: Web service・Recommendation・Quality of service (QoS)・Time-aware・Adaptive Weighted

INTRODUCTION

WEB services are integrated software components designed to support interoperable machine-to-machine interaction over a network. By composing web services, large number of Internet applications developed frequently due to the pervasiveness of Service-Oriented Architecture (SOA). As a consequence, Web service has rapidly increased over the last decade. So, the users have challenging task to find web services as their desired and that satisfy their non – functional requirements also [1].

QoS can be defined as set of non-functional properties such as throughput time, response time, success rate, reliability and so on. Typically, a user prefers to select a best QoS performance of Web service. However, web services QoS depends on users' and web services' status and maychanges from one user to another user.

Usually, Web services are recommended based on different data analysis methods, i.e.,content-based, rule-based and collaborative filtering. Among them, collaborative filtering (CF) techniques had been categorized into two approaches: one is neighborhood methods and another is latent factor models. Theneighborhood methods can be classified into user-oriented and item-oriented. They normally use conventional Pearson's Correlation Coefficient (PCC) method to find similar users or similar items on thebasis of co-ratings and predicted based on ratings of the nearest neighbors. Latent factor models (LFM) find latent factorsfrom the pattern of user*item ratings using techniques like matrix factorization. LFM use the factors to compute the usefulness ofitems to users. Previous CF based recommendation method have rarely taken QoS attributes such as response time and throughput time of web services and rating of web service is not accurately predicted due to sparse of data in nature. Moreover, CF approaches usually accounted the cold-startproblem (system cannot identify the exact inferences of user or item) which is amplified in the dynamic scenario since the rate of new users and new items would be higher.

Corresponding Author: Balika J. Chelliah, Department of Computer Science and Engineering, SRM Univerisity, India.

To overcome the above problems, a novel idea has been proposed as,

- Influential factors of QoS attributes (response time, throughput time, success rate) to be taken into an account for evaluating the performance of web 99999services.
- To improve the recommendation, temporal information of users and web services are to be considered.
- For estimating accurate rating of web services, an adaptive weighted algorithm is used.
- K-means clustering is proposed to select top k-nearest neighbor and group them into cluster for QoS prediction.

This paper remainingstructure as follows. Section 2: Related Work. Section 3: Problem Formulation. Section 4: Overview of Proposed Work. Section 5: Result. Section 6: Conclusion and Future Work and at last ended by Reference.

Related Work: Recommendation Systems are used to provide most suitable services to the users as their specific interests. The QoS factors influence the performance of web services.In general, web services may provide either good or bad QoS to users. Since to recommend the services to users these parameters to be consider while implementing recommendation concept.Varioustechniques of recommendation such as content-based, link prediction-based, CF based and Adaptive based have been applied to web service recommendation.

Collaborative Filtering: Among them, collaborative filtering (CF) techniques had been categorized into two approaches: one is neighborhood methods and another is latent factor models. The neighborhood methods can be classified into user-oriented and item-oriented. J.S. Breese *et al*. [2] proposed web services recommendation system based on user preference via Collaborative filtering and also predicting the similarity between users. B. Sarwar *et al* [3] implemented item based CF algorithm, similarity computation done between items by using conventional Pearson Correlation Coefficient. Weighted sum and regression model are applied for recommendation. M.R. McLaughlin and J.L. Herlocker [4] had proven that algorithm discover the users having different item with ranking. It has interests only related to the active user and it discovers items ranked by the active user that are related to the item being expected. SongJieGong *et al*. [5] Adapted recommendation systems combines the user clustering technology and item clustering technology. The user can discover related items and narrow down their specific interests among the existing items. Z. Maamar *et al*. [6] have worked on context. Web service personalization is developed by using the contexts. Context is the information that characterizes the associations among applications, people and location. Preferences are dissimilar types. Preferences are measured based on performance of Web services when it starts and ends.

Adaptive Algorithm: Recently, Adaptive based has more attention for its simplicity and effectiveness. Jingjing Chen *et al* [7] implemented User oriented adaptive algorithm based on Support Vector Machine to meet personalized needs of customers in E-Supermarket service system.User profiles are organized hierarchically and then Vector Space Model is used for recommendation. Cai Chen and Daniel Zeng [8] focused item based hybrid algorithm that uses a dynamic user adaptive combination strategy. The hybrid algorithm combines the similarity of item based and content based algorithm for similarity computation. Then the Hybrid similarity is performed for item recommendation to users.Jonghyun *et al* [9] investigated a context-dependent search engine that represents user context in a knowledge-based context model, implemented in a hierarchical structure with granularity information. Search results are ordered based on semantic relevance computed as similarity between the current context and tags of search results. Alexander *et al* [10] had worked adaptive services recommendation over time based on user ratings in order to meet user requirements. It supports search algorithm in decision making.They have combined the automatic service composition with this to provide dynamic market services.

Problem Formulation: Collaborative filtering is a method of making automatic predictions about a user interest by collecting user preferences or most wanted information from many users. Formally, a CF domain consists of a set of users U, a set of items I and users' ratings on items. The last is often represented by a user-item matrix R, where each entry $r(x,y)$ (,) represents user x's rating on item y.

Table 1: User Item Rating Matrix.

	User1	User2	User3	User4	User5
Item1	5	4	3	4	5
Item2	4	2	4	2	2
Item3	3	2	4	3	1

In CF, temporal features of users' rating are not extracted. Moreover, it cannot find the exact co-relation in between users and items. The proposed system has solution to solve above problems. To recommend the best services to users based on the following requirements,

- Profile content of web services are included.
- User's specific interest and their rating's on item history are taken as input.
- Temporal Information of users and web services has been taken into an account.

Proposed Work: The proposed work focuses multiple phases of user preferences on web services for recommendation. In the proposed system, an adaptive weighted algorithm is used.

System Architecture:

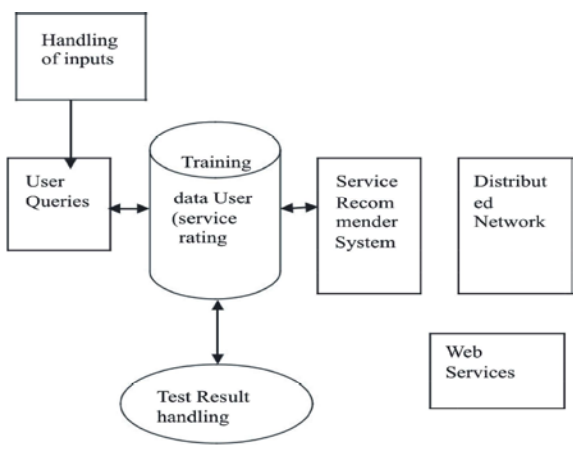

Fig. 1: WebServicerecommendationmethod

The system architecture diagram describes the flow of entire process.

Multiple Phase of User Preference: Users have different interest on web services. In consideration of computation, flexibility and accuracy, a set of dynamic features are defined to describe about users' multi-phase preferences. To learn weights of all ratings for each user is not possible, but it is possible to learn the general weights of ratings in the user's different phases of interest, if the phases include ranges of time that are long enough.

Relation Mining Of Rating Value: For the sparsity of recommendation data, it is difficult to capture users' dynamic preferences, due to lack of valuable information, that may come from the following sources – user profiles, item profiles and historical rating records. Traditional algorithms deeply rely on the co-rate relation (to the same item by different users or to different items by the same user), which is rarely taken when the data is sparse. By using co-rate relation, useful ratings are discovered. Based on QoS, ratings are given. In this proposed method, QoS parameters - response time, throughput time, success rate are calculated for each web services.

Group Similar Users: This module finds users who are similar to the active user by considering users' preference about web services. This proposed method consideruser locations also for similar user selection and stored into the database-means algorithm is used to select top n users and group them.

Dynamic Feature Extraction: Past rating history should impact the predictive features less and have lesser weight. Moreover, Users' preferences or items' reputations are migrating, thus we have to deal with the dynamic nature of data to enhance the correctness of recommendation algorithms. So,recent ratings and remote ratings with the time interval (t1,t2) should have different weights in the prediction. These methods help to make progress in precision of dynamic recommendation.

Dynamic Recommendation: In this module Adaptive Weighted Algorithm is used for large datasets of multiple users. Datasets contain information about user preferences of certain items. For example, Users can rate a web services by assigning a number of stars. The number of stars is a user's preference value, a value from 1 to 5. Based on this cluster of personal preferences of user and a 'similarity function' can recommend the service to users or determine similar users, users with similar thought about service.

Adaptive Weighting Algorithm: The proposed algorithm adapts the second order information for accurate estimation of user rating.

Pseudocode: Define: U ? Set of users. Define: I?Set of web services. Require: R is the rating matrix For each user input Ik do

Select rating for all Ik. with time interval<t1,t2>. Generate rating subset Rs. {Rs =0,1,2,3…}. Estimate average weight wl for each Rs

Extract dynamic features feas,p?Rs. If(feas,p>wl) then

Recommended the service to user. Else

Not recommended. End.

Steps:

- Define, U={u1,u2,…un} is a set of web service users and I={i1,i2,…in} web service items in time <t1,t2>are contributed.
- Users' rating has been given as input,R is a user-item rating matrix, where r(u,i) represents a
- For each web service item i, in the time interval <t1, t2>, rating subset Rs defined. Divide the Rs into various disjoint secondary subsetRs^p= {Rj,kERswith Time Tj,k} in order to extract dynamic feature applying TSA (Time Series Analysis) algorithm is the fundamental feature extraction method. TSA predict features with in time range and will used for further analysis. TSA algorithm applied on a secondary subset of $Rs^p$to get a feature feas,p, there would be a uniform formulation as,

$$fea_{s,p} = \sum_{l=1}^{o} \frac{\mathrm{wl}}{\mathrm{w}} \mathrm{R}_{\mathrm{s,l}}^{\mathrm{p}},$$

where $\#Rs^p$ = o, $\#Rs,l^p$(l = 1, 2,.. ., o) are the rating values which are from the subset Rs^d and listed in reversed time order. And positive weight parameters wl, (l = 1, 2,.. ., o) and normalization factor w should satisfy. Subsets are updated frequently with in the time interval<t1,t2>.

As features like *fea s, p*(*s* = 1, 2,.. ., *p*= 1, 2,.. .) and divisions of time gained by applying *Multiple Phase Division* are all normalized rating values, it is convenient to organize them for accurate rating estimation by adaptive weighting. Sizes of the relevant subsets are also recorded in MPD and could reflect data density. We incorporate these features for recommendation with a linear model since they are homogeneous and it is efficient to learn their weights.

- Rj,k is used to note the estimated rating that user uj could give to item ik at time point Tj,k and the adaptive linear model can be formulated as:

$$R_{j,k} = \sum_{3} \sum_{p} (\propto_{s,p} + \beta(\neq R_s^p) bu_j(s) bi_k(s) fea_{s,p})$$

with: $\propto_{s,p} \geq o, \beta \geq o,$

where sizes of relevant subsets are used as prior information in weighting the features to improve recommendation accuracy, feas,p(s = 1, 2,..., d = 1, 2,.. .) are the features. Rs^p (s = 1, 2,.. ., d = 1, 2,...) denote their relevant secondary rating subsets, buj and bik are binary functions denoting the relating state of candidate rating and relevant subset and ás,dand â are weighting factors which should balance the weights of features and data density, or, maintain a balance of the affection of data consistency and quantity of information.

- Adapt the estimated rating for dynamic recommendation of services.
- Update the database for consecutive access of users.

RESULT

The proposed system will predict accurate estimation of web services. It recommends to user with what their exact needs. Personalized recommendation of services done to each users with requirement of temporal features. This proposed sytem dynamically adapts the web service weightage based recent rating value. Hence recommendation system is highly efficient and cost effective.

Conclusion and Future Work: In this paper, we proposed a novel adaptive weighted recommendation algorithm based on users' history ratings. Multiple phases of user preference provide a dynamic personalized recommendation for existing users. A set of dynamic features are extracted by using TSA algorithm to predict weightage for each web service. For accurate rating estimation, adaptively weighted recommendation is done based on feature extracted. The concept of recommending the web services is the way the user identified the items and how they rate items. This proposed algorithm is highly effective due to dynamic adaptive of recent user rating information. This algorithm uses temporal information in order to accurately predict user needs and recommended the services as their desired. In future two more QoS parameters can be included to predict QoS value for web services. Location information also consider for similarity computation.

REFERENCES

1. Zhang, L.J., J. Zhang and H. Cai, 2007. Service Computing. Springer and Tsinghua University Press.

2. Breese, J.S., D. Heckerman and C. Kadie, 1998. Empirical analysis of predic-tive algorithms for collaborative filtering, in Proc. of the UAI, pp: 43-52.
3. Sarwar, B., G. Karypis, J. Konstan and J. Riedl, 2001. Item-Based Collabora-tive Filtering Recommendation Algorithms, in Proc. 10th Int'l Conf. World Wide Web, pp: 285-295.
4. McLaughlin, M.R. and J.L. Herlocker, 2004. A collaborative filtering algo-rithm and evaluation metric that accurately model the user experience, in SIGIR.
5. Ungar, L.H. and D.P. Foster, 1998. Clustering Methods for Collaborative Filtering, Proc. AAAI Workshop Recommendation Systems.
6. Maamar, Z., S.K. Mostefaoui, and Q.H. Mahmoud, 2005. Context for Personalized Web Services, Proc. 38th Ann. Hawaii Int'l Conf., pp: 166b-166b.
7. Jingjing Chen and Qi Luo, 2006. Research on Adaptive Recommendation Algorithm in Personalized E-Supermarket Service System, in ICSP 2006.
8. Cai Chen and Daniel Zeng, 2012. A Dynamic User Adaptive Combination Strategy for Hybrid Movie Recommendation, in IEEE, 2012.
9. Jonghyun Hana, Hedda R. Schmidtkeb, Xing Xiec and Woontack Wood, 2013. Adaptive content recommendation for mobile users: Ordering recommendations using a hierarchical context model with granularity, in Science Direct, 15 Nov 2013.
10. Alexander Jungmann, Bernd Kleinjohann and Felix Mohr, 2014. Combining Automatic Service Composition with Adaptive Service Recommendation for Dynamic Markets of Services, in IEEE 10th World Congress on Services, 2014.