IJCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Vol.-6, Issue-12, Dec. 2018 E-ISSN: 2347-2693

Fusion of Pearson Similarity and Slope One Methods for QoS Prediction for Web Services

G. Vadivelou[1*], E. Ilavarasan[2]

[1]Dept. of Computer Science, Kanchi Mamunivar Centre for PG Studies, Puducherry, India
[2] Dept. of Science and Engineering, Pondicherry Engineering College, Puducherry, India

[*]Corresponding Author: gvadivelou@gmail.com, Tel.: +00-9345458593

Available online at: www.ijcseonline.org

Accepted: 05/Dec/2018, Published: 31/Dec/2018

Abstract— Web services have become the primary source for constructing software system over Internet. The quality of whole system greatly dependents on the QoS of single Web service, so QoS information is an important indicator for service selection. In reality, QoS 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 paper attempts to settle this difficulty by fusing Pearson similarity and Slope One methods for QoS prediction. In this paper, the Pearson similarity is adopted between two services as the weight of their deviation. Meanwhile, some strategies like weight adjustment and SPC-based smoothing are also utilized for reducing prediction error. In order to evaluate the validity of the proposed algorithm, comparative experiments are performed on the real-world data set. The result shows that the proposed algorithm exhibits better prediction precision than both basic Slope One and the well-known WsRec algorithm in most cases. Meanwhile, the new approach has the strong ability of reducing the impact of noise data.

Keywords— Web services, QoS prediction, Slope One, similarity, collaborative filtering

I. INTRODUCTION

In recent years, the pattern of service-oriented computing (SOC) has been widely accepted to build large-scale sys- tem over Internet [1]. In this new style of software development paradigm, software is no longer built via the traditional process, but in the way of service unit reuse. Accordingly, some new problems such as service discovery, selection and composition are emerging, and play a great impact on the quality of service-based system. In general, service unit is self-describing component to complete a specific task. Quality-of-Service (QoS) is an important way to describe non-functional characteristics of Web services. When several functionally-equivalent Web services exist in the network, QoS is viewed as a critical issue for picking out the appropriate service from equivalent service set. Web service QoS usually includes a number of properties, such as response time, throughput, failure probability, availability, price, popularity, and so on [2]. Due to different network environments, service users will have different QoS metrics for the same Web service. Therefore, each service user has to understand QoSs of all services to be invoked at his/her end.

In order to construct the software meeting the actual requirements, it needs to make the existing service units work together in accordance with the pre-defined business logic, that is the so-called Web service composition (WSC).

During service selection, the quality of each service unit should be carefully considered so as to ensure the trustworthiness of WSC. However, service invoker may be lack of adequate historical information for some specific Web services. He/She has to estimate the QoS value of a given Web service before determining to introduce it into WSC, i.e., QoS prediction for Web services. Since the service user has not even invoked the service in past, the estimation for such service's QoS has to get help from other similar users or self's invocation records on other Web services.

The similar work firstly emerged in the field of E-commerce, vendors used consumer's historical purchase records and the similarity between costumers to recommend products [3]. In contrast, the prediction of Web service's QoS is much harder than product recommendation. Web service is merely an encapsulated and distributed Web API over network. Therefore, for service users, the information related with service execution are hardly collected. In order to improve the prediction precision, the limited available Web services invocation records should be fully utilized.

It is important to note that most of the researchers have used Pearson-based similarity. Although this kind of similarity can provide good prediction effect, it not only cost much computation time but also lose performance for the very sparse data set. Besides the similarity-based collaborative filtering, Slope One [9] has been validated as an effective prediction method due to its simpleness and high performance. In this paper, a hybrid QoS prediction method is proposed through introducing Pearson-based similarity into Slope One method. The experimental results revealed that the proposed hybrid method could outperform the basic Slope One and Pearson-based collaborative filtering methods in term of prediction precision.

The main contributions of this paper are as follows:

1. A prediction algorithm of Slope One co-operated with Pearson similarity measurement has been proposed for providing QoS information for Web service user.
2. Some strategies like weight adjustment and SPC-based smoothing are presented for improving the prediction precision.
3. The detailed performance analysis on real-world data set is performed to verify the effectiveness of the proposed method. Moreover, the two-stage filling strategy is also validated through experimental analysis.

The structure of the paper is as follows. In the next section, we state the QoS prediction problem for Web services, and introduce two typical collaborative filtering algorithms. Section III gives some existing researches that are closely related with the proposed prediction approach. In Section IV, the overall QoS prediction framework is firstly addressed, and then the proposed Slope One algorithm is described in details. The performance comparison and analysis are discussed in Section V. Finally, Section VI concludes the paper.

II. BACKGROUND

A. QoS prediction for Web services

When Web service users prepare to adopt some service units to construct an enterprise-level application, in general, they have to replace each abstract service in service orchestration plan with a concrete service. For each abstract service, perhaps quite a few service implementations will meet the requirement of its function. Therefore, the rational way is to pick out a service with high QoS from the candidate set. However, for a specific service user, the QoS values of some Web services may be not available. As a consequence, it is necessary to estimate the QoSs of such services according to the limited existing information that is so-called QoS prediction problem. With regard to prediction techniques, experiences tell us that collaborative filtering (CF) techniques can be viewed as a good choice.

B. Review on collaborative filtering

In general, collaborative filtering is a technique of suggesting particularly interesting items or patterns based on past evaluations of a large group of users. The fundamental assumption of CF is that if users have similar tastes on some items, and hence they will rate or act on other items similarly. At present, CF techniques can be classified into three categories [10, 11]: (1) memory-based methods, (2) model-based methods, and (3) hybrid methods. Memory-based CF utilizes the user rating data to calculate the similarity or weight between users or items, and then make predictions according to those similarity values. This type of CF is the earlier mechanism and used in many commercial systems such as Amazon, Barnes and Noble. According to the background and feature of QoS prediction problem, memory-based CF is treated as the main research issue in the paper. Especially, two well- known methods, i.e., Pearson correlation CF and Slope One approach, are taken into consideration

C. Pearson correlation based method

In a typical CF scenario, there is a list of m users

$\{u1, u2, \cdots, um\}$ and a list of n items $\{i1, i2, \cdots, in\}$, and

each user ui has a list of items (i.e., $I\ ui$), which the user has rated, or about which their preferences have been inferred through their behaviors [10]. Generally speaking, the basic procedure of CF-based recommendation or prediction can be summarized as the following two steps:

(1) Look for users sharing the similar interests or rating patterns with a given user (called active user).
(2) Use the information from those like-minded users found in step (1) to calculate a prediction for the active user.

Here, we mainly address the case from the perspective of users, but the above process is also suitable for item-oriented analysis. It is not hard to find that, how to find the similar users (or items) for a specific user (or item) is a critical task in the whole process of CF. In practice, the common interests or patterns are expressed via the correlation between users (or items).

At present, Pearson correlation coefficient has been introduces for computing similarity between users or items according to the user-item data like in Table 1, which is usually called user-item matrix. For two given users a and u, their similarity can be computed as follows.

$$Sim(a,u) = \frac{\sum_{i\in I}(r_{a,i} - \bar{r}_a)(r_{u,i} - \bar{r}_u)}{\sqrt{\sum_{i\in I}(r_{a,i} - \bar{r}_a)^2(r_{u,i} - \bar{r}_u)^2}} \tag{1}$$

where $I = I_a$ n I_u is the subset of items which both user a and u have invoked previously, $r_{a,i}$ is a vector of item I observed (or rated) by user a, and $\bar{r}_a$ and $\bar{r}_u$ represent average values of different items observed (or rated) by user a and u, respectively.

The prediction method based on two users' similarity is referred as user-based CF. Similarly, CF can also be conducted through the similarity computation between two items, that is, item-based *CF*. According to the studies from other researchers, item-based CF can outperform user-based CF in most conditions, and has been treated as a preferred choice for prediction or recommendation problems

As mentioned earlier, Shao *et al*. adopted Pearson correlation-based CF for Web services' QoS prediction [4]. Recently, Zheng *et al*. improved prediction precision problem through combining item-based and user-based CF together

[5]. Their WsRec algorithm exhibits better performance than other basic prediction methods, and has caused much attention in these two years.

D. Slope One method

Although previous studies have revealed that Pearson scheme CF can gain good prediction precision, its performance is not so satisfactory for the case of extremely sparse data. Meanwhile, Pearson-based method will cost a lot of computational overhead to measure the similarity between users or items. Fortunately, another well-known method called Slope One [9] can make up such deficiencies. On the one hand, Slope One can show good prediction effect for sparse data. On the other hand, this method can perform prediction activity with less computing cost.

As stated by Lemire *et al*., Slope One algorithm works on the intuitive principle of a "popularity differential" between items for users. In this algorithm, how much better one item is liked than another is determined in a pair-wise fashion. Firstly, the difference between the averages of two items can be calculated via subtract operation. Then, once one item's value is available, the other's value can be predicted according to such difference. For two users (*a* and *b*) and two items (*i and j*) in user-item matrix, the values of these two items for user *a* are known and the differential from *i* to *j* is 1.5-1=0.5. Thus, the item *j*'s value for user *b* can be predicted via this mapping relationship, that is, 2+(1.5-1)=2.5. Of course, many such differentials exist in a training set for each unknown rating, the average of these differentials will be taken for predication.

Formally speaking, for a given user-item matrix, the set of the users who contain rating records both on item *i* and item *j* can be computed and denoted as $U_{i,j}$ here. Obviously, $U_{i,j} = U_{j,i}$. Then, the average deviation of item *i* with respect to item *j* can be denoted as:

$$dev_{j,i} = \sum_{u \in U_{j,i}} \frac{r_{u,j} - r_{u,i}}{card(U_{j,i})} \quad (2)$$

where $card(U_{j,i})$ returns the element number of set $U_{j,i}$. Based on the deviations of items, the rating of user *u* for item *j*, i.e. $r_u j$, can be predicated via the following way.

$$P(r_{u,j}) = \frac{1}{card(R_j)} \sum_{i \in R_j} (dev_{j,i} + r_{u,i}) \quad (3)$$

where $R_j = \{i | r_{u,i} \neq NA, i \neq j \text{ and } card(U_{j,i}) > 0\}$ is the set of items which have co-occurrence relationship with item *j*.

The above discussion belongs to user-oriented prediction. Obviously, Slope One method can also be used in the other style, i.e., item-oriented prediction. In addition, several kinds of extensions are proposed. For instance, single or bivariate regression is used for finding the best mapping relation [12, 13], bi-polar strategy is used for users' two different attitudes

[9]. However, variant algorithms can't lead to obvious improvements over the basic form in all cases.

III. Related Work

From the perspective of service users, how to select a suitable service is a critical step to build a reliable software system. In general, service selection is mainly in accordance with the property of QoS. Accordingly, QoS prediction for Web services has caused widespread attention in the field of service computing.

As we mentioned earlier, Pearson correlation-based algorithms are the main-stream strategies to treat such problem at current stage. Shao *et al*. [4] 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 data set in small scale, and the error analysis is not so sufficient. Subsequently, Zheng *et al*. [5] firstly 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 [7]. However, it is not so easy to obtain such personal information, so the application of their method is limited. Reference [15] 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. [16] 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. Reference [6] presented a personalized context-aware QoS prediction method based on the Slope One approach. In this work, the basic Slope One algorithm is used for prediction, but it has been validated to be not very precise in the proposed experiments. Then, Li *et al*. [17] 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 aims at the rating prediction problem. On the other hand, Bi-Polar phenomenon maybe exists in the data set in rating style, but not obvious in QoS data (i.e. the continous data type)

With regard to the combination of Slope One and Pearson similarity, the preliminary researches in [18] and [19] have contributed an incipient idea for blending them together. However, the above works merely provide a primitive form of similarity- aware Slope One prediction algorithm, that is, the case of λ=1 in the proposed work. As shown in our experimental results, this basic form without weight adjustment is not very effective for QoS prediction problem. At the same time, the experimental analysis and discussion are very limited in their work. Besides the weight adjustment strategy illustrated in formula (7), here, a more important strategy named SPC-based smoothing is also proposed to reduce prediction error.

IV. METHODOLOGY

With regard to the usage scenario of Web services, services' QoS data from different users can form a sparse matrix of service invocation records. In order to help service user make a rational decision about service selection, the prediction for a specific service's QoS w.r.t. of the current user is very necessary. In this paper, we provide a hybrid prediction method through comprehensively adopt the merits both from Pearson correlation-based algorithm and Slope One algorithm.

i. The Proposed Prediction

For an active service user *u*, the number of services which have been invoked by *u* is named *given number* (i.e. *GN*). For all *n* service items, *GN* is usually a little part. In order to provide precise QoS estimations for the remaining service items w.r.t user *u*, we should take full use of other users' invocation records for these services. Here, we assume the historical QoS data about *m* users for *n* service items is matrix *M*. Similarly, each service user only has partial QoS information in that matrix. The proportion of existing QoS data in matrix is denoted as *density* (*d* for short).

In our investigations on collaborative filtering techniques, we have found a fact as follows: Slope One method is suitable for the very sparse data set (i.e. very low density data), whereas Pearson-based CF can achieve desired prediction results for the case of high density data. Therefore, in our method, we mainly adopt Slope One method for prediction and compute Pearson correlation between services to adjust the reference weight. The closer relation between a service and the subject service for user *u*, the higher weight should be assigned to the QoS deviation between these two services.

The whole procedure of Web service QoS prediction is shown in Fig 1. At the initial stage, the historical QoS records of *n* Web services for *m* users can be collected. Here, we call it training data *M*. In general, a service user could not have QoS records for all *n* services, and usually has only very limited ones of them. As a result, training data is a sparse matrix in real-world scenarios. The matrix *M* should be filled as full as possible so that it can provide more useful information for QoS prediction. In the second step, we present a similarity-aware Slope One as a way to fill the 'NA' (a.k.a. *null*) records in the training data set. For the perspective of Web service execution, there may be exist some abnormal QoS records in the above training data, especially for the QoS attribute with wide scale values. In order to handle this problem, in the third step, we adopt *statistical process control* (SPC) strategy to adjust such exception data.

Based on the above treatments, the training data set has been enhanced and its data density has a great promotion. According to the renewed training matrix, our algorithm is also utilized for predicting Web service's QoS for active user. Finally, prediction quality is measured via error analysis.

ii. Prediction Methods

With regard to QoS prediction framework, it is not hard to find that the propsoed algorithm and SPC-based adjustment strategy play important roles for improving the precision. The details of these two key algorithms are addressed as follows.

As mentioned before, Slope One-based CF exhibits its advantage for sparse data. Since each active user has only GN (usually GN << n) QoS records for n Web services, we adopt item-oriented Slope One method to predict QoS value for active user. However, the similarity between items is not taken into consideration in the basic Slope One prediction method. In our work, we introduce the similarity between two items into Slope One method to form a new QoS prediction algorithm for Web services. The basic idea is that, the service with the higher similarity should give the higher priority when considering the deviation in Slope One method. Here, we adopt item-based Pearson correlation to measure the similarity between two Web services. For service i and j, theirs similarity can be calculated as follows:

$$Sim(i,j) = \frac{\sum_{u\in U}(r_{u,i} - \bar{r}_i)((r_{u,j} - \bar{r}_j)}{\sqrt{\sum_{u\in U}(r_{u,i} - \bar{r}_i)^2}\sqrt{\sum_{u\in U}(r_{u,j} - \bar{r}_j)^2}} \quad (4)$$

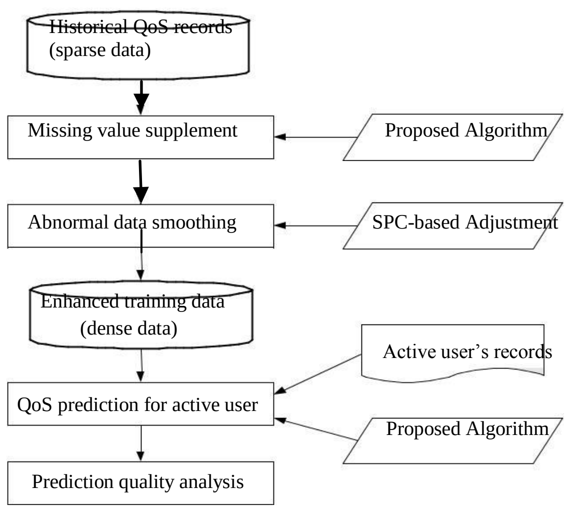

Fig 1: Proposed Framework for QoS Prediction using Slope One method

whee $U = U_i \cap U_j$ is the subset of users who have QoS records both on service i and service j previously (i.e., identical to $U_{i,j}$ in equation (2)), and $_i$ represents the average QoS value of service i observed by different users.

To predict a missing value $r_{u,j}$ in the user-item matrix, we have to measure the similarities between j and other services invoked by user u, that is, I_u-{j}. After removing the services with negative similarity to service j from them, the remaining items are called the related services of j w.r.t. user u, denoted as R(j|u). It can be formally expressed as follows:

$$R(j|u) = \{ I \mid i\varepsilon I_u , Sim(i,j) > 0, i\neq j\} \quad (5)$$

Then, we give the prediction formula based on similarity-aware Slope One algorithm as below.

$$P\left(r_{u,j}\right) = \frac{1}{card(R(j|U))} + \sum_{i\in R(j|u)}(w_{ij}\,.\,dev_{j,i} + r_{u,i}) \quad (6)$$

where $w_{i,j}$ is an adjustment weight in accordance with the similarity between j and another service i (i €R(j|u)). As for a further comment, $w_{i,j}$ can be computed by the following formula:

$$w_{i,j} = \frac{Sim^{\lambda}(i,j)}{\sum_{k\varepsilon R(j|u)} Sim^{\lambda}(k,j)} \quad (7)$$

Here, we denote the original service's QoS record matrix as M, and call the intermediate matrix after filling the missing values as M^t. On the one hand, some exceptional QoS records for Web services perhaps exist in M. The so-called exceptional (or abnormal) record, means the QoS value of a specific user is far away from the records of neighbor users.

On the other hand, it is also very sparse in the original state. As a result, the filled matrix M^t may contain some QoS items which are far from the common situation. Obviously, these abnormal records will cause bad influence on the next stage of prediction. Thus, we should identify them out from matrix M0 firstly, and then smooth them via a heuristic strategy.

In the paper, we borrow the idea from statistical process control (SPC) [14] to tackle the abnormal QoS data in M^t. SPC is a real-time monitoring technique for the process of industrial production in the way of statistical analysis. It can scientifically distinguish the exceptional fluctuation from the normal random fluctuation, so it is used for providing early warning for production process to manager.

We mainly utilize this technique to pick out the abnormal QoS values so as to achieve better prediction performance. At first, for matrix M^t, we judge whether item $r_{u,i}$ (i.e., the QoS of service i for user u) is an exception or not according to the following rule.

$$isAbn(r_{u,i}) = \begin{cases} true, & \mu_i - \theta\cdot\sigma_i < r_{u,i} < \mu_i + \theta\cdot\sigma_i \\ false, & \text{otherwise} \end{cases} \quad (8)$$

where μ_i is the average QoS value of service i (1” i ” n), and ı$_i$ is the standard deviation of service i's QoS records from different users. is a positive integer used for regulating the normal range of QoS value. It is usually set to 3 in most applications of SPC.

When a suspected record of abnormal QoS is detected through the above approach, this isolated item should be smoothed before the prediction step. Here, we introduce a strategy called "small amplitude shift "for smoothing treatment. Suppose $r_{u,i}$ is an abnormal issue according to judgement of equation (8), the smoothing action can be performed via the following formula. The value after adjustment is denoted as $\overline{r}_{u,i}$

$$\overline{r}_{u,i} = \begin{cases} \mu_i - \theta.\sigma_i\ , & r_{u,i} < \mu_i - \theta.\sigma_i \\ \mu_i + \theta.\sigma_i\ , & r_{u,i} > \mu_i + \theta.\sigma_i \\ r_{u,i}\ , & otherwise \end{cases} \tag{9}$$

That is to say, we use the upper (or lower) limit to replace the usually high (or low) QoS record, respectively.

Other particular settings of WSRec algorithm are in accordance with reference [5]. For QoS attribute response time (RT) and throughput (TP), the comparisons on three algorithms is performed respectively. In the experiments, we repeated 75 times for each case of density and GN value, and reported the average NMAE metrics.

V. RESULTS AND DISCUSSION

The experimental results (i.e. NMAEs) on QoS attribute response time (RT) are shown in Table 1. It is clear that the proposed algorithm can outperform WsRec and basic Slope One algorithm for most cases. For the de nsity values (i.e. 10% and 15%), the proposed algorithm (λ=3) can achieve the lowest NMSE for nearly all cases except of density=15% and GN =20. On the whole, the proposed method's performance is better than those of WsRec and basic Slope One for almost all situations, especially λ=2 or 3.

The NMAE values of three algorithms on QoS attribute throughput (TP) are shown in Table 2. It is not hard to find the proposed algorithm (λ=3) has obvious improvement both for WsRec and basic Slope One, except of the case of density=15% and GN =20. With regard to proposed algorithm itself, the predication error of the proposed method reduces with the increase of λ value. When λ reaches to 2, the proposed algorithm outperforms other two algorithms in most conditions.

According to the above experimental analysis, we draw a conclusion that the proposed algorithm is a better choice than WsRec and basic Slope One for service's QoS prediction, especially when the data density of user- service record matrix is low.

TABLE 1: NMAE VALUES FOR SLOPE ONE, WsRec AND PROPOSED ALGORITHMS FOR THE QoS ATTRIBUTE RT

ALGORITHM		*d=10%*			*d=15%*		
		GN=5	GN=10	GN=20	GN=5	GN=10	GN=20
Slope One		0.5980	0.5819	0.5701	0.5928	0.5798	0.5581
WsRec		0.5941	0.5776	0.5540	0.5702	0.5542	**0.5121**
Proposed Algorithm	λ=1	0.5889	0.5731	0.5498	0.5635	0.5496	0.5410
	λ=2	0.5820	0.5645	0.5457	0.5582	0.5403	0.5322
	λ=3	**0.5797**	**0.5611**	**0.5422**	**0.5553**	**0.5311**	0.5277

TABLE 2: NMAE VALUES FOR SLOPE ONE, WsRec AND PROPOSED ALGORITHMS FOR THE QoS ATTRIBUTE TP

ALGORITHM		*d=10%*			*d=15%*		
		GN=5	GN=10	GN=20	GN=5	GN=10	GN=20
Slope One		0.4878	0.4795	0.4757	0.4598	0.4696	0.4604
WsRec		0.4782	0.4743	0.4612	0.4403	0.4489	**0.4401**
Proposed Algorithm	λ=1	0.4769	0.4755	0.4592	0.4484	0.4601	0.4476
	λ=2	0.4711	0.4643	0.4529	0.4421	0.4554	0.4398
	λ=3	**0.4682**	**0.4605**	**0.4382**	**0.4373**	**0.4479**	0.4323

VI. CONCLUSION AND FUTURE SCOPE

With the widespread application of service computing, Web services have been viewed as a prevalent form of components for building software on the Web. In order to ensure the reliability and trustworthy of the composite software system, users generally are very concerned about the quality of service. Unfortunately, the QoS metrics of some services cannot be provided due to the actual situation. Therefore, how to predicate QoS of Web service becomes a valuable task in the field of service engineering.

In the paper, we introduce the Pearson similarity between Web services into Slope One collaborative filtering for solving QoS prediction problem. Instead of assigning the identical weight to each service, we adjust Pearson similarity as a weight for differentiating the deviation between services. In order to improve the prediction accuracy, a SPC-based smoothing is presented for correcting the exceptional data. In the empirical aspects, besides our approach, the basic Slope One and the well-known WsRec algorithm are also implemented. Meanwhile, the comparative analysis is also performed on the public published data set. The experimental results indicate that our hybrid algorithm (the proposed) outperforms other two methods in the term of prediction precision. The SPC-based smoothing strategy can effectively handle the noise data so as to reduce prediction error. Furthermore, an additional strategy called two- stage filling is studied, and the appropriate boundary point for transforming filling methods is also suggested here.

The practice of the proposed algorithm is obvious; it can guide users to pick out desired services from cloud platform. At the same time, this algorithm can also be used in the field of E-commerce to help consumers choose goods. Of course, although the proposed approach achieves some promising results at present, there are still quite a few complicated issues should be further investigated. For instance, the QoS predictions for Web services from the dynamic perspective, as well as the service quality prediction in the environment of mobile computing have to be focused. In addition, to find more effective data filling algorithm for training data is an interesting research direction.

REFERENCES

[1] M. P. Papazoglou and D. Georgakopoulos, (2003). Serive-Oriented Computing, Communcications of the ACM, Vol. 46, No. 10, ACM Press, pp. 25–65.

[2] D. A. Menasce, (2002). QoS issues in Web services, IEEE Internet Computing, Vol. 6, No. 6, IEEE CS Press, pp. 72–75.

[3] J. B. Schafer, J. Konstan, and J. Riedi, (1999). Recommender Systems in E-Commerce, Proc. of the 1st ACM Conference on Electronic Commerce (EC'09), ACM Press, Denver, CO, USA, pp. 158–166.

[4] L. Shao, J. Zhang, Yong Wei, and et al., (2007). Personalized QoS Prediction forWeb Services via Collaborative Filtering, Proc. of the IEEE International Conference on Web Services (ICWS'07), IEEE CS Press, Salt Lake City, Utah, USA, pp. 439–446.

[5] Z. Zheng, H. Ma, M. R. Lyu, and I. King, (2011). QoS-Aware Web Service Recommendation by Collaborative Filtering, IEEE Trans. on Services Computing, Vol.4, No. 2, IEEE CS Press, pp. 140– 152.

[6] Q. Xie, K. Wu, J. Xu, and et al., (2010). Person- alized Context-Aware QoS Prediction for Web Services Based on Collaborative Filtering, Proc. of the 6th International Conference on Advanced Data Min- ing and Applications (ADMA'10), Part II, Springer-Verlag Berlin, Chongqing, China, pp. 368–375.

[7] Y. Jiang, J. Liu, M. Tang, and X. Liu, (2011). An Effective Web Service Recommendation Method based on Personalized Collaborative Filtering, Proc. of IEEE International Conference on Web Services (ICWS'11), IEEE CS Press, Washington, DC, USA, pp. 211–218.

[8] A. K. Menon, K. P. Chitrapura, S. Garg, and et al., (2011). Response Prediction using Collaborative Filtering with Hierarchies and Side-information, Proc. of the 17th ACM SIGKDD International Con- ference on Knowledge Discovery and Data Mining (KDD'11), ACM Press, San Diego, CA, USA, pp.141–149.

[9] D. Lemire and A. Maclachlan, (2005). Slope One Pre- dictors for Online Rating-Based Collaborative Filtering, Proc. of the 2005 SIAM International Data Min ing Conference (SDM'05), Newport Beach, Califor- nia, USA, pp. 1–5.

[10] X. Su and T. M. Khoshgoftaar, (2009). A Survey of Collaborative Filtering Techniques, Advances in Ar- tificial Intelligence, Hindawi Publishing Corporation, pp. 1–19.

[11] Linyuan Lü, Matéš Medo, Chi Ho Yeung, and et al., (2012). Recommender Systems, Physics Reports, Vol. 519, No. 1, Elsevier B. V., pp. 1–49.

[12] S. Vucetic and Z. Obradovic, (2000). A Regression- based Approach for Scaling-up Personalized Recom- mender Systems, Proc. of the ACM WebKDD Workshop (WebKDD'00), Boston, MA, USA, pp. 1–9.

[13] B. M. Sarwar, G. Karypis, J. A. Konstan, and J. Riedl, (2001). Item-based Collaborative Filtering Recom mender Algorithms, Proc. of the 10th International Conference on World Wide Web (WWW'01), ACM Press, Hong Kong, China, pp. 285–295.

[14] J. Oakland, (2003). Statistical Process Control, the 5th Revised Edition, Butterworth-Heinemann Ltd.

[15] H. Sun, Z. Zheng, J. Chen, and M. R. Lyu, (2011). NRCF: A Novel Collaborative Filtering Method for Service Recommendation, Proc. of IEEE International Conference on Web Services (ICWS'11), IEEE CS Press, Washington, DC, USA, pp. 702–703.

[16] Y. Shi, K. Zhang, B. Liu, and L. Cui, (2011). A New QoS Prediction Approach Based on User Clustering and Regression Algorithms, Proc. of IEEE International Conference on Web Services (ICWS'11), IEEE CS Press, Washington, DC, USA, pp. 726–727.

[17] J. Li, L. Sun, and J. Wang, (2012). A Slope One Collaborative Filtering Recommendation Algorithm Using Uncertain Neighbors Optimizing, Proc. of WAIM 2011 International Workshops, Springer- Verlag Berlin, Wuhan, China, pp. 160–166.

[18] P. Wang and H. W.Ye, (2009). A Personalized Recommendation Algorithm Combining Slope One Scheme and User Based Collaborative Filtering, Proc. of International Conference on Industrial and Information Systems (IIS'09), IEEE CS Press, Haikou, China, pp. 152–154.

[19] D. J. Zhang, (2009). An Item-based Collaborative Filtering Recommendation Algorithm Using Slope One Scheme Smoothing, Proc. of the Second International Symposium on Electronic Commerce and Security (ISECS'09), Vol. 2, IEEE CS Press, Nanchang, China, pp. 215–217.

[20] X. Li, F. Zhou, and X. Yang, (2011). Research on Trust Prediction Model for Selecting Web Services based on Multiple Decision Factors, International Journal of Software Engineering and Knowledge En- gineering, Vol. 21, No. 8, World Scientific Publishing Company, pp. 1075–1096.

Authors Profile

G. Vadivelou pursued Bachelor of Science in Mathematics from Pondicherry University, India in 1991, Master of Computer Applications from Bharathiar University, Tamil Nadu, India in 1994 and Ph.D. in Computer Science in 2016 from Bharathiar University, Tamil Nadu, India. He is working as Assistant Professor in the Department of Computer Science, Kanchi Mamunivar Centre for PG Studies (Autonomous) affiliated to Pondicherry University, Puducherry, India since 2009. He has published more than 15 research papers in reputed international journals and conferences including IEEE. His main research work focuses on Data Mining, Distributed Computing, Web Services. He has more than 10 years of teaching experience.

E. Ilavarsan pursued Bachelor of Science, Master of Computer Applications, Master of Technology in Computer Science & Engineering and Ph.D. in Computer Science from Pondicherry University, India. He is working as Professor in Department of Computer Science, & Engineering, Pondicherry Engineering College, Puducherry, India. He has published more than 20 research papers in reputed international journals and conferences including IEEE. His main research work focuses on Data Mining, Microprocessor & Microcontroller and Distributed Computing. He has more than 20 years of teaching experience.

