Personalized Context-Aware QoS Prediction for Web Services Based on Collaborative Filtering

Qi Xie, Kaigui Wu, Jie Xu, Pan He, and Min Chen

College of Computer Science,
Chongqing University,
Chongqing 400044, China
{xieqi,kaiguiwu,hepan,chenmin}@cqu.edu.cn,
scsjx@leeds.ac.uk

Abstract. The emergence of abundant Web Services has enforced rapid evolvement of the Service Oriented Architecture (SOA). To help user selecting and recommending the services appropriate to their needs, both functional and nonfunctional quality of service (QoS) attributes should be taken into account. Before selecting, user should predict the quality of Web Services. A Collaborative Filtering (CF)-based recommendation system is introduced to attack this problem. However, existing CF approaches generally do not consider context, which is an important factor in both recommender system and QoS prediction. Motivated by this, the paper proposes a personalized context-aware QoS prediction method for Web Services recommendations based on the SLOPE ONE approach. Experimental results demonstrate that the suggested approach provides better QoS prediction.

Keywords: Context, QoS Prediction, Collaborative Filtering, Web Service.

1 Introduction

The Service Oriented Architecture (SOA) is “an architecture that represents software functionality as discoverable services on the network” [1]. Web Services are the dominant implementation platform for SOA, it uses a set of standards, SOAP, UDDI, WSDL, which enable a flexible way for applications to interact with each other over networks [2]. The increasing presence and adoption of Web Services call for effective approaches for Web Service selection and recommendation, which is a key issue in the field of service computing [3]. Not only functional but also nonfunctional quality of services (QoS) should be taken into account to help users selecting and recommending the services. The QoS includes service reliability, availability, performance metrics (e.g., response time), scalability, (transactional) integrity, security, trust and execution cost, etc.

Web Service QoS prediction is an important step in selecting services [4]. Since changes of QoS happen due to various reasons, such as number of concurrent users, performance of back end systems, network latency, invocation failure-rate, etc. In addition, there is a problem that QoS changes are observed at run time. So it is critical

L. Cao, J. Zhong, and Y. Feng (Eds.): ADMA 2010, Part II, LNCS 6441, pp. 368–375, 2010.

to predict the QoS before using it. Currently, many researches focus on this prediction problem according to users' QoS date acquired by user. Though this method can obtain accurate results, there are some shortcomings. For example, it is impossible for user to use all candidate services because it requires extra cost on service invocation and provision. Furthermore, most of service users are not experts on the Web Service evaluation, and the common time-to-market constraints limit an in-depth evaluation of the target Web Services [5].

To solve these problems, collaborative filtering (CF) method is introduced to predict QoS for unused services [5]. This method is based on the assumption that the service users, who have similar historical QoS experience on the same set of Web Services, would have similar experience on other services. But these differences on the quality of the same Web Services are caused by many context factors, such as location and time. Users/Web Services in near location may have similar experience on Round-Trip Time (RTT), failure-rate than others. However, existing approaches generally do not consider context in real QoS date prediction with CF. To improve the accuracy of prediction of the QoS, in this paper, we present our work on personalized context-aware QoS prediction based on SLOPE ONE.

The major contributions of this work include:

* we propose a personalized context-aware QoS prediction framework.
* we employ an effective and simple collaborative filtering algorithm (SLOPE ONE) to improve the QoS prediction accuracy.
* A personalized context-aware QoS prediction based on SLOPE ONE is designed for Web Service prediction, which significantly improves the QoS prediction accuracy and remarkably reduces the computing complexity.

The remainder of this paper is organized as follows. Section 2 overviews some concepts e.g., the CF algorithms and context of Web Services. Section 3 presents the approach for personalized context-awareness QoS prediction. The implementation, experiments, and the results of our method are discussed in Section 4. Conclusion of this paper is described in Section 5.

2 Background

CF is a widely used personalized recommendation method in recommender systems. Basically, there are two kinds of CF algorithms, user based and item based approaches [6, 7]. User-based CF is the most successful recommending technique to date, and is extensively used in many commercial recommender systems [6]. It predicts a test user's interest in a test item based on rating information from similar user profiles [7]. Multiple mechanisms such as Pearson Correlation and Cosine based similarity are widely used. Item-based CF methods are similar to the user-based methods. Item-based CF employs the similarity between items and then to select the most similar items for prediction.

The SLOPE ONE is a typical item-based CF. It works on comparing the intuitive principle of a popular differential between items rather than similarity between items [8]. Daniel Lemire defined the SLOPE ONE the average deviation (1) and prediction (2) as the following:

$$dev_{i,j} = \sum_{u \in U(i) \cap U(j)} \frac{r_{u,i} - r_{u,j}}{|U(i) \cap U(j)|}, \tag{1}$$

$$p_{u,i} = \overline{r_u} + \frac{\sum_{j \in R_u} dev_{i,j}}{|R_u|}. \tag{2}$$

The $dev_{i,j}$ is computed by the average difference between item arrays of i and j (1). $\overline{r_u}$ is the vector of average of all known ratings rated by user u. R_u are the all items' ratings rated by u except i. $|R_u|$ is the number of R_u. $|U(i) \cap U(j)|$ is the number of users who rate both item i and item j.

Context is any information that can be used to characterize the situation of an entity [9]. Context -aware computing refers to the ability of a software application to detect and respond to changes in its environment [10]. The idea of the context-aware CF is based that similar users in similar context like similar items [11]. Context is of three types in our method: user (U-context), Web Service (WS-context), and service provider (SP-context).

3 Personalized Context-Aware QoS Prediction

3.1 Deployment

Fig. 1 illustrates the framework that is proposed to predict the QoS for Web Services based on Context-aware CF. The Personalized Context-aware QoS Prediction based on CF (PCQP) can automatically predict the QoS performance of a Web Service for a test user by using historical QoS information from other similar service users on the same set of commonly-invoked Web Services.

The framework consists of the following procedures: 1) Query. A service user submits a Web Service request. 2) Select. The candidates that meet the functional service quality attributes proposed by the user are retrieved by employing Web Service discovery mechanism. 3) Context. Personal U-context and WS-context are considered. 4) Find similar users and services from the training data. 5) Predict. The values of QoS are generated for test users using PCQP. 6) Recommender. It provides test user the optimal Web Services according to the predicted QoS values.

The core parts in this framework are how to find the most similar users and Web Services, and how to predict the missing QoS values more efficiently and effectively. Due to space restrictions, this paper only researches the similar Web Service item and context of service, the others will be studied in future work.

3.2 Identification of Context

Context-awareness allows software applications to use information beyond those directly provided as input by users [12]. In Web Services scenario, the information includes such as location, date, time related to users and services.

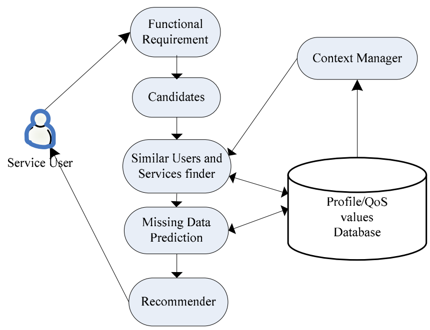

Fig. 1. Personalized context-aware QoS prediction framework overview

We adopt recommendation space (RS) [13] to present PCQP. In RS, the key dimensions could be used as QoS values. For example, if the RS includes user, service item and service location, the QoS value q(u,i,l) lies in a three-dimensional space U×I×L. Every entry in RS presents the QoS values (e.g., RTT), which is observed by the service user u on the Web Service item i that belongs to l. The q(u,i,l)=0 if user u did not use the Web Service item i in location l before.

Existing approaches for QoS prediction based on CF tend to compute the similarity in the whole user-service matrix. Our work is very different from these methods since we employ the information of context related to users and services. According to the reduction RS of similarity computation, our method minimizes the computational cost while satisfying the QoS value prediction accuracy of Web Services.

3.3 Deviation Computation

In this work, the SLOPE ONE based on context for the deviation computation of different service items is defined as:

$$dev_{i,j^*} = \sum_{u \in U(i) \cap U(j^*)} \frac{r_{u,i} - r_{u,j^*}}{|U(i) \cap U(j^*)|} \tag{3}$$

The difference between (1) and (3) is the RS of deviation computation. The RS of SLOPE ONE based context is the three-dimensional space U×I×L while the one without context is two dimensional space U×I. For example, to predict the $P_{u,i}$, we should compute the deviation between i and other service item j. In our context method the space of j becomes the item j* that belongs to the location in which item i is provided.

3.4 Missing Value Prediction

The SLOPE ONE methods use differential service items to predict the missing value (QoS) for test users by using the following equation:

$$p_{u,i} = \overline{r_{u*}} + \frac{\sum_{j* \in R_{u*}} dev_{i,j*}}{|R_{u*}|} \quad (4)$$

This is the same as (2) except for the changes of variables space(e.g., $dev_{i,j}$, r_u, R_u). Owing to the applying of context in our method, the RS is limited to three-dimensional space U×I×L.

4 Experimental Evaluation

4.1 Data Set and Evaluation Metric

In the experiment we use the dataset from the WSRec including a user-contribution mechanism for Web Service QoS information collection, which is implemented by java language and deployed to the real-world environment [5, 14]. This dataset includes 100 Web Services and 150 computer nodes (users) which are distributed in 22 countries. We obtain a 150×100 user-service item matrix from this database, where each entry in the matrix represents the QoS value (RTT).

The error metric used most often in the CF literature is the Mean Absolute Error (MAE) , which is described as:

$$MAE = \frac{\sum_{i,j} |r_{i,j} - \hat{r}_{i,j}|}{N}, \quad (5)$$

where $r_{i,j}$ is the involved QoS value of service item j experienced by user i, $\hat{r}_{i,j}$ presents the predicted QoS value for missing value. In addition, N is the number of predicted values.

Because the different QoS properties of Web Services have different value ranges [5], we employed the Normalized Mean Absolute Error (NMAE) to measure the deviation between predictions and QoS values (RTT). To compare the NMAE employed in other methods more effectively, we adopt the NMAE proposed by ZiBin Zheng [5] as follows:

$$NMAE = \frac{MAE}{\sum_{i,j} r_{i,j} / N}. \quad (6)$$

The lower the NMAE is, the higher the prediction quality is.

4.2 Performance Comparison

To research the predicted results of QoS values, we first compare the SLOPE ONE with other prediction approaches used widely in CF, such as: UPCC, IPCC. Then SLOPE ONE and personalized context-aware prediction method based on the SLOPE ONE (PCSP) will be compared in section 4.3. The total item of QoS values are divided into two parts randomly, one as the training set and the other as the test set. A parameter λ is employed to present the percentage of the matrix entries which are selected to be the training set.

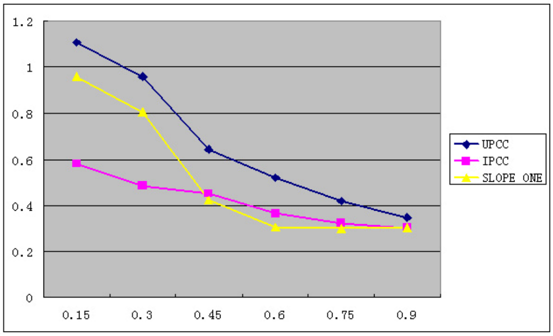

Fig. 2. Normalized mean absolute error rank for three prediction methods

In the experiment, we set λ from 0.15 to 0.9, with 0.15 as increment. The x-axis represents different value of λ, and the y-axis shows the NMAE values. As shown in Fig. 2, the prediction accuracy of UPCC is worse than other methods. The NMAE of SLOPE ONE is higher than IPCC at first two steps and consistently lower than IPCC in the following three steps. The NMAE of SLOPE ONE reaches the optimum value (0.297) when the λ is 0.75. The reason is that when the number of training matrix is small, the number of differential service items of SLOPE ONE is very limited and with the λ increasing more differential service items will appear which will provide more service items with low deviation and make the accuracy improvement more significant. After the step of 0.75 both the SLOPE ONE and IPCC perform worse. This is because too many training data appear diversity influencing the prediction performance.

4.3 Impact of the Context of Service Items

Context plays an important role in our approach, which is quite different from other methods. To research the performance of the context, we employ a simple experimental scenario as Fig. 3.

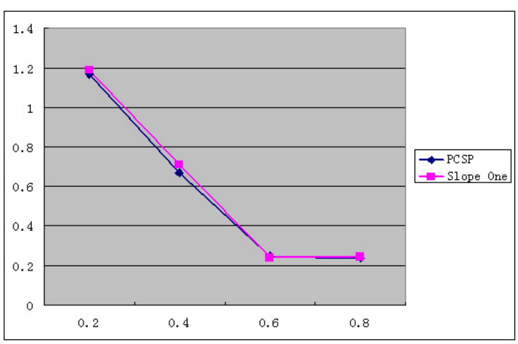

Fig. 3. Impact of the Context of Service Items

In Fig. 3, the Web Service items only belonging to China and Spain will be chose to constitute a new dataset with Web Service item size of 80-150. The new dataset is also divided into training set and test set. The value of λ is set from 0.2 to 0.8 with 0.2 as increment. From Fig. 3 we note that the prediction accuracy consistently become better with the increase of λ value. The performances of PCSP get very close to the SLOPE ONE. The PCSP obtains the minimal NMAE value (0.2356) when λ is set to 0.8.

Owing to the calculations of deviation and prediction in PCSP are implemented in the three-dimensional space U×I×L, the PCSP will efficiently decrease the computational complexity. These experiments data represent that the PCSP algorithm is helpful to improve the prediction results efficiently and effectively.

5 Conclusion

With regard to the context, an effective personalized Context-aware QoS Prediction framework for Web Services based on Collaborative Filtering is proposed. The experiments analysis presents the efficiency and effectiveness of our approach.

However, there is no similarity weight computation in our work which may enhance the accuracy with missing value. One reason is that our method only focuses on the impact of the context. Using too many factors to impact the predictions may compromise the experimental results. In future work, different QoS properties and multi-context will be integrated with collaborative filtering method to predict QoS values.

Acknowledgments. This work is supported by the Major Research Project of the National Natural Science Foundation of China under Grant No.90818028 and the Third Stage Building of "211 Project" (Grant No. S-10218).

References

1. Klusch, M., Kapahnke, P.: Semantic Web Service Selection with SAWSDL-MX. In: 2nd Workshop on Service Matchmaking and Resource Retrieval in the Semantic Web (SMR2), pp. 3–18 (2008)
2. Sathya, M., Swarnamugi, M., Dhavachelvan, P., Sureshkumar, G.: Evaluation of QoS based Web Service Selection Techniques for Service Composition. J. International Journal of Software Engineering
3. Zhang, L.J., Zhang, J., Cai, H.: Services computing. Springer and Tsinghua University Press, Beijing (2007)
4. Shao, L.S., Zhang, J., Wei, Y., Zhao, J., Xie, B., Mei, H.: Personalized QoS prediction for web services via collaborative filtering. In: ICWS, pp. 439–446 (2007)
5. Zheng, Z.B., Ma, H., Michael, R.: L., King, I.: WSRec: A Collaborative Filtering based Web Service Recommender System. In: 7th IEEE International Conference on Web Services (ICWS 2009), Los Angeles, CA, USA, pp. 6–10 (2009)
6. Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., Riedl, J.: GroupLens: An open architecture for collaborative filtering of Netnews. In: Conference on Computer Supported Cooperative Work, pp. 175–186. Chapel Hill, NC (1994)
7. Breese, J.S., Heckerman, D.: Kadie. C.: Empirical analysis of predictive algorithms for collaborative filtering. In: UAI, pp. 43–52 (1998)

8. Lemire, D., Maclachlan, A.: Slope One Predictors for Online Rating-Based Collaborative Filtering. Society for Industrial Mathematics (2005)
9. Dey, A.K.: Understanding and using context. J. Personal and Ubiquitous Computing 5, 4–7 (2001)
10. Roman, G.C., Julien, C., Murphy, A.L.: A Declarative Approach to Agent-Centered Context-Aware Computing in Ad Hoc Wireless Environments. In: 2nd Inter. Work. Software Engineering for Large-Scale Multi-Agent Systems (SELMAS 2002), Orlando, Florida, USA (2002)
11. Gao, M., Wu, Z.H.: Personalized Context-Aware Collaborative Filtering Based on Neural Network and Slope One. In: Luo, Y. (ed.) Cooperative Design, Visualization, and Engineering. LNCS, vol. 5738, pp. 109–116. Springer, Heidelberg (2009)
12. Dey, A.K., Abowd, G.D., Salber, D.: A Conceptual Framework and a Toolkit for Supporting the Rapid Prototyping of Context-aware Applications. J. Human Computer Interaction Journal 16, 97–166 (2001)
13. Weng, S.S., Lin, B.S., Chen, W.T.: Using Contextual Information and Multidimensional Approach for Recommendation. J. Expert Systems with Applications, 126–127 (2009)
14. Zhang, Y.L., Zheng, Z.B., Michael, R.: L.: WSExpress: A QoS-aware Search Engine for Web Services. In: 8th International Conference on Web Services, USA, pp. 5–10 (2010)