Expert Systems With Applications 53 (2016) 75–86

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Context-aware QoS prediction for web service recommendation and selection

Yueshen Xu[a], Jianwei Yin[a], Shuiguang Deng[a,*], Neal N. Xiong[b], Jianbin Huang[c]

[a] College of Computer Science and Technology, Zhejiang University, Hangzhou, Zhejiang 310027, China
[b] School of Computer Science, Colorado Technical University, Colorado Springs, CO 80907, USA
[c] School of Software, Xidian University, Xi'an, Shaanxi 710071, China

ARTICLE INFO

Keywords:
QoS prediction
Context information
Web service
Similarity computation
Matrix factorization

ABSTRACT

QoS prediction is one of the key problems in Web service recommendation and selection. The context information is a dominant factor affecting QoS, but is ignored by most of existing works. In this paper, we employ the context information, from both the user side and service side, to achieve superior QoS prediction accuracy. We propose two novel prediction models, which are capable of using the context information of users and services respectively. In the user side, we use the geographical information as the user context, and identify similar neighbors for each user based on the similarity of their context. We study the mapping relationship between the similarity value and the geographical distance. In the service side, we use the affiliation information as the service context, including the company affiliation and country affiliation. In the two models, the prediction value is learned by the QoS records of a user (or a service) and the neighbors. Also, we propose an ensemble model to combine the results of the two models. We conduct comprehensive experiments in two real-world datasets, and the experimental results demonstrate the effectiveness of our models.

1. Introduction

As the non-functional properties of Web service, quality of service (**QoS** for short) is one of the key criteria in service recommendation and selection (Huang, Lan, & Yang, 2009; Yu, Zhang, & Lin, 2007; Zheng, Ma, Lyu, & King, 2013), and even in service composition and discovery (Menascé, 2004; Parejo, Segura, Fernandez, & Ruiz-Cortés, 2014; Phalnikar & Khutade, 2012; Shehu, Epiphaniou, & Safdar, 2014). QoS includes a series of non-functional properties, such as response time, throughput, reliability and some others (Papazoglou, 2003). Web service recommendation and selection are two of the key problems that are concerned in service computing community (Chen, Zheng, Yu, & Lyu, 2014; Kayastha & Baria, 2015). In Web service recommendation and selection, the services with superior QoS values (e.g., short response time or large throughput) are recommended to users. In most existing QoS-aware works, there is an assumption that all QoS values have been known and stable. However, in the real-world scenario of service invocation, such an assumption is hard to meet due to the following two reasons.

1. With the popularity of cloud computing, Web service is being widely used to provide configurable resources on the Internet, such as data storage, distributed computing and data center management (Microsoft, 2015a; 2015b; VMware, 2015). The number of Web services increases so rapidly that it is costly and time-consuming to invoke all of them.
2. QoS values have high possibility to be changeable due to the change of running environment of Web services.

Usually, only a small number of QoS values are known, and the remaining large numbers of values are missing. An example is shown as the user-service invocation matrix in Fig. 1, in which the grey entries denote the missing QoS values. QoS prediction is an inevitable step for subsequent process, such as Web service recommendation and selection.

In general, there are two kinds of QoS prediction methods, that is, Collaborative Filtering (**CF** for short) and Matrix Factorization (**MF** for short). CF is a mature technique in recommender systems to predict ratings (Deng, Wang, Li, & Xu, 2015; Desrosiers & Karypis, 2011; Su & Khoshgoftaar, 2009), and has been widely studied in QoS prediction (Chen, Liu, Huang, & Sun, 2010; Xie, Wu, Xu, He, & Chen, 2010). CF consists of two steps, i.e., similarity

* Corresponding author. Tel.: +86 13605712329.
E-mail addresses: xyshzjucs@zju.edu.cn (Y. Xu), zjuyjw@zju.edu.cn (J. Yin), dengsg@zju.edu.cn, deng.terrry@gmail.com (S. Deng), xiongnaixue@gmail.com (N. N. Xiong), jbhuang@xidian.edu.cn (J. Huang).

http://dx.doi.org/10.1016/j.eswa.2016.01.010
0957-4174/

	Service1	Service2	Service3	Service4
User1	q_{11}			
User2		q_{22}		q_{24}
User3				
User4	q_{41}			q_{44}

Fig. 1. The user-service invocation matrix.

computation and missing value computation. Using some similarity measures, such as Pearson correlation coefficient (**PCC** for short) and cosine similarity, CF first computes the similarity between two users (or services) based on QoS records. After that, each prediction value is computed as the weighted average of the known values of the similar neighbors. Although CF-based methods are easy to implement, the following defects limit its application.

1. CF fails to compute the similarity in the case of "cold-start" scenario, which means that a user or a service only has few or even no QoS records. The "cold-start" problem widely exists in QoS prediction, and seriously harms the prediction accuracy (Yao, Sheng, Ngu, Yu, & Segev, 2015).
2. It is hard for CF-based methods to integrate the factors that impact the QoS, such as the context information. It is necessary to exploit more extensible models.

Matrix Factorization (**MF** for short) is also a successful base model in recommender systems (Koren, Bell, & Volinsky, 2009), and has been employed for QoS prediction in recent years (He, Zhu, Zheng, Xu, & Lyu, 2014; Lo, Yin, Deng, Li, & Wu, 2012; Zheng et al., 2013). MF-based methods aim to learn the latent factors that are behind the QoS values and decide the QoS values. In QoS prediction, most of existing MF-based methods follow such a methodology. First, they identify the similar user (or service) neighbors via similarity computation based on QoS records. Then, they learn the missing values with the assistance of the QoS records of similar neighbors. The existing works show that the MF-based methods can achieve higher prediction accuracy than CF-based methods. However, there are three drawbacks due to the utilization of QoS values in similarity computation.

1. The QoS value is changeable due to the change of the resource configuration of service invocation, such as the connection speed and connection stability. Such an uncertainty lowers the credibility of the similarity results.
2. The similarity computation has to be conducted specific to each single QoS property. The similarity results cannot be reused among different QoS properties. For example, if we have the QoS records of *response time* and *throughput* (two QoS properties), we have to conduct the similarity computation twice based on the response time records and throughput records, respectively.
3. The similarity is likely to be inaccurate in the case that the QoS records are sparse.

It is necessary to find the reliable impact factors and build more effective models. We explain our observation and intuition here. In the real-world service invocation, QoS is largely impacted by the running environment of the user side and service side. In the user side, the connection speed is one of the dominant factors influencing QoS. Let us see an example. According to the report of Akamai (2015), the average connection speed in South Korea and Japan is very close, and much faster than that in Singapore. It can be inferred that when users in Seoul, Tokyo and Singapore invoke the same service, users living in Seoul and Tokyo are likely to experience much shorter response time than that in Singapore. Also naturally, the users living near to each other are likely to receive similar QoS. In the service side, the running environment, such as network stability and computing power, also determines QoS. Services that are run by the same company tend to provide similar QoS, although the services may have different functions, since they run in the same computing platform and share similar computing resource. The IT infrastructure in the user side and the running environment in the service side, both can be referred to as the *context* of service invocation (Kuang, Xia, & Mao, 2012; Xie et al., 2010). You can see an example of the real-world service invocation in Fig. 2, which depicts the scenario stated in this paragraph.

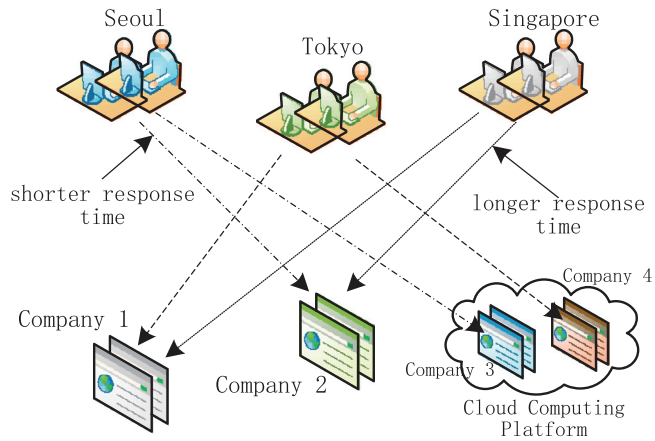

Fig. 2. The real-world web service invocation (a toy example).

In this paper, we focus on studying how to use the context information to improve the QoS prediction accuracy. We propose two context-aware QoS prediction models. The proposed models have the capability of integrating the context information of the user side and the service side, respectively. We use the MF model as the base model. In the user side, we study several similarity functions that provide different mapping relationships between the similarity and the geographical distance. Using the selected similarity function, we propose the user context-aware MF model (**UC-MF** for short). In the service side, we also study the effect of context on QoS, focusing on finding the similar neighbors that are run under the similar running environment. After that, we build the service context-aware MF model (**SC-MF** for short). Also, we propose an ensemble model combining the results of UC-MF and SC-MF. We conduct sufficient experiments, and the experimental results demonstrate the effectiveness of our proposed models.

In summary, this paper makes the following contributions:

1. It studies the mapping relationship between the similarity and the geographical distance. It selects the most effective similarity function from different candidates, which can achieve a more accurate similarity between two users based on geographical information.
2. It proposes two novel context-aware QoS prediction models, and also combines them to construct an ensemble model. All models are capable of integrating the context information. Note that, most existing works study the QoS prediction problem only from one single side, i.e., either the user side or the service side.
3. It demonstrates that compared to many other methods, our proposed models can save much computation, and are quite suitable for the "cold-start" scenario. It is because the similarity computation in our models does not rely on the QoS records. Meanwhile, the similarity results can be used in different prediction tasks for different QoS properties.