Clustering-based uncertain QoS prediction of web services via collaborative filtering

Guobing Zou, Wang Li, Zhimin Zhou and Sen Niu

School of Computer Engineering and Science,
Shanghai University,
Shanghai, China
Email: gbzou@shu.edu.cn
Email: liwang@shu.edu.cn
Email: zmzhou@shu.edu.cn
Email: sniu@shu.edu.cn

Yanglan Gan*

School of Computer Science and Technology,
Donghua University,
Shanghai, China
Email: ylgan@dhu.edu.cn
*Corresponding author

Bofeng Zhang

School of Computer Engineering and Science,
Shanghai University,
Shanghai, China
Email: bfzhang@shu.edu.cn

Abstract: Although collaborative filtering (CF) has been widely applied for QoS-aware web service recommendation, most of these approaches mainly focus on certain QoS prediction. However, they failed to take the natural characteristic of web services with QoS uncertainty into account in service-oriented web applications. To solve the problem, this paper proposes a novel approach for uncertain QoS prediction via collaborative filtering and service clustering. We first establish uncertain QoS model for a service user, where each service is formalised as a QoS matrix. To mine the similar neighbourhood users for an active user, we then extend the Euclidean distance to calculate the similarity between two uncertain QoS models. Finally, we present two kinds of QoS prediction strategies based on collaborative filtering and clustering, called U-Rec and UC-Rec. Extensive experiments have been carried on 1.5 million real-world uncertain QoS transaction logs of web services. The experimental results validate the effectiveness of our proposed approach.

Keywords: collaborative filtering; service clustering; uncertain QoS prediction; web service.

Reference to this paper should be made as follows: Zou, G., Li, W., Zhou, Z., Niu, S., Gan, Y. and Zhang, B. (2017) 'Clustering-based uncertain QoS prediction of web services via collaborative filtering', *Int. J. Web and Grid Services*, Vol. 13, No. 4, pp.403–424.

Biographical notes: Guobing Zou is an Associate Professor in the School of Computer Engineering and Science at Shanghai University, China. He received his PhD in Computer Science from Tongji University, Shanghai, China, 2012. He worked as a visiting scholar at Washington University in St. Louis from 2009 to 2011, USA. His current research interests focus on services computing, data mining, intelligent algorithms and personalised recommendation. He has published more than 45 papers on international journals and conferences, including *IEEE Transactions on Services Computing*, *IEEE ICWS*, *IEEE SCC*, *AAAI*, *IEEE/ACM TCBB*, *Journal of Web Semantics*, *Knowledge-Based Systems*, *Applied Intelligence* and *Soft Computing*.

Wang Li is currently a master student in the School of Computer Engineering and Science, Shanghai University, China. He received his undergraduate degree from Shanghai University. His research interests include web service recommendation and data mining. He has established a public web service dataset scratched from ProgrammableWeb.

Zhimin Zhou received his Master's Degree from the School of Computer Engineering and Science, Shanghai University, China. His research interests include uncertainty QoS prediction of web services and data mining. He has published a paper on *IEEE International Conference on Services Computing*.

Sen Niu is currently a PhD candidate in the School of Computer Engineering and Science, Shanghai University, China. His research interests include web service composition with uncertainty and intelligence planning. He has published several papers on *IEEE ICWS*, *IEEE SCC* and *Journal of Internet Technology*.

Yanglan Gan is an Associate Professor in the School of Computer Science and Technology at Donghua University, Shanghai, China. She received her PhD in Computer Science from Tongji University in 2012. Her research interests include bioinformatics, services computing, and data mining. She published more than 20 papers on international journals and conferences, including *Bioinformatics*, *IEEE/ACM TCBB*, *Neurocomputing* and *Knowledge-Based Systems*.

Bofeng Zhang is a Full Professor in the School of Computer Engineering and Science at Shanghai University, China. His research interests include social computing, service computing, and artificial intelligence. He has published more than 120 papers on international journals and conferences. He served as program chair or PC for varieties of international conferences.

1 Introduction

Web services are self-contained and self-describing software components designed to support machine-to-machine interaction by programmatic calls and deployed for use by personal or organisational invocations on the internet. Currently, there are many widely used standard description languages and protocols. WSDL and UDDI have been employed to describe and register the functionality of service interface, while SOAP has been applied as a communication protocol for exchanging structured information during the execution of web services. Benefiting from the characteristics of cross-language and cross-platform, web services have received many attentions from both industry and academia.

Quality of Service (QoS) has become an important criterion and been employed to describe non-functional characteristics of web services, which is applied for differentiating those services with the same functionality (Zheng and Lyu, 2013). QoS criteria can be divided into two kinds of categories, including user-independent and user-dependent properties. QoS values of user-independent criteria (e.g. price, popularity) are usually advertised by service providers, which are identical to different service requesters. Due to the influence of unpredictable network connections and heterogeneous user environments, however, QoS values of those user-dependent criteria (e.g. failure probability, response time, throughput) can vary widely from different service requesters.

Much research has been made on QoS-aware approaches of web services, including service selection under QoS constraints (Hadad et al., 2008, 2010), dynamic composition of web services with QoS optimality (Yilmaz and Karagoz, 2014; Feng et al., 2013), and QoS prediction for service recommendation. QoS accumulation has been considered as the key factor among all of above works. However, it is always difficult for service requesters to acquire those web services with their desirable QoS information. The reason is that it is a tedious and time-consuming task to conduct real-world invocations on all candidate web services. Furthermore, it is also impractical to release such QoS information from service providers or third-communities, because a variety of non-functional values of web services hold the natural characteristics with QoS uncertainty in different user application scenarios (e.g. geographical location, network deployment). As a result, QoS prediction of web services has become an effective way of making strategic decisions when recommending web services to active users.

In recent years, a lot of research work has been done to apply collaborative filtering (CF) for QoS prediction of web services. The idea is to utilise observed QoS historical logs of different web services and match similar users together to predict QoS of web services that would be potentially invoked by an active user. These approaches are divided into two categories: traditionally collaborative filtering based approaches and context-aware approaches. The traditionally collaborative filtering based approaches for QoS prediction consist of memory-based approaches (Shao et al., 2007; Zheng et al., 2009, 2011; Sun et al., 2011, 2013; Jiang et al., 2011) and model-based approaches (Zheng et al., 2013; Li et al., 2015; Abdullah and Li, 2015). Memory-based approaches either predict missing QoS values for active users based on their similar users or similar services, while model-based approaches make missing QoS prediction by developing a model of user historical invocation records. The model building process is usually performed by different machine learning algorithms, such as Bayesian network, clustering and rule-based reasoning. For context-aware approaches, they predict QoS with the consideration of time series and spatial features, i.e., time-aware collaborative filtering (Tian et al., 2014; Yu and Huang, 2014; Hu et al., 2015) and location-aware collaborative filtering (Chen et al., 2010; Wei et al., 2012; Tang et al., 2012). However, these approaches failed to take the uncertainty of QoS transactions into consideration for effective service recommendation. The fact is that an active user may invoke the same web service for many times rather than just once, which leads to multiple QoS transaction records during the process of practical service executions. Therefore, designing a novel approach to effectively solving QoS prediction of web services for a user has become a research issue. Consequently, the goal of our work is to recommend desired web services regarding uncertain QoS to an active user in real-world service-oriented applications.

To address above research issue, we propose a novel collaborative filtering based approach to enhance the accuracy of uncertain QoS prediction of web services via service clustering. The main contributions are threefold as follows.

- First, we present an uncertain QoS user model, where uncertain QoS of a service user is modeled as a three-layer tree. Furthermore, we mine similar neighbourhood set for an active user by the computation among uncertain QoS user models.
- Second, we propose an innovative collaborative filtering-based approach to uncertain QoS prediction via service clustering, including two kinds of QoS prediction strategies, called U-Rec and UC-Rec. It can achieve more accurate QoS prediction of web services.
- Third, we implement a prototype system for QoS prediction with uncertainty and conduct extensive experiments on a large-scale real-world dataset. It has more than 1.5 million web service QoS transaction logs from the invocations of users among 27 countries. The experimental results validate the effectiveness of the proposed approach for uncertain QoS prediction of web services.

The remainder of this paper is organised as follows. Section 2 shows a motivating example. Section 3 formulates the research problem of uncertain QoS prediction of web services. Section 4 presents our approach to predicting missing QoS. Section 5 reports extensive experimental results and analysis. Section 6 reviews related work. Finally, Section 7 concludes the paper and discusses the future work.

2 Motivating example

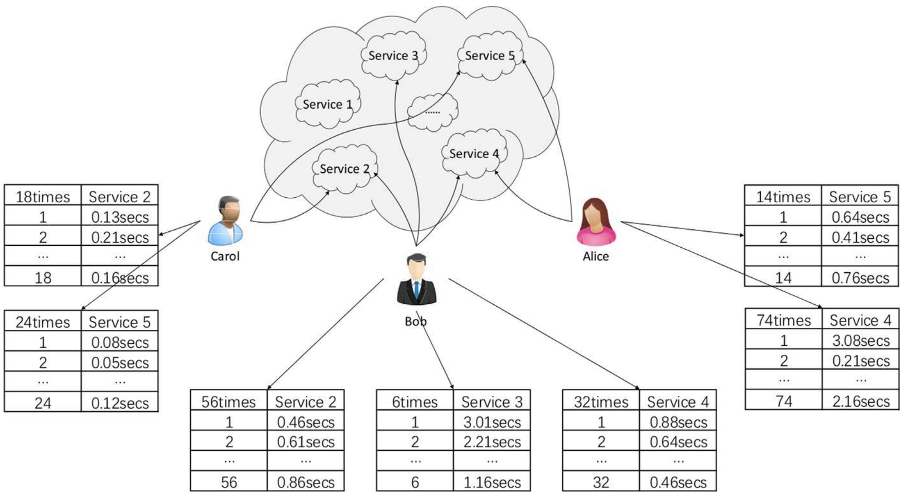
In this section, we show a motivating example to demonstrate the characteristic on QoS uncertainty of web services in real-world applications that we aim to address in this paper. As illustrated in Figure 1, there is a service-oriented application environment which contains a number of web services and users. Each user has invoked a subset of web services for different times, which leads to multiple uncertain service QoS execution logs. The non-functional QoS values (e.g. response time) are obtained by each service requester.

There are three service requesters including Alice, Bob and Carol, while five services are registered for use. Alice has invoked service 4 and service 5 for 14 and 74 times, respectively. These invocations incur different variations on response time due to uncertain internet environment. For example, it costs 0.64 seconds in the first time, while the value changes to 0.41 seconds in the second time. As for another service requester, Bob has invoked service 2, service 3 and service 4 with different uncertain service transaction logs. Finally, Carol has invoked service 2 and service 5 for 18 and 24 times, respectively.

Under above application scenario, suppose that let's take Bob as an active user and we aim to predict his missing QoS value on response time for service 5. In traditional collaborative filtering approaches, they consider the situation where each user invokes a service only once. However, in real-world service-oriented applications, a user may invoke a web service for many times, where it produces multiple QoS invocation logs. Based on above observation in real-world application demands, we should take QoS uncertainty of web services in real applications into consideration so as to improve prediction accuracy.

To address this challenge, we first formulate the research problem and then propose a clustering and collaborative filtering based approach to predict uncertain QoS for desired service recommendation.

Figure 1 The motivating scenario of web service invocations with QoS uncertainty (see online version for colours)



3 Problem formulation

To understand and formulate the problem of uncertain QoS prediction of web services, a set of definitions are given as follows.

Definition 1 (Uncertain web service)**:** An uncertain service s is 3-tuple $s = \langle I, O, Q\rangle$, where $\langle I, O\rangle$ are the inputs and outputs for service functionality. Q represents the non-functional performance with QoS uncertainty and $s.Q$ is denoted as QoS of s.

Here, we focus on the non-functional characteristics of web services. That is, we mainly consider uncertain QoS of those services a user has invoked. Taking Alice in the motivating scenario as an example, those uncertain QoS transaction logs of service 4 or service 5 invoked by Alice are modeled in $s_4.Q$ and $s_5.Q$, respectively.

Definition 2 (Uncertain Service Repository)**:** An uncertain web service repository S consists of a finite set of services, denoted as $S = \{s_1, s_2, ..., s_m\}$, where $s_i(1 \leq i \leq m)$ is an uncertain web service, and m is the number of services involved in S.

To simplify the description, we assume that all of the uncertain web services in a repository have the same functionality, while they have different uncertain QoS values during the invocation and execution.

Definition 3 (Service Users)**:** $U = \{u_1, u_2, ..., u_n\}$ is a set of users, where $u_i(1 \leq i \leq n)$ denotes a user in our uncertain QoS prediction problem, and n is the number of users.

Normally, we partition U into two independent user sets U_1 and U_2, where U_1 is provided as candidate similar neighbourhood users for each active user u_a in U_2, when predicting uncertain QoS values.

Definition 4 (QoS Criteria)**:** Given an uncertain web service $s = \langle I, O, Q\rangle \in S$, its QoS is aligned by a set of QoS criteria, $Q = \{q_1, q_2, ..., q_k\}$, where $q_i(1 \leq i \leq k)$ is used to represent one facet of non-functional QoS values of a web service.

QoS criteria can be modeled by a multi-dimensional vector, such as execution price, response time, reliability, etc. For simplicity, we here consider response time called round-trip time (RTT) as an example to describe QoS uncertainty of web services.

Definition 5 (Uncertain QoS of web Service)**:** Given an uncertain web service $s \in S$ associated with k QoS criteria $Q = \{q_1, q_2, ..., q_k\}$, a set of service users U_s, uncertain QoS of s is composed of transaction logs invoked by all of the service users U_s, which is formalised as a QoS matrix M_{r*k}:

$$M_{r*k} = \begin{pmatrix} q_{1,1} & q_{1,2} & \cdots\cdots & q_{1,k} \\ \vdots & \ddots & & \vdots \\ \vdots & & \ddots & \vdots \\ q_{r,1} & q_{r,2} & \cdots\cdots & q_{r,k} \end{pmatrix}$$

where r is the total number of invocation times on s by all service users in U_s and k represents the number of QoS criteria. From the perspective of service users, M_{r*k} can be partitioned into l number of submatrices, i.e. $M_{r*k} = M_{r_1*k} \cup M_{r_2*k} \cup ... \cup M_{r_l*k}$ where each $M_{r_i*k}(1 \leq i \leq l)$ is applied to record the transaction logs invoked by a service user $u_i \in U_s(1 \leq i \leq l)$, where $|U_s| = l$.

In the motivating example illustrated in Figure 1, Service 5 is invoked by both Carol and Alice for 24 and 14 times. Thus, the uncertain QoS of Service 5 can be represented as $M_{38*k} = M_{24*k} \cup M_{14*k}$, where

$$M_{24*k} = \begin{pmatrix} q_{1,1} & 0.08 \cdots & q_{1,k} \\ \vdots & \vdots \;\; \ddots & \vdots \\ q_{24,1} & 0.12 \cdots & q_{24,k} \end{pmatrix}$$

represents Carol's transaction logs on Service 5, while

$$M_{14*k} = \begin{pmatrix} q_{25,1} & 0.64 \cdots & q_{25,k} \\ \vdots & \vdots \;\; \ddots & \vdots \\ q_{38,1} & 0.76 \cdots & q_{38,k} \end{pmatrix}$$

represents Alice's transaction logs on Service 5. Combining M_{24*k} and M_{14*k}, we have M_{38*k} as below. It represents the uncertain QoS of web Service 5.

$$M_{38*k} = \begin{pmatrix} q_{1,1} & 0.08 & \cdots & q_{1,k} \\ \vdots & \vdots & \ddots & \vdots \\ q_{24,1} & 0.12 & \cdots & q_{24,k} \\ q_{25,1} & 0.64 & \cdots & q_{25,k} \\ \vdots & \vdots & \ddots & \vdots \\ q_{38,1} & 0.76 & \cdots & q_{38,k} \end{pmatrix}$$

Definition 6 (Uncertain QoS Prediction Problem)**:** An uncertain web service QoS prediction problem, denoted as SQP, is defined as a 3-tuple $\langle U, S, P \rangle$, where

- $U = \{u_1, u_2, ..., u_n\}$ is a set of users, where $u_a \in U$ is a service user who needs QoS prediction.
- $S = \{s_1, s_2, ..., s_m\}$ is an uncertain web service repository, where $s_i \in S$ is a target service which a service user u_a has never invoked.
- $P(u_a, s_i)$ is the prediction value of missing QoS if the active user u_a invokes web service s_i.

Following the motivating example on QoS uncertainty of web services, we define an SQP problem as $\langle U, S, P \rangle$, where $U = \{Alice, Bob, Carol\}$ and $u_a = Bob$. $S = \{Service\ 1, Service\ 2, Service\ 3, Service\ 4, Service\ 5\}$ and $s_i = Service\ 5$. $P(Bob, Service\ 5)$ is the predicted QoS when Bob invokes $Service\ 5$.

4 Uncertain QoS prediction of web services

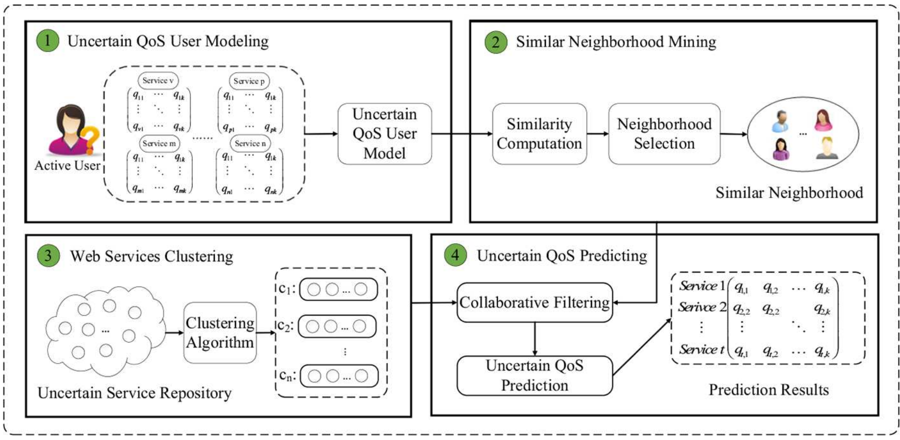
In this section, we first illustrate the overall framework of our approach for uncertain QoS prediction of web services. Then, we present each step during the process of QoS prediction.

4.1 QoS prediction framework

We design an approach for uncertain QoS prediction of web services via collaborative filtering and clustering. The overall framework is shown in Figure 2.

It is composed of four key steps, including: (1) uncertain QoS user modelling, (2) similar neighbourhood mining, (3) web services clustering and (4) uncertain QoS predicting. Initially, we build up an uncertain QoS model for each user according to the historical QoS transaction logs. Then, on the basis of non-functional modelling of a service user, we calculate the similarity between two uncertain QoS models and evaluate the neighbourhood relationship between two service users. As a result, a neighbourhood set can be chosen for an active user by the computation of uncertain QoS models. Subsequently, we partition all of the web services into a set of clusters via their uncertain QoS performance generated from those users who invoked the web services. Finally, combining the neighbourhood set and service clusters, we take a synthetic strategy into consideration to predict the potential QoS when an active user invokes a web service.

Figure 2 The framework of clustering-based uncertain QoS prediction of web services via collaborative filtering (see online version for colours)



4.2 Uncertain QoS user modelling

It is observed that a service user may invoke a set of web services, each of which leads to multiple QoS transaction logs. Thus, these transaction logs can be formalised as a QoS matrix with uncertainty. Based on this formalisation, the uncertain QoS user model is defined as follows.

Definition 7 (Uncertain QoS User Model)**:** An uncertain QoS user model is defined as a 4-tuple $\langle Auser, Lservices, Smatrices, f\rangle$, where $Auser$ represents an active user, $Lservices$ is a list of all the services which the Auser has invoked, $Smatrices$ consists of a number of different matrices and each matrix corresponds to uncertain QoS transaction logs of a service in $Lservices$. f is a mapping function from a service to its corresponding uncertain QoS matrix denoted by $f : Lservices \rightarrow Smatrices$.

In the motivating example, the uncertain QoS user model for Bob is $\langle Auser, Lservices, Smatrices, f\rangle$, where $Auser = Bob$, $Lservices = \{Service\ 2, Service\ 3, Service\ 4\}$, $Smatrices = \{M_{Service\ 2}, M_{Service\ 3}, M_{Service\ 4}\}$, and

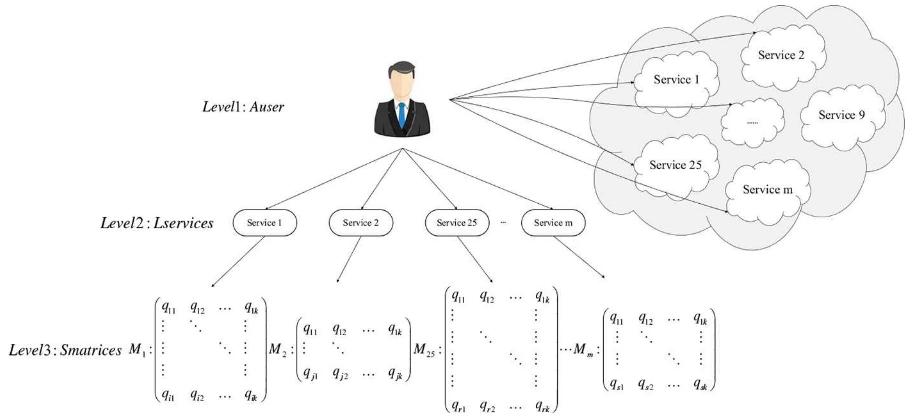
$$f(Service\ 2 \rightarrow M_{Service\ 2}) = \begin{pmatrix} q_{1,1} & 0.46 \cdots & q_{1,k} \\ \vdots & \vdots \ \ddots & \vdots \\ q_{56,1} & 0.86 \cdots & q_{56,k} \end{pmatrix},$$

$$f(Service\ 3 \rightarrow M_{Service\ 3}) = \begin{pmatrix} q_{1,1} & 3.01 \cdots & q_{1,k} \\ \vdots & \vdots \ \ddots & \vdots \\ q_{6,1} & 1.16 \cdots & q_{6,k} \end{pmatrix},$$

$$f(Service\ 4 \rightarrow M_{service\ 4}) = \begin{pmatrix} q_{1,1} & 0.88 \cdots & q_{1,k} \\ \vdots & \vdots \ \ddots & \vdots \\ q_{32,1} & 0.46 \cdots & q_{32,k} \end{pmatrix}.$$

From the above definition, an uncertain QoS user model of web services for an active user can be established as a tree with three layers, It is illustrated in Figure 3, including service user layer, invocation services layer, and service QoS matrices layer.

Figure 3 Uncertain QoS user model of web services for an active user (see online version for colours)



As the network environment is dynamically changing along with the invocation of web services, QoS values may significantly vary in the process of uncertain execution of web services. It may incur noisy QoS transaction logs. For example, a user often invokes a service on response time in a certain range from 0.1 to 0.3 s. However, occasionally it costs 0.8 s for the response of an invocation due to unstable internet environment. To improve the accuracy of uncertain QoS prediction, denoising strategy of uncertain QoS matrix can be taken by '3σ-rule', which is defined as follows.

Definition 8 (3σ-rule)**:** In statistics (Czitrom and Spagon, 1997), the 3σ-rule is a shorted used to remember the percentage of values that lie within a band around the mean in a normal distribution with a width one, two and three standard deviations, respectively. In mathematical notation, these facts can be expressed as follows, where x is an observation from a normally distributed random variable, μ is the mean of the distribution, and σ is its standard deviation: $P(\mu - \sigma \leq x \leq \mu + \sigma) \approx 0.6827$, $P(\mu - 2\sigma \leq x \leq \mu + 2\sigma) \approx 0.9545$, $P(\mu - 3\sigma \leq x \leq \mu + 3\sigma) \approx 0.9973$ and P represents the probability of x within a certain range.

Based on above denoising principle, we take into account a set of non-zero response time $RTT = \{R_1(s), R_2(s), ..., R_l(s)\}$ as an example, which is collected from l number of invocations of service s by a user. To estimate the mean μ and the standard deviation σ of the population, median and median absolute deviation (MAD) are calculated as follows:

$$med(RTT) = median(R_i(s)), i = 1, ..., l. \tag{1}$$

$$MAD(RTT) = median(|R_i(s) - med(RTT)|), i = 1, ..., l. \tag{2}$$

Based on median and MAD, the two estimators μ and σ can be calculated by:

$$\mu = median(R_i(s)), i = 1, ..., l. \tag{3}$$

$$\sigma = MAD(RTT), i = 1, ..., l. \tag{4}$$

By the computation of μ and σ, a noisy QoS $R_i(s)(i = 1, ..., l)$ can be identified and filtered out from the corresponding QoS matrix, if $(\mu - 3\sigma > R_i(s)) \wedge (\mu + 3\sigma < R_i(s))$ is satisfied. After the deletion of those noisy QoS values, the original QoS matrix can be refined for more accurate QoS prediction.

In the motivating example, Bob has invoked Service 3 for 6 times and the set of non-zero response time is $RTT = \{3.01, 2.21, 2.24, 2.25, 2.24, 1.16\}$. Under 3σ-rule denoising principle, we calculate $\mu = 2.24$, $\sigma = 0.02$, $\mu - 3\sigma = 2.18$, and $\mu + 3\sigma = 2.3$. By the application of denoising QoS values, we delete noisy values 1.16 and 3.01 for more accurate QoS prediction of web services.

4.3 Similar neighbourhood mining

Before QoS prediction on a web service invoked by an active user, the neighbourhood set which includes a set of similar users needs to be identified. Similar neighbourhood users mining is of vital importance for QoS prediction, because dissimilar users decrease the accuracy of QoS decision-making.

Given an active user a and its corresponding uncertain QoS model, the neighbourhood set can be mined by the computation of model similarity between a and each candidate service user u. The uncertain QoS model of a user is composed of a set of corresponding QoS matrices, each of which has different QoS invocation times. Here, each matrix is compressed into a vector by the mean of each column for a QoS criterion in the following:

$$\overline{Val(a,s)} = \frac{\sum\limits_{j=1}^{n} V(q_{ji})}{n} \tag{5}$$

Where n counts the total number of invocation times of web service s by user a, $V(q_{ji})$ is the value of jth invocation on ith QoS criterion.

By doing so, we modify Euclidean distance that is employed to calculate the similarity between an active user a and a candidate neighbourhood user u as below:

$$Sim(a,u) = \sum_{s \in (S(a) \cap S(u))} \frac{N_a(s) + N_u(s)}{N(S(a) \cap S(u))} * \frac{1}{1 + \sqrt{\left(\overline{Val(a,s)} - \overline{Val(u,s)}\right)^2}} \tag{6}$$

where $S(a)$ and $S(u)$ represent services a and u have invoked, respectively. $S(a) \cap S(u)$ denotes services that both a and u have invoked, and $N(S(a) \cap S(u))$ counts the total number of invocation times on services in $S(a) \cap S(u)$. $N_a(s)$ calculates the invocation times on s invoked by a. $\overline{Val(a,s)}$ represents the compressed QoS on s invoked by a.

Although the similarity by the revised Euclidean distance computation can be calculated by using the difference of co-invoked web services between two users, it may still incur overestimation on the similarity of two users, when there are very few co-invoked web services. Based on this observation, it can be improved by a correlation significance weighting factor that can devalue the overestimated similarity. To achieve this goal, it is adjusted to calculate the similarity between an active user a and a candidate neighbourhood user u as follows:

$$Sim^{'}(a,u) = \frac{2 * |S(a) \cap S(u)|}{|S(a) \cup S(u)|} * Sim(a,u) \tag{7}$$

where $|S(a) \cup S(u)|$ denotes the number of web services invoked by either a and u, while $|S(a) \cap S(u)|$ is the number of web services invoked by both a and u. After the adjustment, if an active user provides more QoS transaction logs, it is probably associated with more accurate neighbourhood set. Along this way similar neighbourhood set can be identified by the traditional Top-K algorithm by the following equation:

$$T(a) = \{u | u \in U \wedge u \in Top\text{-}K(a), u \neq a\} \tag{8}$$

where $T(a)$ is s set of top K similar neighbourhood users for the active user a.

4.4 Web services clustering

In addition to the computation of the similar neighbourhood set for an active user, clustering web services can further optimise the QoS prediction and make its predicted value closer to real-world QoS transaction logs during the invocation of web services. Our proposed clustering algorithm is described in the following Algorithm 1. It originates from hierarchical clustering.

Algorithm 1: uncertain web services hierarchical clustering

Input: a set of functionally similar web services S, cluster number l;
Output: service clusters $C = \{c_1, c_2, ..., c_l\}$;
$N \leftarrow sizeof(S)$;
foreach $i \in N$ **do**
  assign each service s_i to a single cluster c_i;
  add c_i into C;
while $N \neq l$ **do**
  find the distance $dis(c_i, c_j) = min_{c_i, c_j \in C}\{dis(c_i, c_j)\}$;
  merge c_i, c_j into c_i and update QoS performance of new c_i;
  C.remove(c_j);
  $N \leftarrow N - 1$;
return C;

To cluster web services with uncertain QoS, we apply hierarchical clustering algorithm to partition web services into several clusters based on the average QoS performance on each web service by using the historical QoS transaction logs. For an uncertain web service s, the average QoS performance can be calculated using its corresponding QoS model by the following equation:

$$Per(s) = \frac{\sum\limits_{u \in U_s} \overline{Val(u,s)}}{|U_s|} \tag{9}$$

where $Per(s)$ denotes average QoS performance of s, U_s is a set of users each of which has invoked s, and $\overline{Val(u,s)}$ represents average QoS observed by u on s.

Based on the average QoS performance of uncertain web services, they can be partitioned into different clusters, where those web services in the same cluster have very similar QoS performance. That is, for any two uncertain web services s_i and s_j in a cluster, their distance $dis(s_i, s_j)$ must be as small as possible. Here, the distance $dis(s_i, s_j)$ between any two services s_i and s_j is evaluated by their average QoS performance:

$$dis(s_i, s_j) = |Per(s_i) - Per(s_j)| \tag{10}$$

By using the QoS distance between any two uncertain web services, hierarchical clustering algorithm is applied for partitioning uncertain web services into a set of service clusters. It builds a hierarchy of uncertain service clusters organised as a service tree.

More specifically, in the algorithm we first initialise each service into a single cluster. Then, the QoS distance between every two service is measured by the equation in (10). Afterwards, by the calculation of QoS distance, the nearest two services or clusters are merged into the same new cluster, where the QoS performance value is recomputed and updated by the equation in (9). The process of cluster mergence is repeated by the above steps until the cluster number does not change any more. Finally, all the web services in S are clustered in $C = \{c_1, c_2, ..., c_l\}$.

Note that, the mean value of QoS performance is calculated to represent the new cluster's average QoS performance when two services with the minimum QoS distance are merged into a new cluster. After several rounds of iterations with the combination of web services, the average QoS value of all the services within a cluster is appointed to represent the QoS performance of that service cluster.

4.5 Uncertain QoS prediction

Uncertain QoS prediction depends on two factors, including historical QoS transactions logs provided by an active user and the offering from similar neighbourhood set of the user. To accurately predict QoS on a target service, two kinds of QoS prediction strategies have been proposed for an active user, called U-Rec and UC-Rec, respectively.

Given an active user a, the fundamental strategy U-Rec for predicting the missing QoS on a target web service s can be expressed by the following equation:

$$\begin{aligned} P_{u-rec}(a,s) =& \lambda * \frac{\sum_{s_a \in S(a)} \overline{Val(a, s_a)}}{N(S(a))} \\ &+ (1-\lambda) * \frac{\sum_{u \in T(a)} \overline{Val(u,s)} * Sim'(a,u)}{\sum_{u \in T(a)} Sim'(a,u)} \end{aligned} \tag{11}$$

where $S(a)$ represents those web services that a have invoked, $N(S(a))$ stands for the total number of services in $S(a)$. Here, an adjustable parameter λ is used to balance the predicted QoS generated from transaction logs invoked by a and those by similar neighbourhood users. λ determines the weighting of the hybrid strategy that relies on historical invocation records of a and the QoS from similar neighbourhood users.

Although U-Rec can effectively predict the missing QoS, we observe that some of the services invoked by an active user may lead to negative influence on the decision-making of QoS prediction, when they have less correlation with the target service in terms of QoS transaction logs. Along with this observation, we make a hypothesis that to predict the QoS for an active user on a target service, those services that belong to the same service cluster and were invoked by the active user play a positive influence, while remaining services invoked by the active user decrease the prediction accuracy. Based on this idea, uncertain QoS prediction strategy UC-Rec can be categorised by two application scenarios.

On one hand, when the active user has never invoked any other services in the cluster that the target service belongs to, the average QoS of web services in this cluster is calculated as the prediction QoS of historical transaction logs on active user. It is formalised as below:

$$P_{uc-rec}(a,s) = \lambda * Ave(c(s)) + (1-\lambda) * \frac{\sum\limits_{u \in T(a)} \overline{Val(u,s)} * Sim^{'}(a,u)}{\sum\limits_{u \in T(a)} Sim^{'}(a,u)} \tag{12}$$

where $Ave(c(s))$ denotes the average QoS performance of the cluster c that the target service s is partitioned into.

However, when the intersection set between web services invoked by the active user and those in the same cluster that the target service belongs to is not empty, QoS prediction on the target service invoked by the active user is calculated as below:

$$\begin{aligned} P_{uc-rec}(a,s) =& \lambda * \left(\frac{\sum\limits_{s_a \in (S(a) \cap S(c(s)))} \overline{Val(a,s_a)}}{N(S(a) \cap S(c(s)))}\right) \\ &+ (1-\lambda) * \frac{\sum\limits_{u \in T(a)} \overline{Val(u,s)} * Sim^{'}(a,u)}{\sum\limits_{u \in T(a)} Sim^{'}(a,u)} \end{aligned} \tag{13}$$

where $c(s)$ represents the cluster that the target service s belongs to, and $S(c(s))$ denotes those web services in cluster $c(s)$, while $S(a)$ represents those web services that a have invoked. $S(a) \cap S(c(s))$ represents those services both in $S(c(s))$ and $S(a)$, where $S(a) \cap S(c(s)) \neq \varnothing$.

5 Experimental evaluation

Based on large-scale uncertain QoS of web services, we have made a comprehensive study on the performance of QoS prediction and conduct experiments to compare the accuracy between two prediction strategies of U-Rec and UC-Rec.

5.1 Experimental setup and datasets

In order to evaluate the effectiveness of our proposed approach and compare its performance, we developed a prototype system for uncertain service QoS prediction where all of the components together with graphical user interface (GUI) have been implemented by IDE MyEclipse and Java programming language. All of the experiments are performed on a PC with Intel Pentium(R) dual-core processor 2.4 GHz and 1G RAM. All experimental programs are run on Windows 7 operating system.

The experiments are based on a public real-world large-scale web service uncertain QoS dataset, which is called WS-DREAM and collected by Zheng and Lyu (2012). It contains more than 1.5 million web service invocation records originated from 100 web services invoked by 150 users distributed in more than 20 countries. For each user, there are 100 RTT profiles, and each profile contains the RTT records of 100 services, while each service has about 100 invocation records. The dataset is illustrated Figure 4. For more detailed information on RTT profile of a user, which is shown in Table 1.

Figure 4 Geographical distributions on users and web services in uncertain QoS dataset (see online version for colours)

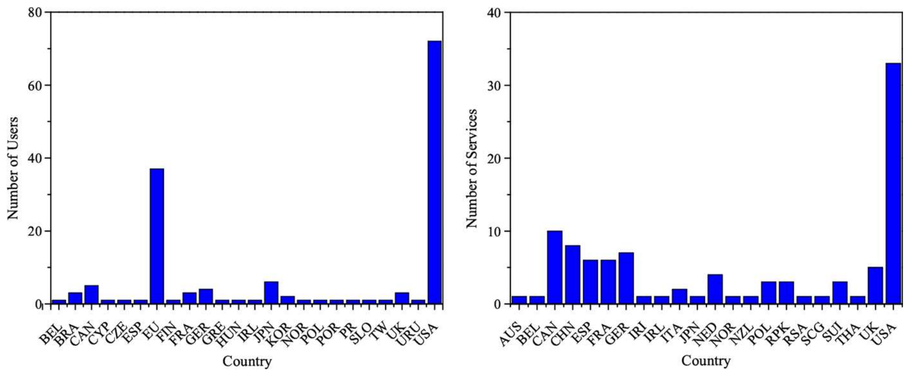


Table 1 An example of RTT profile of a user

UserIP	*WSID*	*Response time*	*DataSize*	*HttpCode*	*HttpMessage*
35.9.27.26	1,521	1,101	1457	200	Bad types
35.9.27.26	1,521	2,032	1457	200	Bad types
35.9.27.26	1,521	1,408	1457	200	Bad types
35.9.27.26	6,405	415	620	200	OK
35.9.27.26	6,405	470	620	200	OK
35.9.27.26	8,953	20,009	2624	−1	Timed out
35.9.27.26	8,953	20,002	2624	−1	Timed out
35.9.27.26	8,953	20,024	2624	−1	Timed out

5.2 Evaluation metrics

In the experiments, mean absolute error (MAE) and root mean square error (RMSE) are used as the evaluation metrics to measure the accuracy of QoS prediction. MAE is the average absolute deviation of QoS prediction to the ground truth data, which is defined as below:

$$MAE = \frac{\sum_{i,j} \left| R_{i,j} - \hat{R}_{i,j} \right|}{N}. \tag{14}$$

where $R_{i,j}$ denotes the QoS of a web service j invoked by a user i, $\hat{R}_{i,j}$ is the predicted QoS, and N is the number of predicted QoS values. MAE stands for the average over the verification samples of the absolute QoS error between those predicted QoS values and their corresponding observations. Since MAE is linear to the deviation of QoS prediction, all the individual differences are weighted equally in the average. It is obvious that smaller MAE is, better QoS prediction accuracy it indicates. Based on the definition of MAE, a revised evaluation metric RMSE is as below:

$$RMSE = \sqrt{\frac{\sum_{i,j} \left(R_{i,j} - \hat{R}_{i,j}\right)^2}{N}}. \tag{15}$$

In the evaluation metric RMSE, the deviations between those predicted QoS and their corresponding observed QoS are squared and averaged over the samples. Finally, it calculates the square root of the average. Since the errors are squared before they are averaged, RMSE has relatively high weighting to large errors. Therefore, it is more applicable to those scenarios where large errors are particularly undesirable.

5.3 Experimental results and analysis

In order to validate the effectiveness of our proposed approach of uncertain QoS prediction, we investigated the state-of-the-art research on uncertain web services. There have been a few number of works on uncertain QoS of web services in recent years. In terms of service discovery and selection, the authors (Wang et al., 2011) employed cloud model to measure the QoS uncertainty for pruning redundant web services, while they extracted reliable services and applied mixed integer programming to choose the optimum web services. In addition, they authors (Benouaret et al., 2012) tackled the problem of service skyline on uncertain QoS by using a possibility distribution and proposed two skyline extensions on uncertain QoS, called pos-dominant skyline and nec-dominant skyline. Based on this, efficient algorithms were developed to compute both the pos-dominant skyline and nec-skyline. As for web service composition, the authors (Sun et al., 2014) exploited information theory and variance theory to abandon those web services with high QoS uncertainty. As a result, they downsized the solution space which can promote the dynamic composition of web services, by using a reliability fitness function to select the best reliable services. Moreover, the authors (Niu et al., 2016) modelled service composition problem with uncertainty (U-WSC) as an artificial intelligence automated planning problem. Two algorithms UCLAO* and BHUC were presented to solve the U-WSC planning problem with state space reduction.

However, all of the above research works do not aim at solving the problem of uncertain QoS prediction of web services. Thus, on the basis of these existing research works, we mainly design the experiments by comparing the accuracy of our proposed two strategies on uncertain QoS prediction of web services.

We divide the 150 service users into two groups, one as training users and the other for active users. For the training users, we randomly remove appointed percentage of entries to make the matrix sparse with different density (e.g. 10%). For an active user, we also randomly remove different number of entries and designate the number of remaining entries as given number. To study the performance of QoS prediction, we compare the two proposed prediction strategies U-Rec and UC-Rec. The experimental results are summarised in Table 2.

Table 2 MAE and RMSE comparison between two QoS prediction strategies U-Rec and UC-Rec

Metrics	*Density (%)*	*QoS prediction strategies*	*The number of training users = 100*		
			Response time		
			G10	*G20*	*G30*
MAE	10	U-Rec	647.727	627.468	616.7005
		UC-Rec	553.052	526.657	511.9068
	20	U-Rec	592.267	568.884	555.0007
		UC-Rec	505.067	471.971	455.4344
	30	U-Rec	551.228	522.276	514.583
		UC-Rec	466.557	425.526	413.3797
RMSE	10	U-Rec	1238.991	1197.284	1170.223
		UC-Rec	1040.994	980.388	944.128
	20	U-Rec	1124.349	1072.065	1041.019
		UC-Rec	930.892	857.427	820.257
	30	U-Rec	1032.583	967.845	950.642
		UC-Rec	843.607	751.846	725.3874

In the experimental results of QoS prediction on response time, we employed 10, 20 and 30% density distributions on all the training matrices. The number of invoked web services varies from 10, 20 to 30 by randomly removing those invoked services by active users, called G10, G20 and G30, respectively. For the other three parameters, we set λ=0.3 as the weighting of historical transaction logs invoked by active user, set Top-K=20 as the number of users when mining similar neighbourhood users, and set the number of service clusters as 25. Towards reliable error estimation, each metrics value of MAE or RMSE is calculated by the mean of 10 times throughout the experiments to evaluate the accuracy of QoS prediction of web services.

From the experimental results, MAE and RMSE become smaller and smaller along with the increase of given number from 10 to 30, which demonstrates that the accuracy of QoS prediction can be improved by providing more historical invoked web services. Furthermore, As the increasing distributions of uncertain QoS density from 10, 20 to 30%, the accuracy of QoS prediction can also be significantly enhanced. The reason is that training matrices with more uncertain QoS can lead to the positive influence on the QoS prediction. In terms of the comparison between the two prediction strategies, the accuracy of UC-Rec outperforms that of U-Rec. The reason is that in UC-Rec those potential negative influences on the unprofitable QoS transaction logs provided by the active user have been removed by clustering web services with uncertain QoS.

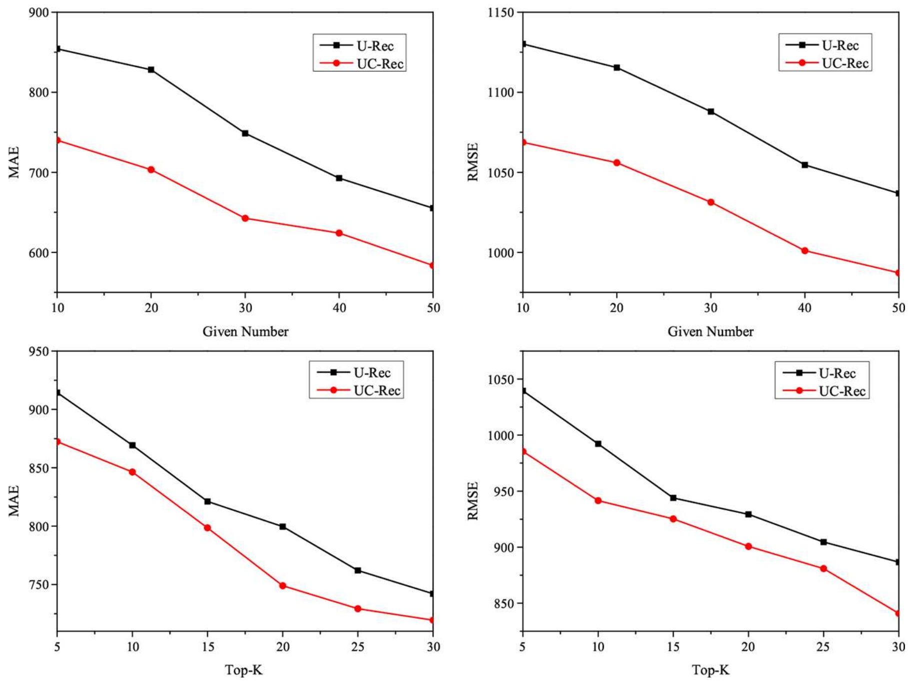
5.4 Impact of parameters

To make a deeper analysis on the impact of parameters in our experiments, we investigate four kinds of parameters to discuss its QoS prediction accuracy on MAE and RMSE, including Given Number, Top-K, Cluster Number and λ.

5.4.1 Impact of given number and Top-K

For the comparison of given number on the prediction accuracy in the experiments, 100 training users have been used. In addition, a set of parameters are assigned as $\lambda = 0.3$, the density of uncertain QoS matrix = 10%, Top-K = 10 and cluster number = 10. Under these parameters setting, the given number varies from 10 to 50 with an interval of 10. The trends of MAE and RMSE on given number are illustrated in Figure 5. It is concluded that the QoS prediction accuracy of two prediction strategies U-Rec and UC-Rec grows with the increasing given number of web services. In fact, it validates the hypothesis that active users are likely to receive better prediction accuracy when they accumulated more historical transaction logs of uncertain QoS.

Figure 5 The experimental results of MAE and RMSE along with the changes of given number and Top-K (see online version for colours)

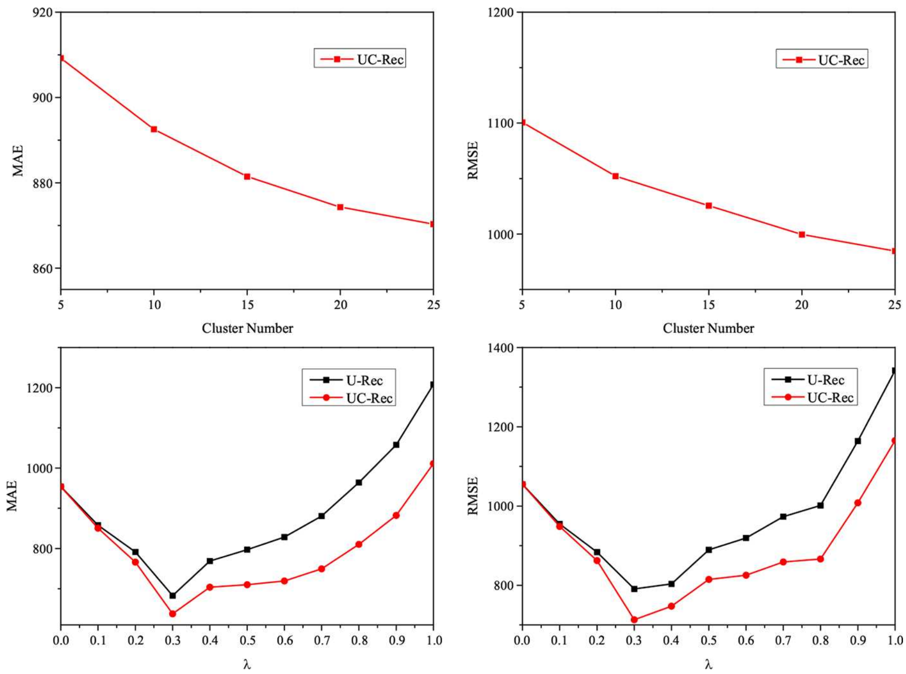


As for the Top-K comparison between U-Rec and UC-Rec, the influence of prediction accuracy is correlated with how many similar neighbourhood users are employed for QoS prediction. We appoint the number of training users as 100, λ = 0.3, given number=10, density of uncertain QoS matrix = 10%, and cluster number = 10. The size of Top-K similar neighbourhood users varies from 5 to 30 with an interval of 5. The trends of MAE and RMSE on Top-K are illustrated in Figure 5. From the experimental results, we observe that the two curves on MAE and RMSE decrease along with an increasing size of neighbourhood users.

5.4.2 Impact of cluster number and λ

The cluster number of web services plays an important role in the QoS prediction of UC-Rec. In order to study the impact of cluster number on the QoS prediction accuracy, we set the number of training users as 100, $\lambda = 0.5$, given number = 10, Top-K = 10 and density of uncertain QoS matrix = 10%. Under these parameters setting, cluster number varies from 5 to 25 with an interval of 5. Figure 6 shows the experimental results on MAE and RMSE of UC-Rec. It demonstrates that the prediction accuracy can be improved along with the increase of cluster number.

Figure 6 The experimental results of MAE and RMSE along with the changes of cluster number and λ (see online version for colours)



Finally, λ is applied for balancing the weighting between the fundamental QoS prediction from historical QoS transaction logs of an active user and that from similar neighbourhood users. Especially, if λ is assigned to 1, we only make QoS prediction from the historical uncertain QoS of active user; if λ is appointed as 0, we predict QoS from the transaction logs of similar neighbourhood users. To study the impact of λ on U-Rec and UC-Rec, we set Top-K = 10, given number = 10, density of uncertain QoS matrix = 10%, the number of training users = 100 and cluster number = 10. The parameter of λ varies from 0 to 1 with an interval of 0.1. The experimental results are illustrated in Figure 6. It concludes that when λ is appointed to 0.3, U-Rec and UC-Rec can both achieve the best prediction accuracy. Consequently, we find that λ will significantly influence the accuracy of QoS prediction, which can be dynamically chosen according to different application scenarios with a set of parameters setting.

6 Related work

In recent years, service recommendation of QoS prediction has received considerable attentions using the techniques of collaborative filtering. Generally, QoS-based service recommendation can be divided into traditional CF-based approaches and context-aware ones.

The authors (Shao et al., 2007) proposed a novel approach for personalised QoS prediction of web services by the technique of collaborative filtering. Their approach predicts QoS values for web services by calculating the similarity among user's experiences under the assumption that users who have similar experiences on some services would have similar experiences on other services. The authors (Zheng et al., 2009) presented a hybrid approach which combines user-based and item-based approach together to predict the QoS of web services. More specifically, they employ two confidence weights to balance these two predicted QoS values. Based on this work, a model based CF algorithm is then developed that achieves higher prediction accuracy (Zheng et al., 2013). they applied the user-based approach as a precursor to identify top-k similar users. By using neighbourhood information, matrix factorisation is employed to construct a global model, which predicts unobserved QoS of web services.

A hybrid personalised CF-based recommendation approach (Jiang et al., 2011) was proposed. They predicted QoS and recommended desired web services for an active user using similarity calculation between service users. They evaluated the contribution of a web service by the standard deviation of QoS metrics. The authors (Abdullah and Li, 2015) proposed a QoS-based integrated-model graph, where users and services are represented as the nodes while weighted QoS magnitudes and user/service similarity measurements serve as the edges. Top-k random walk algorithm is applied to generate final recommendation list to active users. Moreover, the authors (Lee and Ko, 2016) proposed a user-based collaborative filtering approach for service composition. They consider member organisation for a new user group and select neighbourhood user groups that are similar to the new group by the combinations of member organisation-based group similarity (MOGS) metrics. In addition to the traditional CF-based service recommendation, many researchers also solve the issue with the consideration of application context.

From the perspective of geographical information, the authors (Chen et al., 2010) took advantage of the correlation between QoS transaction logs and user locations. They proposed a collaborative filtering algorithm for service recommendation, called RegionKNN. It incorporates user location with the notion that service users with similar IP address are likely to have similar QoS invocation records. Afterwards, the authors (Tang et al., 2012) proposed a method of location-aware collaborative filtering approach to recommend services by incorporating locations from both users and services. With the consideration of temporal factors, the authors (Yu and Huang, 2014) demonstrated that QoS performance of services is highly related to service status and network environment which are variable against time. Thus, they proposed a time-aware collaborative filtering algorithm by taking time factor into account, when computing the similarity between users and services to predict missing QoS values. In addition, a novel personalised QoS prediction approach (Hu et al., 2015) was proposed with the consideration of both the temporal dynamics of QoS attributes and the factors of service users by seamlessly integrating collaborative filtering with improved time series forecasting. Furthermore, the authors (Chen et al., 2016) proposed a time-aware collaborative poisson factorisation (TCPF) approach for service recommendation. It takes poisson factorisation as the theory to separately model mashup service demands and service description, which are incorporated together with historical usage data by collective matrix factorisation.

Under the motivation of above related work on QoS-based prediction and recommendation of web services, this paper solves the research issue of QoS prediction with uncertainty of web services. We model the QoS uncertainty of an active user as a three-layer tree. By doing so, we propose a novel hierarchical clustering-based approach for uncertain QoS prediction of web services using collaborative filtering.

7 Conclusion and future work

Traditional non-functional service recommendation mainly focuses on QoS prediction by collaborative filtering. It has not taken the characteristic of web services with QoS uncertainty into account in real-world service-oriented applications. Under this motivation, we proposed a novel approach to uncertain QoS prediction via collaborative filtering and service clustering. First, uncertain QoS is modeled for a service user by a three-layer tree, where each service is formalised as a QoS matrix. Then, we calculate the similarity of two uncertain QoS models by using modified Euclidean distance to mine similar neighbourhood users for an active user. Finally, two uncertain QoS prediction strategies called U-Rec and UC-Rec are respectively proposed based on collaborative filtering and service clustering. Extensive experiments are conducted in large-scale uncertain QoS dataset which consists of more than 1.5 million historical QoS transaction logs. The experimental results validate the effectiveness of our approach for uncertain QoS prediction of web services.

Since most of existing works pay attention to web service selection and composition, in this paper we mainly focus on the experiments by comparing our proposed two uncertain QoS prediction strategies. Therefore, in the future work we plan to compare it with the upcoming collaborative filtering based approaches for uncertain QoS prediction and evaluate their accuracy on more emerging uncertain QoS datasets as soon as they become available. Furthermore, we would like to investigate how temporal sequence of QoS transaction logs and geographic location of service users can be integrated into the approach to improve the accuracy of uncertain QoS prediction. As a result, the proposed approach could be potentially applied in context-aware mobile service recommender systems.

Acknowledgements

This work was partially supported by the Fundamental Research Funds for the Central Universities (16D111208), the Shanghai Natural Science Foundation (17ZR1400200), the National Natural Science Foundation of China (61303096, 61300100), and platform fund of PAPD and CICAEET.

We would like to appreciate all of the two anonymous reviewers for their insightful suggestions and constructive comments that has significantly improved the quality of our manuscript. We also thank Prof. Zibin Zheng from Sun Yat-sen University for providing the large-scale uncertain QoS dataset with more than 1.5 million invocation transactions of web services for our extensive experiments.

References

Abdullah, A. and Li, X. (2015) 'An integrated-model QoS-based graph for Web service recommendation', in *Proceedings of the IEEE International Conference on Web Services*, June 27–July 2, 2015, IEEE, New York, NY, USA, pp.416–423.

Benouaret, K., Benslimane, D. and Hadjali, A. (2012) 'Selecting skyline Web services from uncertain QoS', in *Proceedings of the IEEE International Conference on Services Computing*, 24–29 June, 2012, IEEE, Honolulu, HI, USA, pp.523–530.

Chen, X., Liu, X. and Huang, Z. (2010) 'RegionKNN: a scalable hybrid collaborative filtering algorithm for personalized Web service recommendation', in *Proceedings of the IEEE International Conference on Web Services*, 5–10 July, 2010, IEEE, Miami, FL, USA, pp.9–16.

Chen, S., Fan, Y., Tan, W., Zhang, J., Bai, B. and Gao, Z. (2016) 'Time-aware collaborative poisson factorization for service recommendation', in *Proceedings of the IEEE International Conference on Web Services*, 27 June–2 July 2016, IEEE, San Francisco, CA, USA, pp.196–203.

Czitrom, V. and Spagon, D. (1997) 'Statistical case studies for industrial process improvement', *SIAM*, Philadelphia.

Feng, Y., Ngan, L. and Kanagasabai, R. (2013) 'Dynamic service composition with service-dependent QoS attributes', in *Proceedings of the IEEE International Conference on Web Services*, 28 June–3 July, 2013, IEEE, Santa Clara, CA, USA, pp.10–17.

Hadad, J., Manouvrier, M., Ramirez, G. and Rukoz, M. (2008) 'QoS-driven selection of web services for transactional composition', in *Proceedings of the IEEE International Conference on Web Services*, 23–26 September, 2008, IEEE, Beijing, China, pp.653–660.

Hadad, J., Manouvrier, M., Ramirez, G. and Rukoz, M. (2010) 'TQoS: transactional and QoS-aware selection algorithm for automatic web service composition', *IEEE Transactions on Services Computing*, Vol. 3, No. 1, pp.73–85.

Hu, Y., Peng, Q., Hu, X. and Yang, R. (2015) 'Web service recommendation based on time series forecasting and collaborative filtering', in *Proceedings of the IEEE International Conference on Web Services*, June 27–July 2, 2015, IEEE, New York, NY, USA, pp.233–240.

Jiang, Y., Liu, J., Tang, M. and Liu, X. (2011) 'An effective Web service recommendation method based on personalized collaborative filtering', in *Proceedings of the IEEE International Conference on Web Services*, 4–9 July, 2011, IEEE, Washington, DC, USA, pp.211–218.

Lee, J. and Ko, I. (2016) 'Service recommendation for user groups in internet of things environments using member organization-based group similarity measures', in *Proceedings of the IEEE International Conference on Web Services*, June 27–July 2, 2016, IEEE, San Francisco, CA, USA, pp.276–283.

Li, Y., Haihong, E., Song, M. and Song, J. (2015) 'User familiar degree aware recommender system', in *Proceedings of the IEEE International Conference on Web Services*, June 27–July 2, 2015, IEEE, New York, NY, USA, pp.385–391.

Niu, S., Zou, G., Gan, Y., Zhou, Z. and Zhang, B. (2016) 'UCLAO* and BHUC: Two novel planning algorithms for uncertain Web service composition', in *Proceedings of the IEEE International Conference on Services Computing*, June 27–July 2, 2016, IEEE, San Francisco, CA, USA, pp.531–538.

Shao, L., Zhang, J., Wei, Y. and Zhao, J. (2007) 'Personalized QoS prediction for Web services via collaborative filtering', in *Proceedings of the IEEE International Conference on Web Services*, July 9–13, 2007, IEEE, Salt Lake City, UT, USA, pp.439–446.

Sun, H., Zheng, Z., Chen, J. and Lyu, M.R. (2011) 'NRCF: A novel collaborative filtering method for service recommendation', in *Proceedings of the IEEE International Conference on Web Services*, July 4–9, 2011, IEEE, Washington, DC, USA, pp.702–703.

Sun, H., Zheng, Z., Chen, J. and Lyu, M.R. (2013) 'Personalized Web service recommendation via normal recovery collaborative filtering', *IEEE Transactions on Services Computing*, Vol. 6, No. 4, pp.573–579.

Sun, L., Wang, S., Li, J., Sun, Q. and Yang, F. (2014) 'QoS uncertainty filtering for fast and reliable Web service selection', in *Proceedings of the IEEE International Conference on Web Services*, June 27–July 2, IEEE, Anchorage, AK, USA, 2014, pp.550–557.

Tang, M., Jiang, Y., Liu, J. and Liu, X. (2012) 'Location-aware collaborative filtering for QoS-based service recommendation', in *Proceedings of the IEEE International Conference on Web Services*, June 24–29, 2012, IEEE, Honolulu, HI, USA, pp.202–209.

Tian, G., Wang, J., He, K. and Sun, C. (2014) 'Time-aware Web service recommendations using implicit feedback', in *Proceedings of the IEEE International Conference on Web Services*, June 27–July 2, 2014, IEEE, Anchorage, AK, USA, pp.273–280.

Wang, S., Zheng, Z., Sun, Q., Zou, H. and Yang, F. (2011) 'Cloud model for service selection', in *Proceedings of the IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, April 10–15, 2011, IEEE, Shanghai, China, pp.666–671.

Wei, L., Yin, J., Deng, S. and Li, Y. (2012) 'Collaborative Web service QoS prediction with location-based regularization', in *Proceedings of the IEEE International Conference on Web Services*, June 24–29, 2012, IEEE, Honolulu, HI, USA, pp.464–471.

Yilmaz, A. and Karagoz, P. (2014) 'Improved genetic algorithm based approach for QoS aware Web service composition', in *Proceedings of the IEEE International Conference on Web Services*, June 27–July 2, 2014, IEEE, Anchorage, AK, USA, pp.463–470.

Yu, C. and Huang, L. (2014) 'Time-aware collaborative filtering for QoS-based service recommendation', in *Proceedings of the IEEE International Conference on Web Services*, June 27–July 2, 2014, IEEE, Anchorage, AK, USA, pp.265–272.

Zheng, Z. and Lyu, M. (2012) 'Collaborative reliability prediction of service-oriented systems', in *Proceedings of the 32nd ACM/IEEE International Conference on Software Engineering*, May 1–8, 2010, ACM/IEEE, Cape Town, South Africa, pp.35–44.

Zheng, Z. and Lyu, M. M. (2013) '*QoS Management of Web Services*', Springer-Verlag, Berlin Heidelberg.

Zheng, Z., Ma, H., Lyu, M.R. and King, I. (2009) 'WSRec: a collaborative filtering based Web service recommender system', in *Proceedings of the IEEE International Conference on Web Services*, July 6–10, 2009, IEEE, Los Angeles, CA, USA, pp.437–444.

Zheng, Z., Ma, H., Lyu, M.R. and King, I. (2011) 'QoS-aware Web service recommendation by collaborative filtering', *IEEE Transactions on Services Computing*, Vol. 3, No. 4, pp.140–152.

Zheng, Z., Ma, H., Lyu, M.R. and King, I. (2013) 'Collaborative Web service QoS prediction via neighborhood integrated matrix factorization', *IEEE Transactions on Services Computing*, Vol. 6, No. 3, pp.289–299.