This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TSC.2017.2782793

CASR-TSE: Context-aware Web Services Recommendation for Modeling Weighted Temporal-Spatial Effectiveness

Xiaoliang Fan, *Member, IEEE,* Yakun Hu, Zibin Zheng, *Senior Member, IEEE,* Yujie Wang, Patrick Brézillon, *and* Wenbo Chen

Abstract—Recent years have witnessed the growing research interest in the Context-Aware Recommender System (CARS). CARS for Web service provides opportunities for exploring the important role of temporal and spatial contexts, separately. Although many CARS approaches have been investigated in recent years, they do not fully address the potential of temporal-spatial correlations in order to make personalized recommendation. In this paper, the Context-Aware Services Recommendation based on Temporal-Spatial Effectiveness (named CASR-TSE) method is proposed. We first model the effectiveness of spatial correlations between the user's location and the service's location on user preference expansion before the similarity computation. Second, we present an enhanced temporal decay model considering the weighted rating effect in the similarity computation to improve the prediction accuracy. Finally, we evaluate the CASR-TSE method on a real-world Web services dataset. Experimental results show that the proposed method significantly outperforms existing approaches, and thus it is much more effective than traditional recommendation techniques for personalized Web service recommendation.

Index Terms—context awareness, Web services, recommender system, QoS, temporal and spatial effects

1 INTRODUCTION

Web services have been considered as building blocks for the Service-Oriented Computing (SOC) paradigm for decades. In recent years, a large number of Internet applications have been constructed by the combination of Web services, raising Web service recommendation as an important and challenging task. Specifically, users expect to explore functionally equivalent Web services that also satisfy their personal non-functional requirements, such as personal preferences and interests. Under these conditions, personalized Web service recommendation that incorporates the non-functional requirements of users has aroused a great deal of interest in the services computing field [1], [2].

A Context-Aware Recommender System (CARS) is aiming at recommending items similar to the ones already rated with the highest score by the users. Moreover, CARS for Web services provides opportunities to incorporate contextual information into Web service recommendations [5]. Previous work has demonstrated the benefits of recommending Web services by considering various contextual factors [6], [7], [8]. Specifically, to provide personalized recommendations [39], [40], several methods extract temporal [9], [10], [11], [12], spatial [13], [14] and social [15], [26] contexts from Web service invocation records.

However, existing CARS approaches do not fully address the potential of temporal-spatial correlations when making personalized recommendation. Our first claim is that, in terms of spatial correlations, existing work [8, 46, 54] pays inadequate attention to the spatial correlations on user preference expansion. Most work mainly focuses on changes of either users' location [8] or services' location [46], while ignoring the correlations between user's spatial context and service's spatial context. The second claim is that, in terms of temporal correlations, several approaches have investigated the temporal decay models [10, 35] in the similarity computation. However, they merely considered each QoS value equally during the similarity computation, which might possibly neglect the weighted rating effect from different ratings.

In this study, we hypothesize that the accuracy of Web service recommendation can be improved by incorporating a significance-weighted rating factor into exploring the temporal-spatial effectiveness. The major contributions of this paper are threefold:

- We model the effectiveness of spatial correlations on user preference expansion. The method takes into account the dynamic characteristics of geographical location for both the user and the service, in order to apply the personalized filter of the services before the similarity computation.

- *Xiaoliang Fan is with Fujian Key Laboratory of Sensing and Computing for Smart City, Xiamen University, Xiamen, China, and with School of Information Science and Engineering of Lanzhou University, Lanzhou, China. E-mail: fanxiaoliang@lzu.edu.cn.*
- *Yakun Hu, Yujie Wang and Wenbo Chen are with School of Information Science and Engineering of Lanzhou University, Lanzhou, China. E-mail: {huyk14, wangyujie15, chenwb} @lzu.edu.cn.*
- *Zibin Zheng is with School of Data and Computer Science, and Key Laboratory of Machine Intelligence and Advanced Computing, Sun Yat-sen University, Ministry of Education, China. E-mail: zhzibin@mail.sysu.edu.cn*
- *Patrick Brézillon is with Laboratoire d'Informatique de Paris 6, University Pierre and Marie Curie, Paris, France. E-mail: patrick.brezillon@lip6.fr.*

This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TSC.2017.2782793

- We propose an enhanced temporal decay model for similarity computation, which incorporates the weighted rating effect into the traditional temporal decay model to improve the prediction accuracy.
- We conduct a set of comprehensive experiments based on a real-world Web service dataset. The results demonstrate that the proposed method significantly outperforms existing Web service recommendation approaches.

The remainder of this paper is organized as follows. Section 2 discusses the related work. Section 3 introduces the motivation of the work. Section 4 proposes the method. Section 5 describes the implementation, experimental results and discussions. Finally, Section 6 concludes this study.

2 Related Work

2.1 Collaborative Filtering

Collaborative filtering (CF) is a method of making predictions by collecting preferences from many collaborating users or items based on the target user or the target item [17], [49], [50], [52]. CF methods consist primarily of two types: model-based and memory-based CF [18]. Memory-based CF methods are further divided into two categories: item-based [19], [20] and user-based [21], [22].

The common similarity computation measurements used in CF are the Pearson Correlation Coefficient (PCC) and Cosine Similarity, which have been widely incorporated into QoS-aware Web service recommendation [37], [38], [39], [40]. Furthermore, various xPCC extensions [4], such as user-based PCC, item-based PCC, and user-item-based PCC, have been widely investigated. In addition, Yao et al. [51] proposed a content-based CF method considering not only semantic data (e.g., functionalities), but also rating data (e.g., QoS) of Web services.

However, QoS attributes of Web services, such as response time and response message, rely much on contextual information, such as network locations, invocation time, availability of services, etc. The traditional CF methods face difficulties in making a personalized Web service recommendation for different users, considering the dynamic characteristics of users and services.

2.2 Context-Aware Web Services Recommendation

The context-aware recommender system (CARS) has been widely employed to explore the significant role of contextual factors for personalized recommendation over the years [5]. Various approaches for obtaining, representing and managing contextual information in recommender systems have been proposed, based on the idea of context as a complicated concept with usually infinite dimension [23], [24].

In context-aware recommendation, three types of contextual information are usually extracted from the invocation records of Web services [42]. The first widely utilized context factor is spatial context [46], or location-aware context. Xiong et al. [36] recognized the influence of preference propagation and proposed a Location-based Matrix Factorization method to address the cold-start issue in QoS prediction. Kuang et al. [8] proposed a context-aware service recommendation model considering historical invocation records at the similar locations of the target user. Tang et al. [46], [54] considered locations of both services and users in order to select similar neighbors for the target user. Similarly, the influence of regional correlations on user's interest was also described in [13], [14]. However, the work considering spatial context mainly finds similar users or Web services based on the target user's or the target service's location, respectively. As a result, they generally speaking ignored correlations between user's spatial context and service's spatial context, which may have a great impact on user preference expansion. For example, when the user's geographical location changes (e.g., go abroad for a conference), the user may prefer Web services near his new location. Similarly, when the service's geographical location changes, the user preference on services would be expanded, because of anomalous events such as server migration. To the best of our knowledge, little attention has been paid to explore the correlations between user's spatial context and service's spatial context.

The second category is the temporal context. For example, Zhang et al. [9] presented the "user-service-time" relations to investigate latent features for recommendation. In addition, a time decay function has been widely used to compute time weights for different services according to the timespan between the historical invocation records and the current request in [10], [35], [49]. Other temporal-based methods have been employed in [11], [43], [44], [45]. However, those approaches merely considering each QoS value (i.e., response time [31]) equally in similarity computation, which could possibly neglect the weighted rating effect from different ratings. For instance, a user will obviously like a Web service S_1 with only 67 milliseconds of response time over another service S_2 with 5,000 milliseconds. Likewise, a very long response time with almost 50,000 milliseconds indicates that the user will dislike a service S_3 at all. As a result, compared to the existing temporal decay models, in this paper, we exploit the weighted effectiveness of differential QoS values in similarity computation, to highlight the significance of both higher and lower QoS values.

Therefore, this work focuses on improving QoS prediction accuracy by exploring the weighted temporal-spatial effectiveness for a personalized Web service recommendation.

3 Motivation

In this section, we explain the motivation of our work according to Fig. 1. Section 3.1 discusses why the effect of spatial correlations on user preference expansion is vital to improve QoS prediction accuracy. Section 3.2 discusses the need for the weighted effectiveness of differential QoS values in the similarity measurement.

3.1 Incorporating Spatial Correlations into User

This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TSC.2017.2782793

Preference Expansion

The scenario of a context-aware Web service recommender system is shown in Fig. 1. The objective is to recommend a weather forecast Web service considering the effect of correlations between user's spatial context and service's spatial context on **user preference expansion**.

Fig. 1 contains three layers: **1) the service layer** lists all available services in a repository. We assume that the repository includes several weather forecast Web services (S_1, S_2, …, S_n). Consider S_1= "NYC Weather[1]", S_2= "US Weather Service[2]"; S_3 = "Moji China Weather[3]"; S_4 = "Le Figaro météo[4]", S_5 = "Weather in China[5]". It is noted that the Web services are distributed globally; **2) the spatial layer** shows the continuously changing position of the target user (u_1). The curves between the service layer and the spatial layers connect services with their corresponding locations. For example, S_1 is deployed in New York City, USA, and S_5 is a weather forecast service from Beijing, China; and **3) the temporal layer** illustrates how the user travels from one location to another over time. For example, u_1 is a scientist at New York University on January 12th, who will go to Beijing for an international conference till January 16th, and finally she will spend a 3-day vacation in Paris.

Normally, due to Internet transfer delay, the user would prefer the nearby services because QoS properties such as response time are largely dependent on the network distance between the user and invoked Web services. In addition, as shown in Fig. 1, a user would prefer to her native service as the neighborhood weather forecasting service is more accurate than others (known as "regional correlation"). As a result, due to the effect of both the network distance and the regional correlation mentioned above, u_1 will prefer S_1 or S_2 when she is at her residence in NYC.

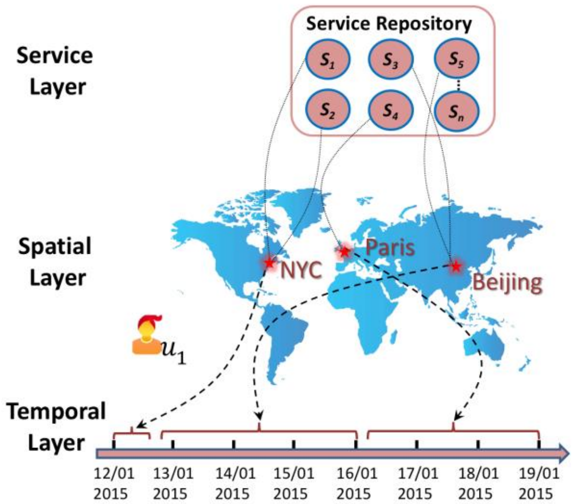

Fig. 1. Weather forecast service recommendation

Furthermore, it is also interesting to note in Fig. 1 that when the geographical location of the user or the service is changed, the user preference will be expanded accordingly. **For one thing**, when the user' geographical location changes (e.g., in Fig. 1, u_1 will go to Beijing for a conference from NYC), it is reasonable for u_1 to prefer S_3 or S_5, instead of S_1 or S_2. Because they are close to her new location. **For another thing**, when the service's geographical location changes (e.g., in Fig. 1, the server that deploys S_1 would migrate to a new location), u_1 would not prefer S_1 due to the server migration.

By taking spatial correlations between users and services into consideration, we can help uncover the real requirement of the target user as well as real-time service QoS, thus improving the accuracy of QoS prediction.

3.2 Incorporating Weighted Rating Effect into Temporal Decay Model

In Fig. 1, u_1 could invoke S_1 for multiple times. When computing the similarity of two users' QoS values, a longer timespan between invocation time and the current time may imply a deviation of QoS value. Many previous methods have coined this phenomenon as temporal decay effect. However, in the similarity computation, existing temporal decay models which consider every QoS value equally could possibly neglect the weighted rating effect from different ratings. For example, as shown in Fig. 1, u_1 will obviously like a Web service S_1 with only 67 milliseconds of response time over another service S_2 with 5,000 milliseconds. Likewise, a very long response time represents a very low QoS, which indicates that u_1 will dislike S_3 at all with almost 50,000 milliseconds. In short, we believe that it is necessary to increase the weighting of both higher and lower QoS values when computing the similarity measurement. In this paper, we coin this as "weighted rating effect".

By introducing the weighted rating effect into traditional temporal decay model, we could infer a set of users with the similar preference for the target user, thus we could make a personalized Web services recommendation.

We elaborate on the proposed CASR-TSE method in the next section.

4 CASR-TSE Method

The design idea of CASR-TSE method is described as follows. First, we introduce one filtering step to get a filtered dataset by modeling the effect of the correlations between user's spatial context and service's spatial context on user preference expansion in Section 4.2. Second, we introduce another filtering step via the enhanced temporal decay model by introducing the weighted rating effect into the traditional temporal decay model in Section 4.3. Finally, based on the filtered training data from both the first and second step, the Bayes model will get the personalized prediction results for a specific user in Section 4.4. Finally,

[1]NYC Weather, http://www1.nyc.gov/site/severeweather/index.page
[2] US weather service, http://www.weather.gov/
[3] Moji China Weather, http://www.moweather.com/
[4] Le Figaro météo, http://www.lefigaro.fr/meteo/france/index.php
[5] Weather in China, http://en.weather.com.cn/

both the spatial and temporal information are integrated to make the personalized recommendation.

Here, we first give the overall architecture of the proposed approach in Fig. 2.

In Fig. 2:

Step 1: Find the active user's preferred Web services based the effect of spatial correlations on user preference expansion;

$\boldsymbol{P_{u,s}}$: The set of the active user's preferred Web services;

PLT1: The set of users' invocation records of Web services included in $\boldsymbol{P_{u,s}}$;

Step 2: Find similar users combing temporal decay model with the weighted rating effect;

T1: the set of similar users generated from **Step 2**;

Step 3: Employ the QoS prediction by Bayesian inference, and finally make the personalized Web service recommendation for the user.

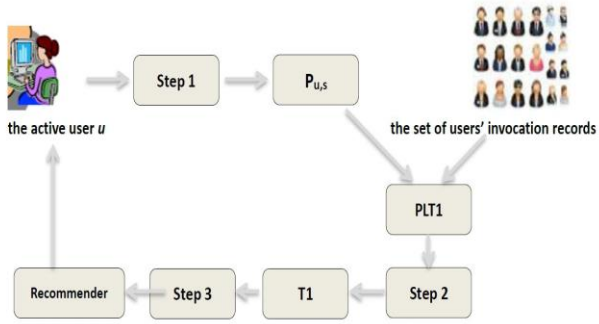

Fig. 2. Overview of our Web service recommendation method

4.1 Definitions and Notations

In the rest of this paper, the following notations and definitions will be used in describing our method.

Suppose that there are a set of service users $\mathrm{U} = \{\mathrm{u}_1, \mathrm{u}_2, \dots, \mathrm{u}_\mathrm{m}\}$ in a CARS system, where $u_i (1 \le i \le m)$ denotes a Web service user, and m denotes the total number of Web service users.

$S = \{s_1, s_2, \dots, s_n\}$ denotes the set of Web services, where $s_k (1 \le k \le n)$ is a Web service.

S_R denotes the set of Web services that are related to region.

$W_{ui,uj} = \{s_1, s_2, \dots, s_k\}$ is a set of Web services that are commonly invoked by service users u_i and u_j.

$R = \{r_{ui,sk}\}$ denotes the set of QoS records on the Web service s_k for the user u_i.

$\bar{R} = \{\bar{r}_1, \bar{r}_2, \dots, \bar{r}_i, \dots, \bar{r}_m\}$ is the set of the mean QoS values of all Web services invoked by $\mathrm{U} = \{\mathrm{u}_1, \mathrm{u}_2, \dots, \mathrm{u}_\mathrm{m}\}$.

$L_U = \{l_{ui}\}(1 \le i \le m)$ is the set of l_{ui}, the network location of the user u_i.

$L_S = \{l_{sk}\}$ is the set of l_{sk}, the network location of the Web service s_k.

$\mathrm{Q} = \{\mathrm{q}_1, \mathrm{q}_2, \dots, \mathrm{q}_\mathrm{k}\}$ denotes a set of QoS properties which record a Web service invocation.

4.2 Modeling the Effect of Spatial Correlations on User Preference Expansion

In order to model the effect of spatial correlation on user Web service preferences, this section begins with an introduction to user or Web service location information.

After reviewing the relevant literatures, we find that users' or Web services' location information can be determined through the IP address if you have an IP address.

First, according to [3], it raises a mechanism for user-collaborated that encourages users to share client data. This mechanism will record the contribution of the user's IP address, so the general server will collect the user's IP address to invoke it.

Second, how can we get the location information of a user or Web service through an IP address? Based on [46], we can represent a user's location as a triple ($\mathrm{IP_U}$, $\mathrm{ASN_U}$, CountryID), where $\mathrm{IP_U}$ denotes the IP address of the user, $\mathrm{ASN_U}$ denotes the ID of the Autonomous System that $\mathrm{IP_U}$ belongs to, and $\mathrm{CountryID_U}$ denotes the ID of the country that $\mathrm{IP_U}$ belongs to. Similarly, we model a Web service's location as (IP, ASN, CountryID), where $\mathrm{IP_S}$ denotes the IP address of the server hosting the service, $\mathrm{ASN_S}$ denotes the ID of the AS that IPS belongs to, and CountryID denotes the ID of the country that $\mathrm{IP_S}$ belong to. Acquiring the location information of both Web services and service users can be easily done. Because the users' IP addresses are already known, to obtain full location in-formation of a user, we only need to identify both the AS and the country in which he is located according to his IP address. A number of services and databases are available for this purpose. In this work, we accomplished the IP to AS mapping and IP to country mapping using the GeoLite Autonomous System Number Database[6]. The database is updated every month, ensuring that neither the IP to AS mapping nor the IP to country mapping will be out-of-date. Acquiring the location information of Web services is similar to acquiring the location information of users. Because the services' URLs or DSNs are already known, only a prior DSN name to IP address translation is required. This is also easy to be implemented.

As mentioned in Section 3.1, there are some region-related Web services, such as weather forecast Web services, whose accuracy depends heavily on the geographical region (a.k.a., regional correlation). Here, we define the $P_{u,RCS}$ as the impact of regional correlation on user preference and obtain

$$\mathrm{P_{u,RCS}} = \begin{cases} 1(s \in S_R) \\ 0(s \notin S_R) \end{cases} \tag{1}$$

where S_R is the set of Web services related to a specific geographical region. When Web service s belongs to S_R, for consistency with the normalized QoS, $P_{u,RCS}$ is set to 1; otherwise, $P_{u,RCS}$ is 0.

For region-unrelated Web services, it is reasonable for users to prefer Web services that are nearby in network distance. $P_{u,NDS}$ is defined to describe the influence of network distance on user preference:

[6] http://www.whois.net

$$P_{u,NDS} = P_0 Dis\left(l_{ui}, l_{sk}\right)_{nor} \tag{2}$$

where P_0 is a constant. For consistency with the normalized QoS, we set P_0 as 1. $Dis(l_{ui}, l_{sk})$ denotes the network distance between the user's network location l_{ui} and the Web service's network location l_{sk}. Furthermore, $Dis(l_{ui}, l_{sk})$ is normalized to $Dis(l_{ui}, l_{sk})_{nor}$ for consistency with the normalized QoS.

Later, we model both $P_{u,RCS}$ and $P_{u,NDS}$ to describe the effect of spatial correlations on user preference expansion. Specifically, we assign different weights to the effects of both regional correlation and network distance (i.e., w_1 to $P_{u,RCS}$, and w_2 to $P_{u,NDS}$). As a result, the effect of spatial correlations on user preference expansion $P_{u,S}$ can be described as follows:

$$P_{u,S} = w_1 P_{u,RCS} + w_2 P_0 \mathrm{Dis}\left(l_{ui}, l_{sk}\right)_{\mathrm{nor}} \tag{3}$$

Finally, we can use $P_{u,S}$ to obtain the invocation records of Web services that correspond to the current user's preference, by providing the personalized service filtering before the similarity computation.

4.3 Enhanced Temporal Decay Model with Weighted Rating Effect

Existing CARS approaches [35] introduce the temporal decay effect of Web service invocation into similarity computation techniques, such as Pearson Correlation Coefficient. However, traditional PCC methods pay little attention to an important factor: the weighted rating effect, as detailed in Section 3.2.

Based on the existing temporal decay model [16], an enhanced temporal decay model incorporating a weighted rating effect is proposed in this section. In the below, we introduce this enhanced temporal decay model with weighted rating effect in detail.

As shown in Fig. 3, t_{ik} is the time point when Web service s_k was invoked by user u_i and similarly, t_{ik} is the time point when Web service s_k was invoked by user u_j; Δt_i is the time span between $t_{ik}$and the current time $t_{current}$; and similarly, Δt_j is the time span between $t_{jk}$and the current time. In the general case, Δt_i and Δt_j are different, so we use $\Delta t = (\Delta t_i + \Delta t_j)/2$ to represent the factor of temporal decay. Thus, we consider that when computing the interest similarity between u_i and u_j, the contribution of s_k would decay many fold with increasing Δt.

When considering the weighted rating effect, we define α, the coefficient of the weighted rating effect. The higher or lower the rating is, the more special attention should be given to computing the interest similarity between two users. Accordingly,

$$\alpha = \frac{1}{\sqrt{\left(r_{u_i,s_k} - \bar{r}_{u_i}\right)^2 + \left(r_{u_j,s_k} - \bar{r}_{u_j}\right)^2}} \tag{4}$$

where $r_{u,s}$ represents the overall QoS of Web service s invoked by user u; $\bar{r}_{ui}$ represents the mean overall QoS of all Web services invoked by u_i; and similarly, $\bar{r}_{uj}$ represents the mean overall QoS of all Web services invoked by u_j.

TABLE 1
RESULTS OF BAYES INFERENCE

Record	QoS	OS
<s_1, u_1, 1>	0.75	1
<s_3, u_2, 1>	0.75	1
<s_2, u_3, 1>	0.60	0
<s_2, u_1, 1>	0.80	1
<s_2, u_2, 1>	0.50	0
<s_3, u_3, 1>	0.55	0
<s_1, u_1, 2>	0.45	0
<s_1, u_3, 1>	0.85	1

Now, combining the traditional temporal decay model and the weighted rating effect, we can obtain the decay function including timespan Δt and the coefficient of weighted rating effect α:

$$f\left(\alpha, t\right) = e^{-\alpha\Delta t} = e^{-\frac{1}{\sqrt{\left(r_{u_i,s_k} - \bar{r}_{u_i}\right)^2 + \left(r_{u_j,s_k} - \bar{r}_{u_j}\right)^2}}\Delta t} \tag{5}$$

If we assume that:

$$x = -\frac{1}{\sqrt{\left(r_{u_i,s_k} - \bar{r}_{u_i}\right)^2 + \left(r_{u_j,s_k} - \bar{r}_{u_j}\right)^2}}\Delta t \tag{6}$$

we can obtain the simplified formula (7):

$$f\left(x\right) = e^x \tag{7}$$

where $x < 0$, and $f(x) \in (0,1)$. Formula (7) is described in Fig. 4. When X in the horizontal axis increases, $f(x)$ in the vertical axis will increase accordingly.

According to formulas (6) and (7), we know that when $\left|r_{ui,sk} - \bar{r}_{ui}\right|$ or $\left|r_{uj,sk} - \bar{r}_{uj}\right|$ increases, i.e., the rating is higher or lower, both x and $f(x)$ will increase accordingly. Thus, the effect will be more amplified. Meanwhile, when $\left|r_{ui,sk} - \bar{r}_{ui}\right|$ or $\left|r_{uj,sk} - \bar{r}_{uj}\right|$ decreases, i.e., the rating is closer to the average value, both x and $f(x)$ will decrease accordingly. Thus, the effect will be reduced.

Finally, we incorporate the temporal decay function (i.e., formula (5)) into the similarity measurement method PCC to describe the enhanced temporal decay model with weighted rating effect. Thus, the novel similarity computation method can be defined as:

$$\mathrm{sim}\left(u_i, u_j, \alpha, t\right) = \frac{\sum_{s_k \in W_{ui,uj}} \left(r_{ui,sk} - \bar{r}_{ui}\right)\left(r_{uj,sk} - \bar{r}_{uj}\right) f\left(\alpha, t\right)}{\sqrt{\sum_{s_k \in W_{ui,uj}} f\left(\alpha, t\right)\left(r_{ui,sk} - \bar{r}_{ui}\right)^2} \sqrt{\sum_{s_k \in W_{ui,uj}} f\left(\alpha, t\right)\left(r_{uj,sk} - \bar{r}_{uj}\right)^2}} \tag{8}$$

where $w_{ui,uj}$ denotes the set of Web services that the target user u_i and another user u_j commonly invoked, and s_k is an arbitrary Web service from $w_{ui,uj}$. If U denotes the whole user set, we can select the users using

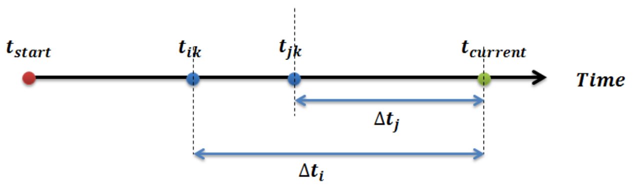

Fig. 3. Illustration of traditional temporal decay model

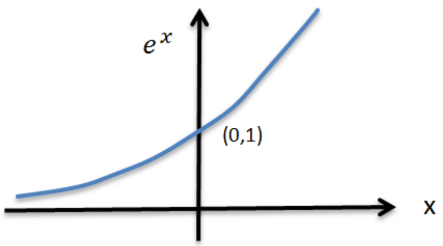

Fig. 4. Illustration of formula (7)

the temporal decay model with weighted rating effect from U to obtain the set $T(u_i)$ of users similar to the current user by formula (8).

4.4 QoS Prediction and Services Recommendation

In this section, we use the invocation records of Web services filtered from Sections 4.2 and 4.3 to make QoS predictions using Bayesian inference and thus make suitable service recommendations.

First, we employ Bayesian inference in QoS prediction. Bayesian inference considers both previous experiences and real-time contexts. The formula is:

$$P\left(OS=1|s_i\right)=\frac{P\left(s_i|OS=1\right)*P\left(OS=1\right)}{P\left(s_i\right)} \tag{9}$$

where $P(OS=1|s_i)$ describes the predicted QoS value of the Web service s_i for the current user; $P(OS=1)$ represents the probability of satisfactory QoS for a user in all the Web service invocation records, and $P(s_i|OS=1)$ is the probability that Web service s_i is satisfactory.

For instance, to explain the Bayesian inference formula (11), we set the threshold q to 0.7. We will also set different values of q to discuss the parameter's impact on the results in the experimental section. A Web service will satisfy the target user only if $QoS > 0.7$. For instance, "1" denotes "satisfied", and "0" denotes "not satisfied at all".

Table 1 shows an example of QoS prediction based on Bayesian inference. As shown in Table 1, each triple set $\langle s_i, u_i, n\rangle$ denotes that user u_i invokes Web service s_i with n-th invocation. Thus, $P(OS=1|s_i)$ is calculated using Bayesian formula as follows:

$$P\left(OS=1|s_1\right)=\frac{P\left(\left(s_1|OS=1\right)*P\left(OS=1\right)\right)}{P\left(s_1\right)}=\frac{1/2*1/2}{3/8}=\frac{2}{3} \tag{10}$$

$$P\left(OS=1|s_2\right)=\frac{P\left(\left(s_2|OS=1\right)\right)*P\left(OS=1\right)}{P\left(s_2\right)}=\frac{1/4*1/2}{3/8}=\frac{1}{3} \tag{11}$$

$$P\left(OS=1|s_3\right)=\frac{P\left(\left(s_3|OS=1\right)\right)*P\left(OS=1\right)}{P\left(s_3\right)}=\frac{1/4*1/2}{2/8}=\frac{1}{2} \tag{12}$$

As a result, s_1 is best candidate which will be recommended to the target user over s_2 and s_3.

The full CASR-TSE procedure is presented below.

Algorithm: Context-aware Web Services Recommendation for Modeling Weighted Temporal-Spatial Effectiveness (CASR-TSE)

Input: **q**: QoS threshold; **U**: set of test users; **Dataset:** processed training dataset
Output: **MAE/MSPE**: the error between the real values and predicted values

for every test user u_i in **U do**
  for q = 0.65:0.05:0.95
    $P_S = PE(S)$
    //PE represents preference elicitation according to the location L of the current user u_i to obtain the set P_S of Web services corresponding to current user preference//
    $PLT1(u_i)$ = Filtered (P_S, $u_i dataset$)
    //obtain the filtered dataset $PLT1(u_i)$ according to the preference set P_s//
    $T1(u_i) = TD\_WRE\_UPCC$ $(PLT1(u_i))$
    //TD_WRE_UPCC represents the UPCC considering temporal decay and the weighted rating effect and $T1(u_i)$ is the dataset got from TD_WRE_UPCC //
    PreQoS = Bayesian (q, $T1(u_i)$)
    // obtain the different QoS predictions for different q based on dataset $T1(u_i)$ using Bayesian inference//
    MAE = MAEfun (preQoS, real QoS)
    MSPE = MSPEfun (preQoS)
    // Compute both MAE and MSPE//
  end for
end for

4.5 Computational Complexity Analysis

In this section, we discuss the computational complexity of predicting one unknown QoS value using our CASR-TSE method.

For the convenience of computational complexity analysis, we first assume that the dataset is a $m \times n$ matrix including m Web service users and n Web services and that each entry in this matrix is an overall QoS value for a user invoking a Web service.

In the first step of CASR-TSE, i.e., modeling the effect of spatial correlations on user preference expansion, we identified the Web services invoked by every user. Thus, the computational complexity is $O(m*n)$.

Second, in the step of an enhanced temporal decay model with weighted rating effect, the similarity calculations were performed. We know that there are at most n Web services commonly invoked by both user u_i and u_j, then the complexity of formula (8) should be $O(n)$. Therefore, the complexity of all similarity calculations for user u is $O(m*n)$, because there are m users in the dataset.

Furthermore, we employed the Bayesian inference for the prediction of QoS. After modeling the effect of spatial correlations on user preference expansion, $service\_topK$ Web services were selected. Then, after modeling the enhanced temporal decay model with weighted rating effect, $user\_topN$ neighbor users were selected. As a result, the computational complexity of predicting the QoS of a Web service is $O(service\_topK * user\_topN)$, according to formula (9).

In summary, the total computational complexity of the proposed CASR-TSE algorithm is $2*O(mn)+$

$O(service\_topK * user\_topN)$.

TABLE 2
DISTRIBUTION OF 150 USERS

Rank	Country or Region	Number of Users	Proportion of Users
1	United States	73	48.67%
2	European Union	59	39.33%
3	Japan	6	4.00%
4	Canada	5	3.33%
5	Brazil	4	2.67%
6	Korea	2	1.33%
7	Taiwan	1	0.67%
Total	World	150	100%

5 EXPERIMENTS

In this section, we describe the extensive experiments conducted to evaluate the performance of the proposed CASR-TSE method. Employing the WS-Dream dataset, six methods including CASR-TSE, were evaluated with two evaluation metrics, i.e. MAE and MSPE. We also conducted experiments considering only one factor, such as traditional temporal decay, temporal decay with weighted rating effect, location-aware user similarity mining and spatial correlations effect on user preference expansion, to demonstrate the contribution of each factor to QoS prediction. More specifically, we addressed the following questions:

- How much better is our proposed CASR-TSE method than several previous well-known methods that considered only the temporal or spatial context?
- Does the threshold (q) affect the prediction accuracy?
- What is the performance of our method and the other baselines under different ratios of training data and testing data?
- When changing the number N of neighbors in our method and several other methods, what are the results? Does the proposed method still perform well for different N?
- Does the enhanced temporal decay model with weighted rating effect contribute to the improvement in the similarity computing between two users, compared with the traditional temporal decay method?

TABLE 3
DISTRIBUTION OF 100 WEB SERVICES

Rank	Country or Region	Number of Web Services	Proportion of Web Services
1	European Union	40	40%
2	United States	33	33%
3	Canada	10	10%
4	China	8	8%
6	Korea	3	3%
7	Australia	2	2%
7	Japan	2	2%
8	South Africa	1	1%
9	Thailand	1	1%
Total	World	100	100%

5.1 Datasets and Data Pre-processing

We adopted the WS-Dream[7] [41] for the experiment. This dataset contains 1,542,884 invocation records of Web services, by 150 Web service users on 100 Web services. Specifically, 150 users are from the seven countries or regions in Table 2, and the 100 Web services are from the nine countries or regions in Table 3. Every Web service was invoked approximately 100 times by every user. Furthermore, an invocation record contains six QoS properties: WSID (i.e., ID of a Web service), IP address of the user, data size, RTT (i.e., round-trip time), message of Response HTTP, and code of Response HTTP.

Due to the different types of QoS properties, the values of a Web service invocation record should be normalized before overall evaluation. Generally speaking, the values of these QoS properties should be of the ratio or numeric type. The value of the ratio type is limited to a range of [0, 1], while the value of the numeric type can have any range. To make data in different ranges contribute proportionately to QoS prediction, the values of different numeric types should be normalized.

In addition, the QoS records collected from each user should be normalized separately according to the standard collaborative filtering format. Common normalization methods include zero-mean normalization, decimal scaling, min-max normalization, Gaussian approach and so on. Here, we adopted the Gaussian method (as shown in formula (13)) to normalize the QoS property data because of the well-balanced distribution. According to the Gaussian method, the QoS properties RTT and Data Size in the WS-Dream dataset can be normalized as follows:

$$v_l^{k,j} = 0.5 + \left(q_l^{k,j} - \overline{q_l^{j}}\right) / \left(2*3\sigma_j\right) \tag{13}$$

where $v_l^{k,j}$ is the normalized result of the original $q_l^{k,j}$; $\overline{\overline{q_l^{j}}}$ is defined as the mean value of user u_i's QoS data on the l-th property; σ_i is the standard deviation of user u_i's QoS on the l-th property; and $3\sigma_i$ is employed according to the 3-σ_i rule. The normalized results show that the probability of the normalized values falling in the range of $[0, 1]$ is approximately 99%.

However, the Gaussian method is not suitable for the normalization of the Response HTTP Message because the Response HTTP Message is not a numeric but a string. The Response HTTP Message reflects the success of invoking a Web service. If the message is "OK", it represents success; otherwise, it represents failure. There are 25 types of failure Response HTTP Message, including "OK", "java. net. Socket Time out Exception: connect timed out" and others. If the message of Response HTTP is "OK", the normalized value is defined as 1. Otherwise, it is defined as 0. Furthermore, because the Response HTTP Code and Message are closely related ("OK" in the Response HTTP

[7] WS-Dream dataset, http://www.wsdream.net/dataset.html

This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TSC.2017.2782793

Message corresponds to the code "200" in the Response HTTP Code, and the other Response HTTP Message values corresponds to different codes in the Response HTTP code, e.g., the message "Internal Server Error" corresponds to the code "500"), we omit the property Response HTTP Code in the evaluation of the overall QoS of a Web service.

Then, we will have a computation of the overall QoS. When computing the overall QoS using multiple QoS properties, some QoS properties, such as response time, are inversely proportional to the user's QoS. That is to say, the larger the response time is and the lower the QoS should be. As a result, when performing the overall QoS calculation, we should consider properties (such as response time) which have values that are inversely proportional to the user's satisfaction. Thus, inspired by the method proposed in [53], we give the calculation of the overall QoS with the following formula:

$$QoS = \sum_{proi \in Q_D} w_i v_{proi} + \sum_{proj \in Q_N} w_j \frac{1 - v_{proj}^2}{1 + v_{proj}^{1/2}} \quad (14)$$

where Q_D denotes the set of QoS properties whose values are directly proportional to the user's satisfaction, and Q_N denotes the set of QoS properties whose values are inversely proportional to the user's satisfaction. The formula $1 - v_{proj}^2 / 1 + v_{proj}^{1/2}$ is used to give a transformation between them; w_i and w_j are the weights of property I and property j, respectively; and v_{pro} denotes the normalized QoS values. To make data in different ranges influence QoS prediction proportionately, the values of the different properties' QoS should be normalized.

The threshold q is defined to describe whether a Web

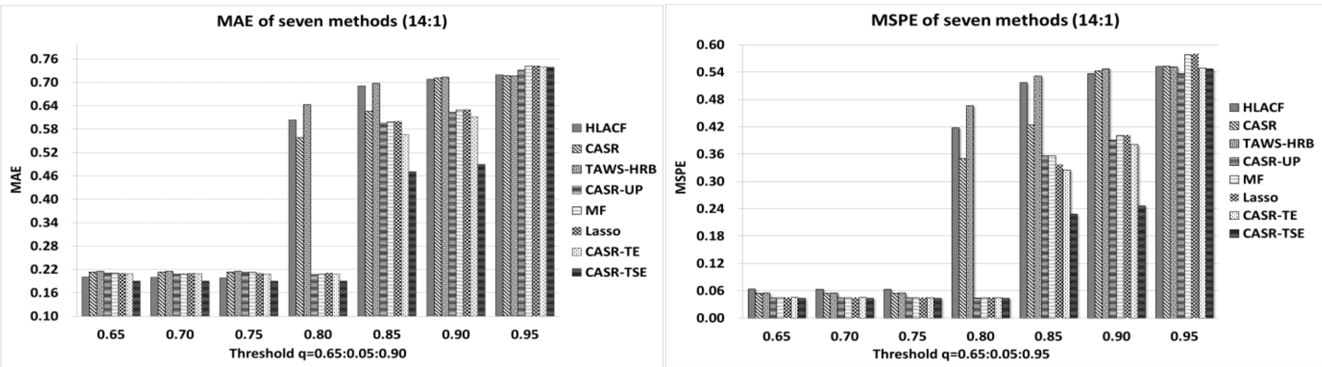

Fig. 5. MAE and MSPE results of eight methods (14:1)

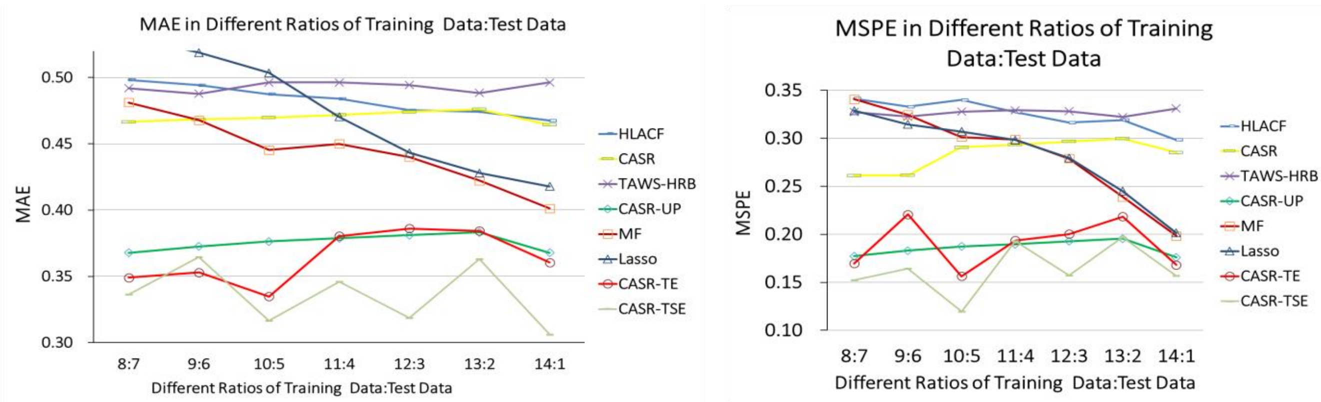

Fig. 6. MAE, RMSE and MSPE results of compared methods (at various ratios)

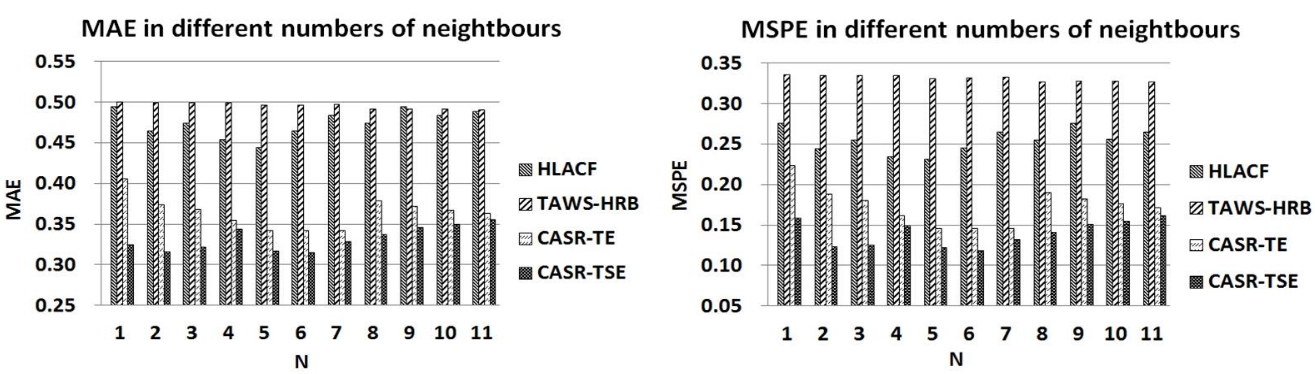

Fig. 7. MAE and MSPE results of HLACF, TAWS-HRB, CASR-TE and CASR-TSE (with different numbers of neighbors)

service is deemed satisfying. If QoS<q, the Web service is deemed unsatisfying; if QoS>q, the Web service is deemed satisfying. In Section 5.4, we show the detailed experimental results for different values of q.

The simulation and experiment were developed using MATLAB 2013 and conducted on an ASUS K55V PC with Windows 7 operating system, 32 GB RAM and Intel Core I7 3.6 GHz CPU.

5.2 Evaluation Metrics

The evaluation metrics used in our experiments are Mean Absolute Error (MAE) and mean squared prediction error (MSPE). The formulas for MAE and MSPE are as follows:

$$MAE = \frac{\sum_{u,s}\left|Q_{u,s} - \hat{Q}_{u,s}\right|}{N} \quad (15)$$

$$\text{MSPE} = \frac{1}{N}\sum_{u,s}\left(Q_{u,s} - \hat{Q}_{u,s}\right)^2 \quad (16)$$

In formulas (15) and (16), $Q_{u,s}$ represents real QoS values of a Web service invoked by a user. $\hat{Q}_{u,s}$ denotes predicted QoS values of a Web service invoked by user u; and N is the total number of predicted QoS values.

5.3 Comparative Methods

We conducted a series of experiments to compare our proposed CASR-TSE method with several existing methods:

- **CASR** [8]: this method of recommending Web services to a user is based on Web service invocation experiences under similar spatial contexts to the current user.
- **HLACF** [46]: this method leverages both locations of users and Web services when selecting similar neighbors for the target user or service to have a Web services recommendation.
- **CASR-UP** [34]: this method of recommending Web services to a user considers the user’s preference, as determined by user’s location, to make recommendations.
- **TAWS-HRB** [49]: this method makes recommendations for users by considering the time decay effects in UPCC.
- **MF** [55]**:** matrix factorization is a latent factor based approach whose entries in matrix are user-given ratings on different items.
- **Lasso** [56]: is spatial temporal QoS prediction approach to time-aware Web service recommendation.
- **CASR-TE** [16]: this method makes recommendations for users with the consideration of the temporal effectiveness.

5.4 Results and Analysis

5.4.1 Impact of *q*

In this section, for different methods, we first describe the experimental results of MAE and MSPE generated by different threshold q values. As shown in Fig. 5, the horizontal axis is the threshold q (from 0.65 to 0.95), and the vertical axis is the MAE and MSPE results, respectively.

In this experiment, we divided the dataset into 15 segments and all the MAE and MSPE results are obtained with a 14:1 ratio of the training dataset and test dataset. To avoid the contingency of the experimental results, we performed standard cross-validation which uses every segment (total 15) as testing data. The MAE and MSPE results shown in Fig. 5 were generated with the average results of the cross-validation process.

In general, as shown in Fig. 5, the MAE and MSPE results of the proposed method are better than 7 baseline methods, demonstrating the significance of the CASR-TSE method in recommending personalized Web services. Furthermore, we also calculated the average accuracy improvements of the proposed method compared with baselines with different q (0.65:0.05:0.95). As shown in Table 5, the proposed CASR-TSE method outperforms 7 baselines with a significant margin. In addition, we will provide a detailed analysis on abnormal results in the next paragraph.

From Fig. 5, we can infer the impact of q from the results: **(1)** When the threshold $q \le 0.90$, both MAE and MSPE results of our method are smaller than most baseline methods; **(2)** Why is $q = 0.95$ abnormal? By considering the effect of spatial correlations on user preference expansion and the weighted rating effect, our method could have filtered out a large number of Web services, so that the invocation records of the selected Web services are more useful and positive for QoS prediction. However, as the threshold q increases continuously, most of positive Web services may be excluded. Thus, the result becomes abnormal; **(3)** We could also observe that when $q \le 0.70$, the results of all the methods remain almost invariable. Why? We think it is reasonable that when q decreases sufficiently, almost all Web services are included. Thus, results of MAE and MSPE will be invariable; and **(4)** the threshold q for the calculated probability is highly relevant to the results. We believe that for the proposed CASR-TSE method, the best q is approximately 0.75.

In summary, for most q values, after considering the effect of spatial correlations on user preference expansion and the weighted rating effect, we could get more useful and positive invocation records for the target user, which is very important to make a more accurate prediction in QoS.

5.4.2 Impact of different ratios

In this section, we show the experimental results of different ratios of the training data and testing data. The results of the six methods with different ratios (8:7, 9:6, 10:5, 11:4, 12:3, 13:2, and 14:1) of the training dataset and testing dataset. As shown in Fig. 6, the horizontal axis shows the different ratios, and the vertical axis is MAE or MSPE. The MAE and MSPE values are not generated from a specific q but the average values from different q (0.65:0.05:0.95). Furthermore, the number of the neighborhoods in HLACF, CASR, TAWS-HRB, CASR-UP, MF, Lasso, CASR-TE and CASR-TSE is set to 5. In

This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TSC.2017.2782793

TABLE 4 COMPARED RESULTS

	q = 0.90		q = 0.85		q = 0.80		q = 0.75		q = 0.70	
	MAE	**MSPE**	**MAE**	**MSPE**	**MAE**	**MSPE**	**MAE**	**MSPE**	**MAE**	**MSPE**
A	0.7107	0.5432	0.6264	0.4236	0.5580	0.3503	0.2133	0.0545	0.2133	0.0545
B	0.6218	0.3886	0.5924	0.3532	0.2074	0.0446	0.2074	0.0446	0.2078	0.0448
C	0.7125	0.5463	0.6935	0.5230	0.6342	0.4571	0.2142	0.0548	0.2142	0.0548
D	0.6096	0.4429	0.5904	0.4203	0.5138	0.3285	0.2061	0.0501	0.2127	0.0541
E	0.4896	0.2464	0.4707	0.2284	0.1901	0.0467	0.1901	0.0467	0.1902	0.0467

A: *Traditional Context Similarity Effect.* ***B:*** *Modeling Spatial Correlations on user preference expansion.* ***C:*** *Traditional Temporal Decay.* ***D:*** *Enhanced Temporal Decay combing Weighted Rating Effect.* ***E:*** *Weighted Temporal-Spatial Effectiveness.*

addition, the final results are retrieved from the average experimental results conducted in three times.

Fig. 6 shows that, in general, the MAE RMSE results of the proposed CASR-TSE method are better than baseline methods in different ratios. In addition, we also calculated the average accuracy improvements of the proposed method compared with baselines with different ratios (8:7, 9:6, 10:5, 11:4, 12:3, 13:2, 14:1) As shown in Table 5, the proposed CASR-TSE method outperforms 7 baselines with a significant margin. However, there are also several abnormal results, and we will provide a detailed analysis in the next paragraph.

From Fig. 6, we draw several conclusions: **(1)** The MAE and MSPE results of the six methods are expected to decrease as the ratio of training data to testing data increases, because more training data will help obtain more accurate evaluation. However, the experimental results show a certain degree of turbulence. As far as we are concerned, the reason for this is possibly due to different numbers of testing data used in different ratios. For different testing data, prediction errors may exist because of the contingency of the testing dataset itself. **(2)** Our CASR-TSE method outperforms the baselines when the ratio of the training data and testing data is high (i.e., 12:3, 13:2, 14:1). However, when the ratio is low (i.e., 8:7, 9:6, 10:5), the MAE and MSPE of our CASR-TSE method are close to the results of CASR-UP and CASR-TE. Because a large number of invocation records has been filtered out by leveraging the effect of both the spatial correlations and the weighted rating effect in our CASR-TSE method. As a result, the filtered dataset to be used for QoS prediction may be quite small.

Generally speaking, for different ratios of training data and testing data, the proposed CASR-TSE method still performs better than the other seven methods.

5.4.3 Impact of N

We are also interested in whether the number N of neighbor users has a significant impact on the QoS prediction accuracy. To evaluate the impact of N, we conducted a series of experiments comparing our CASR-TSE method with three other methods (i.e., HLACF, TAWS-HRB, CASR-TE) which also involve the neighbor number N.

In Fig. 7, we show the results of HLACF, TAWS-HRB, CASR-TE and CASR-TSE with different numbers of neighbors (i.e., 1:1:11). In this experiment, we divided the dataset into 15 segments and all the MAE and MSPE results are obtained with a 14:1 ratio of the training and test data. To avoid the contingency of the experimental results, we also used every segment (total 15) as testing data and performed the cross-validation. The MAE and MSPE results shown in Fig. 7 were generated by average results of all cross-validation values. In addition, the MAE and MSPE values are the average values with different values of q (0.65:0.05:0.95).

Based on the results in Fig. 7, we conclude that: **(1)** CASR-TSE outperforms the other three methods regardless of the number of neighbors. In addition, in Table 5, the average accuracy improvements of different N (1:1:11) showed the good performance of the CASR-TSE method compared with three baselines. **(2)** As the number of neighbors increases in a certain range (1~6), the MAE and MSPE results gradually decrease, except $N = 4$. It is expected that with the increase of N (1~6), more useful neighbor users will be included and the result should be better. Why $N = 4$ is abnormal? As far as we are concerned, prediction errors may exist because of the contingency of the testing dataset itself. **(3)** When the number of neighbors is large enough (6~11), the MAE and MSPE results become worse. Why? We believe that when the number of neighbor users rises, some neighbor users may be not useful for the prediction, which might result in poor prediction results. Furthermore, we did not show the results after N>11, because the results remain to be the same.

5.4.4 Impact of two effects: Spatial Correlations, and Weighted Rating Effect

To test our two claims proposed in the Section 1, we conducted a group of experiments considering the effectiveness of spatial correlations and weighted rating, respectively.

In these experiments, the ratio of the training and testing data is 14:1, and the number of neighbors N is 5. To avoid the contingency of the experimental results, we perform the cross-validation. The MAE and MSPE results shown in Table 4 were generated by averaging the results of all cross-validation results.

In Table 4, we show the results for different q values (0.70:0.05:0.90) when considering different factors. **A** considers the traditional context similarity effect, and **B** takes spatial correlations on user preference expansion into account. **C** is the traditional temporal decay model. **D** is the enhanced temporal decay model with weighted rating effect. Finally, **E** is the proposed CASR-TSE method, i.e., the combination of **B** and **D**.

From the results in Table 4, we can draw the following conclusions:

First, the results of considering the spatial correlations on user preference expansion (i.e., **B**) are better than the traditional user context similarity (i.e., **A**). Specifically, the average improvement of accuracy in MAE and MSPE are 14.42% and 27.81%, respectively.

TABLE 5 AVERAGE ACCURACY IMPROVEMENT OF CASR-TSE COMPARED WITH OTHER METHODS

Compared Methods	Average Accuracy Improvement of different q (0.65:0.05:0.95)		Average Accuracy Improvement of different ratios (8:7, 9:6, 10:5, 11:4, 12:3, 13:2, 14:1)		Average Accuracy Improvement of different N (1:1:11)	
	MAE	**MSPE**	**MAE**	**MSPE**	**MAE**	**MSPE**
HLACF	25.90%	45.94%	30.43%	49.88%	29.99%	45.19%
CASR	24.37%	40.51%	28.53%	42.66%	—	—
TAWS-HRB	27.92%	47.08%	31.85%	50.17%	32.98%	57.88%
CASR-UP	11.73%	18.06%	10.50%	12.42%	—	—
MF	12.41%	21.03%	24.33%	42.45%	—	—
Lasso	12.51%	20.28%	28.98%	42.25%	—	—
CASR-TE	10.56%	16.59%	7.70%	14.08%	8.88%	19.59%

"—" represents that there is no neighbor users in this method.

TABLE 6 P-VALUES AND PERCENTILES OF THE WEIGHTED RATING EFFECT

q	**0.70**	**0.71**	**0.72**	**0.73**	**0.74**	**0.75**	**0.76**	**0.77**	**0.78**
P-values	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
Percentiles 90	-0.0039	-0.0831	-0.0982	-0.0163	-0.01215	-0.0156	-0.02015	-0.0214	-0.0492
q	**0.79**	**0.80**	**0.81**	**0.82**	**0.83**	**0.84**	**0.85**	**0.86**	**0.87**
P-values	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
Percentiles 90	-0.3879	-0.4045	-0.4095	-0.4131	-0.0453	-0.0121	-0.0103	-0.1213	-0.1163
q	**0.88**	**0.89**	**0.90**	**0.91**	**0.92**	**0.93**	**0.94**	**0.95**	
P-values	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	
The 90th Percentile	-0.1129	-0.1195	-0.1180	-0.1173	-0.116	-0.1112	-0.0062	-0.0020	

Second, the results of the enhanced temporal decay with weighted rating effect (i.e., **D**) are better than traditional temporal decay (i.e., **C**). Specifically, the average improvement of introducing weighted rating effect is significant, with the increase rate of 10.56%, and 15.31% in MAE and MSPE, respectively. Furthermore, in order to testify whether the weighted rating effect is statistically significant, a two-step validation is conducted and results are shown in Table 6. For one thing, we reported p-values of the weighted rating effect with Mann-Whittney U method [57]. Specifically, we first assumed that the distribution of the prediction error of D is the same across categories of the prediction error of C. Then, all p-values with different threshold q are 0.00 as shown in Table 6, indicating that the previous assumption is rejected and variations between the prediction error of D and C are huge. For another thing, we computed the 90th percentile of the variation between the prediction error of D and C. In Table 6, when $q >= 0.70$, the 90th percentiles are all negative, demonstrating that D outperforms C significantly by statistics. In conclusion, we could validate that the weighted rating effect could improve the accuracy of QoS prediction significantly in both values and statistics.

Third, the results for **E** (i.e. the proposed CASR-TSE method) are better than for **A, B, C** or **D**, indicating that the proposed CASR-TSE method (i.e., a combination of **B** and **D)** could achieve the best results, compared with all baselines.

6 Conclusions

The paper proposes the CASR-TSE method to model the effectiveness of temporal-spatial correlations on user preference expansion for personalized Web service recommendation. In order to improve the QoS prediction accuracy, we first model the effectiveness of spatial correlations to apply the personalized service filter before the similarity computation. Second, we incorporate the weighted rating effect into the traditional temporal decay model for similarity computation. Finally, we conduct comprehensive experiments with various settings of parameters on a real-world dataset. Experimental results show that the proposed method can significantly improve the accuracy of QoS prediction and outperforms several existing well-known methods.

In the future, we will incorporate other contextual information, such as social contexts, to improve the accuracy of QoS prediction. In addition, we are also planning to combine temporal-spatial effectiveness with other similarity measurement approaches for better personalized Web service recommendation.

Acknowledgments

The corresponding author of this paper is Zibin Zheng with School of Data and Computer Science, Sun Yat-sen University, China. The work was supported by grants from the National Natural Science Foundation of China (61300232); the Gansu Provincial Science and Technology Support Program (1504WKCA087); the China Postdoc Foundation (2015M580564); and Fundamental Research Funds for the Central Universities (lzujbky-2015-100, lzujbky-2016-br04).

References

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

[2] M. P. Papazoglou, "Service-oriented computing: concepts, characteristics and directions," Proc. Fourth International Conference on Web Information Systems Engineering (WISE 03), IEEE Press, December 2003, pp. 3–12, doi: 10.1109/WISE.2003.1254461

[3] Z. Zheng, H. Ma, M.R. Lyu, and I. King, "QoS-aware Web Service Recommendation by Collaborative Filtering," IEEE Transactions on Services Computing, vol. 4, no. 2, 2011, pp. 140-152, doi: 10.1109/TSC.2010.52.

[4] Z. Zheng, and M.R. Lyu, "Collaborative Reliability Prediction for Service-Oriented Systems," Proc. ACM/IEEE 32nd International Conference on Software Engineering Web (ICSE 10), vol. 1, 2010, pp. 35-44, doi: 10.1145/1806799.1806809.

[5] G. Adomavicius and A. Tuzhilin, Context-aware recommender systems: Recommender systems handbook, Springer, US, 2011.

[6] P. Dourish, "What we talk about when we talk about context," Personal and Ubiquitous Computing, vol. 8, no. 1, 2003, pp. 19–30, doi: 10.1007/s00779-003-0253-8.

[7] D. Jannach, T.U. Dortmund, and G. Friedrich, "Tutorial: Recommender Systems," Proc. International Joint Conference on Artificial Intelligence Beijing (IJCAI 13), August 2013.

[8] L. Kuang, Y. Xia, and Y. Mao, "Personalized Services Recommendation Based on Context-Aware QoS Prediction," Proc. IEEE International Conference on Web Services (ICWS 12), IEEE Press, June 2012, pp. 400-406, doi: 10.1109/ICWS.2012.12.1.

[9] W. Zhang, H. Sun, X. Liu, and X. Guo, "Incorporating Invocation Time in Predicting Web Service QoS via Triadic Factorization," Proc. IEEE International Conference on Web Services (ICWS 14), IEEE Press, June 2014, pp.145-152, doi: 10.1109/ICWS.2014.32.

[10] Y. Hu, Q. Peng, and X. Hu, "A Time-Aware and Data Sparsity Tolerant Approach for Web Service Recommendation," Proc. IEEE International Conference on Web Services (ICWS 14), IEEE Press, June 2014, pp. 33-40, doi:10.1109/ICWS.2014.18.

[11] Y. Zhang, Y Fan, K. Huang, W. Tan, and J. Zhang, "Time-Aware Service Recommendation for Mashup Creation in an Evolving Service Ecosystem," Proc. IEEE International Conference on Web Services (ICWS 14), IEEE Press, June 2014, pp. 25-32, doi:10.1109/ICWS.2014.17.

[12] X. Zhang, K. He, J. Wang, C. Wang, G. Tian, and J. Liu, "Web Service Recommendation Based on Watch list via Temporal and Tag Preference Fusion," Proc. IEEE International Conference on Web Services (ICWS 14), IEEE Press, June 2014, pp. 281-288, doi: 10.1109/ICWS.2014.49.

[13] K. Lee, J. Park, and J. Baik, "Location-based Web Service QoS Prediction via Preference Propagation for Improving Cold Start Problem," Proc. IEEE International Conference on Web Services (ICWS 15), IEEE Press, June 2015, pp. 177-184, doi: 10.1109/ICWS.2015.33.

[14] P. He, J. Zhu, Z. Zheng, and J. Xu, "Location-based Hierarchical Matrix Factorization for Web Services Recommendation," Proc. IEEE International Conference on Web Services (ICWS 14), IEEE Press, June 2014, pp. 297-304, doi: 10.1109/ICWS.2014.51.

[15] F. Zhu, G. Liu, Y. Wang, A. Liu, Z. Li, P. Zhang, and L. Li, "A

Context-Aware Trust-Oriented Influencers Finding in Online Social Networks," Proc. IEEE International Conference on Web Services (ICWS 15), IEEE Press, June 2015, pp. 456-463, doi: 10.1109/ICWS.2015.67.

[16] X. Fan, Y. Hu, R. Zhang, W. Chen, and P. Brézillon, "Modeling Temporal Effectiveness for Context Aware Web Services Recommendation," Proc. IEEE International Conference on Web Services (ICWS 15), IEEE Press, June 2015, pp. 225-232, doi: 10.1109/ICWS.2015.39.

[17] http://en.wikipedia.org/wiki/Collaborative_filtering.

[18] G. Adomavicius, and A. Tuzhilin, "Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions," IEEE Trans. Knowledge and Data Engineering, 2005, pp.734 – 749.

[19] M. Deshpande and G. Karypis, "Item-Based Top-N Recommendation," ACM Trans. Information System, 2004, vol. 22, no. 1, pp. 143-177.

[20] G. Linden, B. Smith, and J. York, "Amazon.com Recommendations: Item-to-Item Collaborative Filtering," IEEE Internet Computing, 2003, vol. 7, no. 1, pp. 76-80.

[21] R. Jin, J. Y. Chai, and L. Si, "An automatic weighting scheme for collaborative filtering," in Proc. of the 27th annual international ACM SIGIR conference on Research and development in information retrieval, 2004. doi:10.1145/1008992.1009051.

[22] G. Xue, C. Lin, Q. Yang, W. Xi, H. Zeng, Y. Yu, and Z. Chen," Scalable Collaborative Filtering Using Cluster-Based Smoothing," in Proc. 28th Int'l ACM SIGIR Conf. Research and Development in Information Retrieval, 2005, pp. 114-121.

[23] J. McCarthy, "Notes on formalizing context," Proc. International Joint Conference on Artificial Intelligence (IJCAI 93), vol. 1, 1993, pp. 555-560.

[24] P. Brézillon, "Task-realization models in contextual graphs," Proc. Modeling and Using Context: 5th International and Interdisciplinary Conference (Context 05). Lecture Notes in Computer Science, vol. 3554/2005, 2005, pp. 55-68, doi: 10.1007/11508373_5.

[25] B. Deva, and P. Ruppel, "Location Analytics as a Service: Providing Insights for Heterogeneous Spatiotemporal Data," Proc. IEEE International Conference on Web Services (ICWS 15), IEEE Press, June 2015, pp. 353-360, doi: 10.1109/ICWS.2015.114.

[26] X. Zheng, Y. Wang, and M.A. Orgun, "Binet: Trust sub-network extraction using binary ant colony algorithm in contextual social networks," Proc. IEEE International Conference on Web Services (ICWS 15), IEEE Press, June 2015, pp. 321-328, doi: 10.1109/ICWS.2015.51.

[27] K. Fletcher, and X.F. Liu, "A Collaborative Filtering Method for Personalized Preference-based Service Recommendation," Proc. IEEE International Conference on Web Services (ICWS 15), IEEE Press, June 2015, pp. 400-407, doi: 10.1109/ICWS.2015.60.

[28] Q. Liu, E. Chen, H. Xiong, C. Ding, and J. Chen, "Enhancing Collaborative Filtering by User Interest Expansion via Personalized Ranking," IEEE Transactions on Systems, Man, and Cybernetics—Part B: Cybernetics, vol. 42(1), February 2012, pp. 218-233, doi: 10.1109/TSMCB.2011.2163711.

[29] V.P. Modekurthy, K.K Fletcher, X.F. Liu, and M.C. Leu, "Personal Preference and Trade-off based Additive Manufacturing Web Service Selection," Proc. IEEE International Conference on Web Services (ICWS 15), IEEE Press, June 2015, pp. 440-447, doi: 10.1109/ICWS.2015.65.

[30] F. Hang, and L. Zhao, "Supporting End-User Service Composition: A Systematic Review of Current Activities and Tools," Proc. IEEE International Conference on Web Services (ICWS 15), IEEE Press, June 2015, pp. 479-486, doi: 10.1109/ICWS.2015.70.

[31] G. Zhang, M. Tang, S. Cheng, G. Zhang, H. Song, and J. Cao, "P2P Traffic Optimization," Science China Information Science, 2012, 55:1475-1492.

[32] J. Breese, D. Heckerman, and C. Kadie, Empirical analysis of predictive algorithms for collaborative filtering, San Francisco, CA: Morgan Kaufmann Publishers Inc., 1998.

[33] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl. Grouplens, "An open architecture for collaborative filtering of netnews," Proc. ACM Conference on Computer Supported Cooperative Work (CSCW 94), 1994, pp. 175-186, doi: 10.1145/192844.192905.

[34] X. Fan, Y. Hu, and R. Zhang, "Context-aware Web Services Recommendation based on User Preference," Proc. IEEE Asia-Pacific Services Computing Conference (APSCC 2014), December, 2014, IEEE press, pp. 55-61, doi: 10.1109/APSCC.2014.20.

[35] X. Cui, G. Yin, Q. Han, and Y. Dong, "An Improved Time-effectiveness Reliability Prediction Approach of Web Service," Journal of Computational Information Systems, vol. 10, no. 4, 2014, pp. 1365–1374, doi: 10.12733/jcis9139.

[36] K. Lee, J. Park, and J. Baik, "Location-based Web Service QoS Prediction via Preference Propagation for Improving Cold Start Problem," Proc. IEEE International Conference on Web Services (ICWS 15), IEEE Press, June 2015, pp. 177-184, doi: 10.1109/ICWS.2015.33.

[37] M. Silic, G. Delac, and S. Srbljic, "Prediction of Atomic Web Services Reliability for QoS-Aware Recommendation," IEEE Transactions on Services Computing, vol. 8, no. 3, 2015, pp. 425-438, doi: 10.1109/TSC.2014.2346492.

[38] Z. Zheng, H. Ma, M.R. Lyu, and I. King, "Collaborative Web Service QoS Prediction via Neighborhood Integrated Matrix Factorization", IEEE Transactions on Services Computing, vol. 6, no. 3, 2013, pp. 289-299, doi: 10.1109/TSC.2011.59.

[39] M.S. Saleem, C. Ding, X. Liu, and C. Chi, "Personalized Decision-Strategy based Web Service Selection using a Learning-to-Rank Algorithm," IEEE Transactions on Services Computing, vol. 8, no. 5, 2015, pp. 727-739, doi: 10.1109/TSC.2014.2377724.

[40] J. Zhang, C. Chow, and Y. Li, "iGeoRec: A personalized and efficient geographical location recommendation framework," IEEE Transactions on Services Computing, vol. 8, no. 5, 2015, pp. 701-714, doi: 10.1109/TSC.2014.2328341.

[41] Z. Zheng, Y. Zhang, and M.R. Lyu, "Investigating QoS of Real-World Web Services", IEEE Transactions on Services Computing, vol. 7, no. 1, 2014, pp. 32-39, doi: 10.1109/TSC.2012.34.

[42] Y. Ma, S. Wang, F. Yang, and R. Chang, "Predicting QoS Values via Multi-Dimensional QoS Data for Web Service Recommendations," Proc. IEEE International Conference on Web Services (ICWS 15), IEEE Press, June 2015, pp. 249-256, doi: 10.1109/ICWS.2015.42.

[43] Y. Zhong, Y. Fan, K. Huang, W. Tan, and J. Zhang, "Time-Aware Service Recommendation for Mashup Creation," IEEE Transactions on Services Computing, vol. 8, no. 3, 2015, pp. 356-367, doi: 10.1109/TSC.2014.2381496.

[44] Y. Hu, Q. Peng, X. Hu, and R. Yang, "Web Service Recommendation Based on Time Series Forecasting and Collaborative Filtering," Proc. IEEE International Conference on

Web Services (ICWS 15), IEEE Press, June 2015, pp. 233-240, doi: 10.1109/ICWS.2015.40.

[45] B. Bai, Y. Fan, K. Huang, W. Tan, B. Xia, and S. Chen, “Service Recommendation for Mashup Creation Based on Time-aware Collaborative Domain Regression,” Proc. IEEE International Conference on Web Services (ICWS 15), IEEE Press, June 2015, pp. 209-216, doi: 10.1109/ICWS.2015.37.

[46] J. Liu, M. Tang, Z. Zheng, X. Liu, and S. Lyu, “Location-Aware and Personalized Collaborative Filtering for Web Service Recommendation”, IEEE Transactions on Services Computing, vol. PP, issue 99, 2015, pp. 1, doi: 10.1109/TSC.2015.2433251

[47] F. Hao, S. Li, G. Min, H. Kim, S. Yao, and L. Yang, “An Efficient Approach to Generating Location-Sensitive Recommendations in Ad-hoc Social Network Environments”, IEEE Transactions on Services Computing, vol. 8, no. 3, 2015, pp. 520-533, doi: 10.1109/TSC.2015.2401833.

[48] O. Khalid, M. Khan, S. Khan, A. Zomaya, “OmniSuggest: A Ubiquitous Cloud-Based Context-Aware Recommendation System for Mobile Social Networks”, IEEE Transactions on Services Computing, vol. 7, no. 3, 2014, pp. 401-414, doi: 10.1109/TSC.2013.53.

[49] Y. Hu, Q. Peng, Y. Hu, “Time Aware and Data Sparsity Tolerant Web Service Recommendation Based on Improved Collaborative Filtering,” IEEE Transactions on Services Computing, vol. 8, no. 5, 2015, pp. 782-794, doi: 10.1109/TSC.2014.2381611.

[50] D. Athanasopoulos, A.V. Zarras, G. Miskos, V. Issarny, and P. Vassiliadis, “Cohesion-Driven Decomposition of Service Interfaces without Access to Source Code,” IEEE Transactions on Services Computing, vol. 8, no. 4, 2015, pp. 550-562, doi: 10.1109/TSC.2014.2310195.

[51] L. Yao, Q. Sheng, A. Ngu, J. Yu, and A. Segev, “Unified Collaborative and Content-Based Web Service Recommendation,” IEEE Transactions on Services Computing, vol. 8, no. 3, 2015, pp. 453 - 466, doi: 10.1109/TSC.2014.2355842.

[52] C. Wu, W. Qiu, Z. Zheng, X. Wang, and X. Yang, “QoS Prediction of Web Services based on Two-Phase K-Means Clustering,” Proc. IEEE International Conference on Web Services (ICWS 15), IEEE Press, June 2015, pp. 161-168, doi: 10.1109/ICWS.2015.31.

[53] Y. Liu, A.H. Ngu, L.Z. Zeng, “QoS computation and policing in dynamic web service selection,” Proc. of the 13th international World Wide Web conference (WWW 04), ACM Press, May 2004, pp. 66-73, doi: 10.1145/1013367.1013379.

[54] M. Tang, Y. Jiang, J. Liu, and X. Liu, “Location-aware collaborative filtering for QoS-based service recommendation,” Proc. IEEE International Conference on Web Services (ICWS 12), IEEE Press, June 2012, pp. 202-209, doi: 10.1109/ICWS.2012.61.

[55] W. He, J. Zhu, Z. Zheng, J. Xu, and M.R. Lyu, ”Location-Based Hierarchical Matrix Factorization for Web Service Recommendation,” Proc. IEEE International Conference on Web Services (ICWS 14), IEEE Press, June 2014, pp. 297-304, doi: 10.1109/ICWS.2014.51.

[56] X. Wang, J. Zhu, Z. Zheng, W. Song, Y. Shen, and M.R. Lyu, “A Spatial-Temporal QoS Prediction Approach for Time-aware Web Service Recommendation,” ACM Transactions on the Web, vol. 10, no. 1, 2016, pp. 7:1-25, doi: http://dx.doi.org/10.1145/2801164.

[57] R.E. Walpole, R. Myers, S.L. Myers, E.Y. Keying, Essentials of Probability & Statistics for Engineers & Scientists. Pearson Higher Ed, 2013.

Xiaoliang Fan is a Postdoc fellow at Fujian Key Laboratory of Sensing and Computing for Smart City, Xiamen University. He is also an assistant professor at School of Information Science and Engineering, Lanzhou University. He received his PhD degree at University Pierre and Marie CURIE, France in 2012. He served as PC member of IEEE Big Data, SCC, etc. His research interests include services computing, and big data analytics. He has published 30+ journal and conference papers in these areas. He received Eiffel Excellent PhD Fellowship and CSC-IBM Chinese Excellent Teacher Fellowship. He is member of ACM, IEEE, IEEE CS, and senior member of CCF.

Yakun Hu is a master student at School of Information Science & Engineering, Lanzhou University, China. He is majoring in Computer Architecture. He received the BE degree from Lanzhou University in 2014. His research interests include services computing, context-aware recommender system and data mining, etc. He has published 4 conference papers in these areas. He is also a student member of CCF.

Zibin Zheng is an associate professor at School of Data and Computer Science, Sun Yat-sen University. He received Outstanding Ph.D. Thesis Award of the Chinese University of Hong Kong at 2012, ACM SIGSOFT Distinguished Paper Award at ICSE2010, Best Student Paper Award at ICWS2010, and IBM Ph.D. Fellowship Award at 2010. He served as PC member of IEEE CLOUD, ICWS, SCC, ICSOC, SOSE, etc. His research interests include service computing and cloud computing.

Yujie Wang is a master student at School of Information Science & Engineering, Lanzhou University, China. He is majoring in Computer Architecture. He received the BE degree from JingGangshan University in 2015. His research interests include services computing, context-aware recommender system and data mining, etc. He has published 2 conference papers in these areas. He is also a student member of CCF.

Patrick Brézillon is a professor at Laboratoire d'Informatique de Paris 6, University Pierre and Marie Cuire, France. He defended his Thèse d'Etat (6-year duration) in 1983 at University Paris 6. He published about 420 papers, alone or with a grand total of 130 co-authors in international conferences and international journals (e.g., IEEE Expert, AI Magazine, The Knowledge Engineering Review, etc.) His research interests include context management and AI.

Wenbo Chen is an associate professor at School of Information Science & Engineering, Lanzhou University. He received his Ph. D. degree at Lanzhou University in 2010. His research interests include Cloud Computing, High Performance Computing, and Services Computing. He has published 10+ refereed journal and conference papers in these areas. He is member of ACM and CCF.