IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received August 29, 2018, accepted October 5, 2018, date of publication October 22, 2018, date of current version November 14, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2877137

Location-Aware Service Recommendation With Enhanced Probabilistic Matrix Factorization

YUYU YIN[1,2], (Member, IEEE), LU CHEN[1,2], YUESHEN XU[3], AND JIAN WAN[2,4]

[1]School of Computer, Hangzhou Dianzi University, Hangzhou 310018, China
[2]Key Laboratory of Complex Systems Modeling and Simulation, Ministry of Education, Hangzhou 310018, China
[3]School of Computer Science and Technology, Xidian University, Xi'an 710126, China
[4]School of Information and Electronic Engineering, Zhejiang University of Science and Technology, Hangzhou 310023, China

Corresponding author: Yueshen Xu (ysxu@xidian.edu.cn)

This work was supported in part by the National Key Research and Development Program of China under Grant 2017YFB1400601, in part by the National Natural Science Foundation of China under Grant 61702391, in part by the Natural Science Foundation of Zhejiang Province under Grant LY12F02003, in part by Shaanxi Province under Grant 2018JQ6050, and in part by the Fundamental Research Funds for Central Universities under Grant JBX171007.

ABSTRACT Owing to the ever-growing popularity of mobile computing, a large number of services have been developed for a variety of users. Considering this, recommending useful services to users is an urgent problem that needs to be addressed. Collaborative filtering (CF) approaches have been successfully adopted for services recommendation. Nevertheless, the prediction accuracy of the existing CF approaches is likely to reduce due to many reasons, such as inability to use side information and high data sparsity, which further lead to low quality of services recommendation. In order to solve these problems, some model-based CF approaches have been proposed. In this paper, we propose a novel quality of service prediction approach based on probabilistic matrix factorization (PMF), which has the capability of incorporating network location (an important factor in mobile computing) and implicit associations among users and services. First, we propose a novel clustering method that is capable of utilizing network location to cluster users. Based on the clustering results, we further propose an enhanced PMF model. The proposed model also incorporates the implicit associations among users and services. In addition, our model incorporates the implicit relationships between the users and the services. We conducted experiments on one real-world data set, and the experimental results show that our model outperforms the compared methods.

INDEX TERMS Implicit association, network location, probabilistic matrix factorization, QoS prediction, services recommendation.

I. INTRODUCTION

Service-oriented architecture (SOA) is widely-used in distributed computing environment [1], such as cloud computing and mobile computing. As the core element of SOA, services are also largely adopted as a popular way to provide configurable functions, especially in mobile computing [2]. So the number of services is increasing dramatically. It becomes an inevitable problem to select suitable services for a user [3]. Meanwhile, since the number of candidate services in mobile computing environment is large, it is hard for a user to finish the services selection task, and thus it is an urgent task to develop an effective services recommendation system. In services recommendation, quality of service (QoS for short) is a quite important factor, and the recommendation task can focus on uncovering those services that can provide best QoS. Because the number of services that a user can invoke or use is usually quite limited, the number of the known QoS values that a user can have is also limited. So the prediction of unknown or missing QoS values becomes the key task.

In recent years, collaborative filtering (CF) approaches have been successfully adopted in traditional recommender systems [4]. CF approaches utilize the invocation records of a user to identify similar users and further use such similarity to predict QoS for the target user. However, in QoS prediction for services, CF approaches usually suffer from low accuracy, especially when the QoS records are sparse. Traditional CF approaches are classified into two categories: neighborhood-based CF (also known as memory-based CF) and model-based CF [5]. Some previous studies proposed model-based CF approaches to predict QoS, such as the methods extended from matrix factorization (MF). Their results show that the MF-based methods can effectively improve the

QoS prediction accuracy under the case of high sparsity [6]. However, the traditional MF methods still have limitations, because different from the traditional rating prediction in recommender systems, QoS values are largely affected by physical and geographical factors [7], such as network location and geographical location, especially in mobile computing case. The traditional MF methods only utilize users' invocation records as explicit information to predict missing values, lacking of the capability of utilizing the context information, such as network or geographical locations, which are important factors in mobile computing.

Among the existing MF approaches, the probabilistic MF model (PMF), performs well on the large and sparse Netflix dataset [8]. Although PMF is a popular method in recommender systems, it still suffers from drawbacks in QoS prediction, which are two-fold. First, PMF represents users and services with latent features, but ignores the effect biases of users and services. Second, PMF assumes that the latent features of users and services are independent, ignoring the implicit associations among users and services.

In this study, we propose a novel QoS prediction approach based on PMF, which aims to incorporate the network location and implicit associations among users and services. Our work is motivated by the following observation. In the process of service invocation, the performance of the underlying network has a considerable effect on QoS (for example, response time and throughput) [9]. Users in the same country or same autonomous system share similar network configuration, such as bandwidth and routing protocols, so the QoS values received by these users are likely to be similar. Thus, our algorithm first clusters users into several regions based on network location and QoS records. The extensive experiments were conducted on one real-world QoS dataset. The contributions of this paper are summarized as follows.

- It proposes a new clustering algorithm based on K-prototype clustering algorithm. The proposed method is capable of leveraging a user's features to find users with similar network environment.
- It constructs an extended prediction model using the clustering results, and further propose an ensemble model which combines the baseline model and PMF model.
- It proposes a new prediction model extended from matrix factorization, which incorporates the services' network location information and implicit associations among users and services.
- It proposes an ensemble model, which combines the above two models. The experiments conducted on the real-world datasets show that our approaches can generate superior prediction results.

The remainder of this paper is organized as follows. Section II discusses the related work. Section III presents our framework. Section IV elaborates on the proposed prediction models. The experimental results and their analyses are presented in Section V. Finally, Section VI concludes the paper and discusses our future work.

II. RELATED WORK

Service recommendation systems aim at providing high quality services to target users. In recent studies, CF has been widely used in recommendation systems [4]. However, these CF recommendation algorithms often face many challenges, such as the cold start problem, data sparseness, and algorithm scalability [10], [11]. Considering the large number of services and the enormous cost for service users to invoke all services, it is infeasible to immediately acquire the QoS value to select the optimal service. To address this problem, many personalized QoS prediction approaches have been proposed in recent years [12]. Yu *et al.* [13] proposed a personalized QoS prediction approach for web services using latent factor models; in their study, they explain how latent factor models can be utilized to predict the unknown QoS values. Zhang *et al.* [14] built feature models and employed these models to make personalized QoS prediction for different users. Li and Ou [15] developed a new model named pairwise PMF, which employs two techniques, including learning to rank and PMF, to learn the relative preference for items, which is advantageous for QoS prediction. Xie *et al.* [16] proposed an asymmetric correlation regularized MF to alleviate the data sparseness problem.

The CF approaches are divided into two categories: neighborhood-based and model-based approaches [8]. Some of the neighborhood-based approaches achieve good prediction results. Shao *et al.* [17] proposed a user-based CF approach to predict the QoS values based on the similarity between service users. Zhao *et al.* [18] explored an improved item-based movie recommendation algorithm, which increases cinematic genres' effect on computing items' similarity. Further, Jiang *et al.* [19] developed an effective personalized hybrid collaborative filtering (PHCF) technique by integrating personalized user-based algorithm and personalized item-based algorithm for web services recommendations. However, the neighborhood-based recommendation approaches are usually unable to flexibly integrate many useful information, such as users' interests [20]. We need to utilize the explicit or implicit information to obtain a more accurate prediction. Considering this, some researchers found that the model-based approach can effectively integrate this implicit information for QoS prediction [21]. Qi *et al.* [22] proposed a novel QoS prediction method based on the MF model, integrating both user network neighborhood information and service neighborhood information with the MF model to predict personalized QoS values. Wu *et al.* [23] proposed a general contex.t-sensitive MF approach (CSMF) for collaborative QoS prediction. By considering the complexity of service invocations, CSMF models the interactions of users-to-services and environment-to-environment simultaneously, and makes full use of implicit and explicit contextual factors in QoS data. Further, Wei *et al.* [24] proposed an extended MF (EMF) framework with relational regularization for predicting missing QoS values. To avoid the expensive and costly web services invocations, they first elaborated the MF model from a general perspective. Then, they

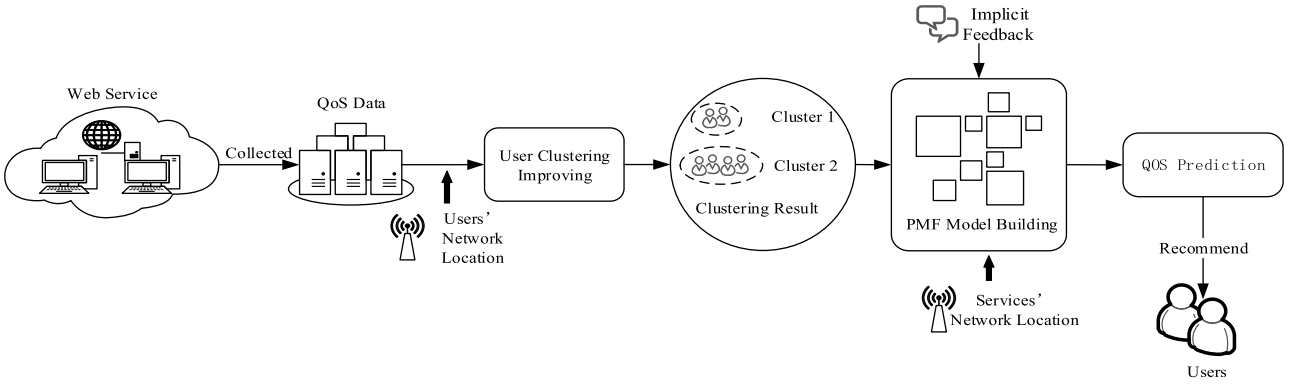


FIGURE 1. Service recommendation framework.

systematically designed two novel relational regularization terms inside a neighborhood. Finally, they combined both terms into a unified MF framework to predict the missing QoS values. Moreover, some researchers have found that integrating the context information into CF algorithms can lead to better prediction accuracy [9]. Li *et al.* [25] proposed an alternative and efficient approach to predict the missing QoS values, called the Location and Reputation aware MF-based Location Information (LRMF). The LRMF combined both the user's reputation and location information to achieve more accurate prediction results. He *et al.* [26] designed a location-based hierarchical MF method to perform personalized QoS prediction, based on which, effective service recommendation can be made.

Although the abovementioned studies made some improvements for QoS prediction models, the effect of implicit feedback information on the QoS prediction is not considered. Considering that the country information can affect the QoS values, in our study, we combined the implicit service location feedback information with the PFM model for QoS prediction. Consequently, by modelling the users' preference behavior in different countries, we make a personalized prediction.

III. THE PROPOSED SERVICE RECOMMENDATION FRAMEWORK

To recommend high quality services, we focus on achieving QoS prediction with high accuracy. We propose a service recommendation framework containing the proposed prediction models, which are illustrated in Figure 1. The three components of this framework are stated as follows.

Users clustering: This component uses a new initialization method to improve the accuracy of users clustering, by integrating users' network location information. This component aims at finding similar users in similar network environments, and then applies this clustering result to construct a novel PMF model.

PMF model building: In this component, a PMF model is proposed, exploiting the results of the mentioned clustering. Besides, we incorporate the implicit associations among users and services into t the proposed PMF model.

QoS Prediction: We use the proposed PMF model to predict the missing QoS values.

Using the users' and services' network locations, we first propose a baseline model based on this clustering result, and then build a PMF model based on implicit feedback information. To provide effective service recommendation, we combine these two models to predict the missing QoS values.

IV. PROPOSED PREDICTION MODELS

In this section, we present the two models used in our approach, i.e., the user clustering model and the enhanced PMF model.

A. IMPROVED USER CLUSTERING MODEL

We propose a clustering method based on the K-prototype algorithm. The K-prototype algorithm can be used to cluster data with mixed attributes, including numerical and categorical attributes. The results of the clustering using K-prototype algorithm are largely determined by the quality of the selected initial points. It indicates that this algorithm is sensitive to the choice of initial points. To solve this problem, different from the traditional initialization method in K-prototype algorithm, we propose a new initialization method fully take advantage of the characteristics of QoS records.

1) K-PROTOTYPE ALGORITHM

In this section, we introduce our clustering method based on the users' QoS records, network location and geographical location. We define a cluster as a group of users who are located close to each other and have similar QoS records.

QoS values are usually strongly related to the user's network environment, such as network bandwidth and network distance. In the same autonomous system, users are subject to the same routing protocol (usually it is the internal gateway protocol), using the same router group, so the routing capability of the users' devices are the same. Thus, the network transmission status within the same autonomous system is similar. It can be inferred that users in close network locations are likely to experience similar QoS. We cluster users based on the users' QoS records and network location information, and

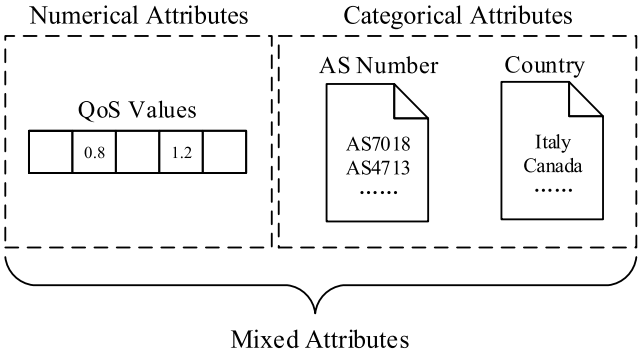


FIGURE 2. Mixed attributes.

those users within the same cluster are similar in receiving QoS. However, because QoS records are numeric attributes, whereas network location is categorical, we utilize the K-prototype algorithm to handle the clustering problem which is involved in mixed data. A novel PMF model is further proposed based on the clustering results.

The K-prototype algorithm is typically used for clustering of data with mixed attributes, and is based on the K-means and K-mode algorithms. The K-means algorithm is a simple, yet time-consuming clustering algorithm, especially on large-scale datasets. However, the K-means algorithm can be only used for data with numerical attributes, limiting its application scope. On the contrary, the K-mode algorithm can be used to cluster data with categorical attributes, but cannot be used for data with numeric attributes. The K-prototype algorithm combines the advantages of K-means and K-mode algorithms and can be used to cluster data with mixed attributes.

In service recommendation, the QoS records are numerical attributes, while the country affiliation and the number of autonomous systems (AS) are categorical attributes. The three attributes are then unified into a mixed attribute. The mixed data vectors composed of mixed attributes are clustering objects, as shown in Figure 2.

In this specific implementation, the sample refers to the users. First, we have $\overrightarrow{X} = \{X_1, X_2, \ldots, X_n\}$, which denotes the dataset with n users. Each X_i has a mixed attribute vector $(x^n_{i,1}, x^c_{i,2}, x^c_{i,3})$, which represents the three attribute values mentioned above for user X_i. In detail, $x^n_{i,1}$ is the QoS vector of user X_i, while $(x^c_{i,2}, x^c_{i,3})$ represents the country affiliation and the number of AS of the user.

Let k be the number of clusters, and $C = \{C_1, C_2, \ldots, C_k\}$ be the set of clusters. In K-prototype algorithm, the center of each cluster is named as a prototype. Let $Q = \{Q_1, Q_2, \ldots, Q_k\}$ denote the prototypes set for the k clusters. Each Q_l also has a mixed attribute vector $(q^n_{l,1}, q^c_{l,2}, q^c_{l,3})$, which represents the abovementioned three attribute values for the prototype Q_l. In clustering, the core task is to calculate the dissimilarity between each user and the prototype of each cluster. The dissimilarity of numerical attributes is computed by Euclidean distance between the numerical feature of a user and a cluster's prototype. This dissimilarity between user X_i and prototype Q_l is computed as follows.

$$d_n(X_i, Q_l) = \sum_{j=1}^{p} \left| x_{i,j} - q_{l,j} \right|^2 \tag{1}$$

where p represents the number of numerical features. The dissimilarity between two categorical features is calculated using Hemingway distance as

$$d_c(X_i, Q_l) = \sum_{j=p+1}^{m} \delta\left(x_{i,j}, q_{l,j}\right) \tag{2}$$

where m represents the number of categorical features. If $x_{i,j} = q_{l,j}$, then $\delta(x_{i,j}, q_{l,j}) = 0$, and if $x_{i,j} \neq q_{l,j}$, then $\delta(x_{i,j}, q_{l,j}) = 1$. Combining Eq.1 and Eq.2, the dissimilarity of the mixed features is computed as follows.

$$d(X_i, Q_l) = d_n + \gamma d_c = \sum_{j=1}^{p} \left(x^r_{ij} - q^r_{lj}\right)^2 + \gamma \sum_{j=p}^{m} \delta\left(x^c_{ij}, q^c_{lj}\right) \tag{3}$$

Finally, the cost function of clustering is

$$E = \sum_{i=1}^{n} \sum_{l=1}^{k} u_{il} d(X_i, Q_l)$$
$$s.t. \begin{cases} \sum_{i=1}^{k} u_{il} = 1 \\ u_{il} \in [0, 1] \end{cases} \tag{4}$$

In the case of $u_{il} = 1$, user X_i is in cluster C_l. In the case of $u_{il} = 0$, user X_i is not in the cluster C_l. The steps of the improved K-prototype algorithm are as follows.

Step 1: It sets the number of clusters and select a prototype for each cluster, and gets k prototypes.

Step 2: It computes the dissimilarity between each user and each prototype based on Eq. 3. Based on clustering results, it divides users into different clusters, wherein a user belongs to the cluster with the smallest dissimilarity value.

Step 3: Using Eq. 3, it recalculates the prototype of each cluster, and updates the prototypes correspondingly. Step 3 is repeated until users in each cluster remain unchanged.

In Step 3, after the prototype of each cluster is generated, the value of numerical feature of the prototype is set as the average value of numerical features of all users in the cluster. The value of categorical feature of the prototype is set as the categorical value of the user in the cluster with the highest similarity.

Due to high data sparsity, the number of known QoS records is small. To ensure that the dimension of each QoS vector is the same, the missing QoS value in each vector is filled with the mean of each user's QoS records. The pseudo-code of K-prototype algorithm is present in Algorithm 1. In Algorithm 1, $X[i]$ represents the mixed vector of user u_i, and $X[i,j]$ is the value of feature j in $X[i]$. QoSvalues[] and attributes[] store QoS values and features,

Algorithm 1: The Extended K-Prototype Algorithm

Input: QoS records

```
FOR i=1 to number of users DO
    Mindistance = 0
    FOR j=1 to number of clusters DO
        distance = d(X[i], prototype[j])
        IF distance < Mindistance DO
            Mindistance = distance, cluster = j
        END
    END
    IF C[i] ≠ cluster DO DO
        oldCluster = C[i]
        FOR j = 1 to numberofQoSvalues DO
            Sum[cluster, j] = Sum[cluster, j] + X[i,j]
            Sum[oldcluster, j] = Sum[oldcluster, j] –
            X[i,j]
            QoSvalues[cluster, j] = Sum[cluster,
            k]/ClusterCount[cluster]
            QoSvalues[oldcluster, j] = Sum[oldcluster,
            k]/ClusterCount[oldcluster]
        END
        FOR k = 1 to numberofattributes DO
            attributes[cluster, k] = HighestFeq(cluster,
            j)
            attributes[oldcluster, k] =
            HighestFreq(oldcluster, j)
        END
    END
END
```

respectively. QoSvalues[*i,j*] and attributes[*i,j*] are the numerical and categorical elements of the prototype for cluster *j*. *ClusterCount*[] denotes the number of users in clusters. *Sum*[] denotes the sum of QoS values in clusters. Function *HighestFreq*() is used to update the features of prototypes.

2) IMPROVED INITIALIZATION METHOD

The traditional K-prototype algorithm randomly selects several samples as initial points. However, as shown in Figure 3, the deficiency of random selection of initial points is likely to impair clustering results. We propose an improved initialization method that utilizes the user density and distance.

First, we use Eq. 3 to obtain the distance between two users, and then we define the density of a user as follows.

$$\rho(X_t) = \frac{|N(X_t)|}{\pi\theta^2}, \quad |N(X_t)| = \{X_i \in X, d(X_t, X_i) \le \theta\} \quad (5)$$

where X_t denotes the target user, $\rho(X_t)$ is the density of user X_t, and $|N(X_t)|$ represents the number of users satisfying the constraint of θ. A higher density $\rho(X_t)$ of a user X_t indicates, a larger number of users close to X_t and a higher probability of X_t being a cluster prototype. So we select the user with the maximum density as the first initial point. For selection of the rest of initial points, we compute the distance

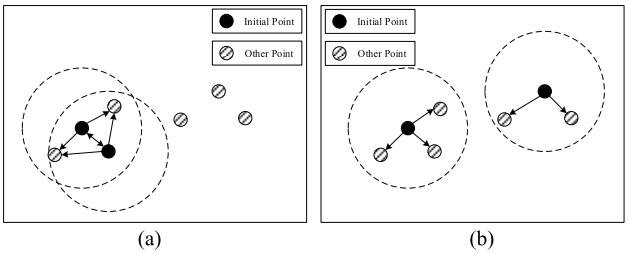


FIGURE 3. Two initialization methods for K-prototype algorithm. (a) Random initialization (b) Density-based initialization.

as follows. Let $Q = \{Q_1, Q_2, \ldots, Q_k\}$ denote the k initial points.

Step 1: $Q = \emptyset$. For each $X_i \in X$, this step computes $\rho(X_i)$.

Step 2: The second initial point Q_2 is the user who has the farthest distance from the first point Q_1, where Q_2 satisfies $Q_2 = X_i, d(X_i, Q_1) = \max\{d(X_1, Q_1), \ldots, d(X_n, Q_1)\}$.

Step 3: To find the rest of initial points, we compute the nearest distance from each initial point Q_l to each user X_i, where X_i can get $d_{\min}(X_i) = \min_{l=1}^{|Q|}\{d(X_i, Q_l)\}$, $X_i \in \{X - Q\}$, $Q_l \in \{Q\}$.

3) CLUSTERING RESULTS ANALYSIS

As used in the evaluation of common clustering methods, we use two metrics *Compactness* (CP) and *Separation* (SP) [27] to evaluate our proposed clustering method. CP is defined as

$$\overline{CP}_i = \frac{1}{|C_i|}\sum_{X_i \in C_i} \|X_i - Q_l\| \quad (6)$$

where C_i is the set of users X_i that have been grouped into a cluster, and Q is the set of Q_l prototypes of clusters in C_i. The average CP value of all clusters is calculated as follows.

$$\overline{CP} = \frac{1}{k}\sum_{i=1}^{k}\overline{CP}_i \quad (7)$$

where k denotes the number of clusters. As the members of each cluster should be close to each other, a lower value of CP indicates more compact clusters. SP measures the average distance between each prototype.

$$\overline{SP} = \frac{2}{k^2 - k}\sum_{i=1}^{k}\sum_{j=i+1}^{k}\left\|Q_i - Q_j\right\|^2 \quad (8)$$

where a higher value of $\overline{SP}$ indicates a better degree of separation. Figure 4 shows the CP and SP of two initialization methods on four matrices with densities being 5%, 10%, 15%, and 20%. It can be seen that our initialization method (Density-based) can generate more effective clustering results.

B. THE PROPOSED MODEL BASED ON PMF

1) TRADITIONAL PMF MODEL

We have a set of users $U = \{u_1, u_2, \ldots, u_m\}$, a set of services $S = \{s_1, s_2, \ldots, s_n\}$, and an $M \times N$ user-service

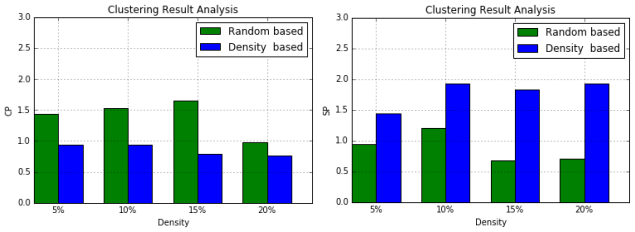


FIGURE 4. Effect of initialization method on clustering results.

QoS invocation matrix $R = [R_{ij}]_{M\times N}$, where M denotes the number of users, N denotes the number of services, and r_{ij} denotes the QoS value received by user i after invoking service j. If r_{ij} is unknown, it indicates that user u_i has never invoked service s_j. The conditional distribution over observed QoS values is

$$p\left(R|U,S,\sigma_R^2\right)=\prod_{i}^{N}\prod_{j}^{M}N\left(R_{ij}|U_i^TS_j,\sigma_R^2\right)^{I_{ij}} \tag{9}$$

where $N(x|\mu,\sigma^2)$ is the probability density function of Gaussian distribution, with mean u and variance σ^2, and I_{ij} is an indicator. The zero-mean spherical Gaussian priors are also followed by user and service feature vectors.

$$\begin{aligned}p\left(U|\sigma_U^2\right)&=\prod_{i=1}^{N}N\left(U_i|0,\sigma_U^2I\right)\\ p\left(S|\sigma_S^2\right)&=\prod_{j=1}^{N}N\left(S_j|0,\sigma_S^2I\right)\end{aligned} \tag{10}$$

Following Bayesian rule, we have

$$\begin{aligned}&p\left(U,S|R,\sigma_R^2,\sigma_U^2,\sigma_S^2\right)\\ &\propto p\left(R|U,S,\sigma_R^2\right)p\left(U|\sigma_U^2\right)p\left(S|\sigma_S^2\right)\\ &=\prod_{i=1}^{m}\prod_{j=1}^{n}N\left[\left(r_{ij}|U_i^TS_j,\sigma_R^2\right)\right]^{I_{ij}^R}\times\prod_{i=1}^{m}N\left(U_i|0,\sigma_U^2I\right)\\ &\quad\times\prod_{j=1}^{n}N\left(S_j|0,\sigma_S^2I\right)\end{aligned} \tag{11}$$

2) MOTIVATION FOR PMF IMPROVEMENT

We first present the motivation for improving the basic PMF model. The key improvements of our model are as follows.

- As discussed in Section III, users in the same cluster have a similar network environment. and such similarity is also applied to the service invocation. With the clustering results, our method extends the prediction way of basic PMF.
- In services recommendation, the basic PMF model only takes the‘ QoS values as input. Considering the different network configurations of users located in different regions, we incorporate implicit feedbacks to the basic PMF model.

In the remainder of this section, we present our two models, and the ensemble model.

3) THE EXTENDED PREDICTION MODEL

The QoS values are largely impacted by the network environment in which users and services are located. Although it is expected that users in the same cluster receive similar QoS values after invoking the same service, the network condition of some users is unstable at a certain time, which further impairs QoS. Thus, in average, the QoS of these users is worse than that of other users in the cluster. Likewise, there are some services that have a worse QoS because of unstable network condition. To incorporate the QoS deviation caused by the instability of network, we propose to incorporate a bias term devised based on clustering results, which is given below.

$$B\left(i,j,C\right)=\mu_C+BU_i+BS_j \tag{12}$$

where μ_C represents the average QoS value in cluster C, and BU_i and BS_j represent the bias terms of user i and service j, respectively BU_i and BS_j are computed with

$$BU_i=\frac{\sum_{j\in R(i)}\left(q_{ij}-\mu_C\right)}{\beta_1+|R\left(i\right)|},\quad BS_j=\frac{\sum_{i\in R(j)}\left(q_{ij}-\mu_C-BU_i\right)}{\beta_2+|R\left(j\right)|} \tag{13}$$

where $R(i)$ represents the set of services invoked by user i. $R(j)$ represents the set of users that invoke service j, and q_{ij} represents the QoS value received by user i after invoking service j.β_1 and β_2 represent the regularization coefficients. Based on BU_i and BS_j, we propose a new PMF model, in which the conditional distribution of user-service QoS matrix R is

$$\begin{aligned}&p\left(R|U,S,BU,BS,\sigma_R^2\right)\\ &\qquad=\prod_{i}^{N}\prod_{j}^{M}N\left(R_{ij}|B\left(i,j,C\right)+U_i^TS_j,\sigma_R^2\right)^{I_{ij}}\end{aligned} \tag{14}$$

The probability distributions of BU and BS are given as

$$\begin{aligned}p\left(BU|\sigma_i^2\right)&=\prod_{i=1}^{N}N\left(BU_i|0,\sigma_{BU}^2I\right)\\ p\left(BS|\sigma_j^2\right)&=\prod_{j=1}^{M}N\left(BS_j|0,\sigma_{BS}^2I\right)\end{aligned} \tag{15}$$

where BU and BS follow Gaussian distribution.

4) IMPLICIT FEEDBACK

We propose another QoS prediction model based on PMF model incorporating the implicit associations among users and services. The QoS records are commonly explicit feedbacks, such as response time. However, implicit feedback is not represented by numerical values, but by the user's invocation records to reflect a user's preference for

TABLE 1. The user-service matrix of explicit feedback (toy example).

	service1	service2	service3	service4	servcie5
user1	?	?	0.74	?	?
user2	0.66	?	?	?	2.28
user3	?	?	?	0.56	?
user4	1.13	?	?	?	1.06

TABLE 2. The user-service matrix of implicit feedback (toy example).

	service1	service2	service3	service4	servcie5
user1	?	?	1	?	?
user2	1	?	?	?	1
user3	?	?	?	1	?
user4	1	?	?	?	1

services. The explicit and implicit feedback in service invocation can be organized as two matrices that are shown in Table 1 and Table 2.

The elements in explicit feedback matrix E (shown in Table 1) are QoS records. The element $E_{i,j}$ represents the QoS value received by user i after invoking service j. In the implicit feedback matrix T(shown in Table 2), value 1 means that user i has invoked service j, and value 0 means that user i has never invoked service j before.

The traditional CF method takes QoS values as input, and usually relies on an assumption that most of QoS values are known or stable. However, because of the instability of network environment, users usually cannot invoke all services, resulting in noise data, which further decrease the prediction accuracy. In contrast, the implicit feedback can also reflect user invocation preferences, such as the frequency of a user invoking different services. Based on the observation, it can be inferred that a service's location is an important factor that influences user invocation choice. For example, if a user is to analyze the weather of a certain location, this user is highly likely to frequently invoke the weather forecast services of that location. Based on this observation, we propose an implicit feedback model.

We define the preference vector of user i as

$$Z_i = U_i + \sum_{c \in G} \left(\omega_{ic} \left| N_i(c) \right|^{-\frac{1}{2}} \sum_{s_k \in N_i(c)} Y_k \right) \tag{16}$$

where U_i represents the user latent feature vector. G represents the set of all countries in which services are located. $N_i(c)$ represents the set of services that are located in country c and meanwhile are invoked by user. Y_k represents the implicit feedback vector of service k. $|N_i(c)|^{-1/2}$ is used to normalize the sum of implicit feedback vectors. ω_c represents the user's choice for the service in a country, and ω_c is defined as $\omega_c = |N_i(c)| / |N(u_i)|$, where $|N_i(u_i)|$ represents the services sets invoked by user i. $|\cdot|$ represents the number of services.

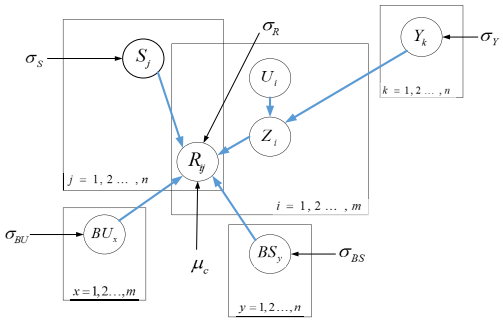


FIGURE 5. Proposed context-aware prediction model.

The conditional distribution of the user-service QoS matrix is

$$p\left(R|U,S,Y,\sigma_R^2\right) = \prod_i^m \prod_j^n N\left(R_{ij} \left| S_j^T \left(U_i + \sum_{c \in G} \left(\omega_{ic} \left| N_i(c) \right|^{-\frac{1}{2}} \times \sum_{s_k \in N_i(c)} Y_k \right) \right), \sigma_R^2 \right)^{I_{ij}} \tag{17}$$

The implicit feedback factor vectors Y follows mean spherical Gaussian prior.

C. THE ENSEMBLE QOS PREDICTION

In previous sections, we state two proposed prediction models. The first one is the prediction model based on clustering results, and the second one is the model based on implicit feedback. We name the clustering features and implicit feedback features as context features. We further propose a prediction model, named as context-aware PMF (CA-PMF), and the graphical model is given in Figure 5.

The conditional distribution of the user-service QoS matrix is

$$p\left(R|U,S,BU,BS,Y,\sigma_R^2\right) = \prod_i^m \prod_j^n N\left(R_{ij} \left| B(i,j,C) + S_j^T \left(U_i + \sum_{c \in G} \left(\omega_{ic} \left| N_i(c) \right|^{-\frac{1}{2}} \sum_{s_k \in N_i(c)} Y_k \right) \right), \sigma_R^2 \right)^{I_{ij}} \tag{18}$$

We can get the posterior probability of latent variables U, S, BU, BS, and Y, and further derive the logarithm of the posterior distribution as

$$E = In\left(p\left(U,S,BU,BS,Y \middle| R,\sigma_R^2,\sigma_U^2,\sigma_S^2,\sigma_{BU}^2,\sigma_{BS}^2,\sigma_Y^2\right)\right) = -\frac{1}{2\sigma_R^2} \sum_{i=1}^m \sum_{j=1}^n I_{ij} \left(R_{ij} - \left(B(i,j,C) + S_j^T \left(U_i + \sum_{c \in G} \times \left(\omega_{ic} \left| N_i(c) \right|^{-\frac{1}{2}} \sum_{s_k \in N_i(c)} Y_k \right) \right) \right) \right)$$

$$-\frac{1}{2\sigma_U^2}\sum_{i=1}^{m}U_i^T U_i-\frac{1}{2\sigma_S^2}\sum_{j=1}^{n}S_j^T S_j$$
$$-\frac{1}{2\sigma_{BU}^2}\sum_{x=1}^{m}(BU_x)^T(BU)$$
$$-\frac{1}{2\sigma_{BS}^2}\sum_{y=1}^{n}(BS_y)^T(BS)-\frac{1}{2\sigma_Y^2}\sum_{k=1}^{n}Y_k^T Y_k \quad (19)$$

The QoS value $\widehat{q}_{ij}$ of user i invoking service j is predicted by

$$\hat{q}_{ij}=B(i,j,C)+S_j^T\left(U_i+\sum_{c\in G}\left(\omega_c|N_i(c)|^{-\frac{1}{2}}\sum_{s_k\in N_i(c)}Y_k\right)\right) \quad (20)$$

In order to learn the variables in the model, the regularized squared error is minimized as follows.

$$E=\min\sum\left[q_{ij}-B(i,j,c)-S_j\left(U_i+\sum_{c\in G}\omega_c|N_i(c)|^{-\frac{1}{2}}\sum_{l\in N_i(c)}Y_l\right)\right]+\lambda_1\cdot\left(BU_i^2+BS_j^2\right)+\lambda_2\cdot\left(\|U_i\|^2+\|S_j\|^2+\|Y_l\|^2\right) \quad (21)$$

where λ_1 and λ_2 are two parameters to control regularization, and $\|\cdot\|^2$ denotes Frobenius norm. We use the stochastic gradient descent algorithm to optimize Eq. 21, which is computed as follows.

$$\begin{cases}BU_i\leftarrow BU_i+\alpha\cdot\left(e_{ij}-\lambda_1\cdot BU_i\right)\\ BS_j\leftarrow BS_j+\alpha\cdot\left(e_{ij}-\lambda_1\cdot BS_j\right)\\ S_j\leftarrow S_j+\alpha\cdot\left[e_{ij}\cdot\left(U_i+\sum_{c\in G}\omega_c|N_i(c)|^{-\frac{1}{2}}\sum_{l\in N_i(c)}Y_l\right)-\lambda_2\cdot S_j\right]\\ U_i\leftarrow U_i+\lambda_2\cdot\left(e_{ij}\cdot U_i-\lambda_2\cdot U_i\right)\\ \forall l\in N_i(c):Y_l\leftarrow Y_l+\alpha\cdot\left(e_{ij}\cdot\omega_c|N_i(c)|^{-\frac{1}{2}}\cdot S_j-\lambda_2\cdot Y_l\right)\end{cases} \quad (22)$$

where the parameter α is the learning rate.

V. EXPERIMENTS AND EVALUATION

In this section, we evaluate the performance of our proposed prediction model. We also will study the impact of model parameter.

A. DATA SET

In this study, we adopt a public dataset of real-world services provided by Zheng *et al.* [28], which contains 1,974,675 QoS records from 339 users and 5,825 services distributed all over the world. This dataset contains network location information on both user side and service side, IP addresses of users and WSDL (Web Services Description Language) files of services. The data statistics are shown in Table 3.

B. DATA PREPROCESSING

The autonomous system information of the original dataset is not provided. In order to solve this issue, we use a free and public database GeoLite, and map the IP address (in WSDL files) of a user and a service to the corresponding autonomous system number. After mapping, 137 and 1021 autonomous domain codes are obtained, which correspond to 339 users and 5102 services respectively. However, there are 723 services remained that could not be mapped to autonomous domain numbers.

TABLE 3. Data statistics of the services QoS dataset used in our experiments.

Number of users	339
Number of services	5828
Number of invocation records	1974675
Number of user countries	30
Number of service countries	73
Average value of RTT	0.81
Average value of throughput	44.03

To evaluate the performance of our method in different data sparsities, we generate the training sets by randomly removing a part of records, and form four sparsity cases with 5%, 10%, 15%, and 20%. The density of response time in 10%, as an example, from the original response time, a random matrix of size $M\times N\times 0.1$ QoS values is as the training set. The remaining $M\times N\times 0.9$ QoS values is as the test set. We repeat 100 times for each experimental value and report the average result, to avoid the probable instability of methods.

C. EVALUATION METRICS

We use the Root Mean Squared Error (RMSE) to measure the prediction accuracy. RMSE computes the standard deviation of the prediction error, which is computed as

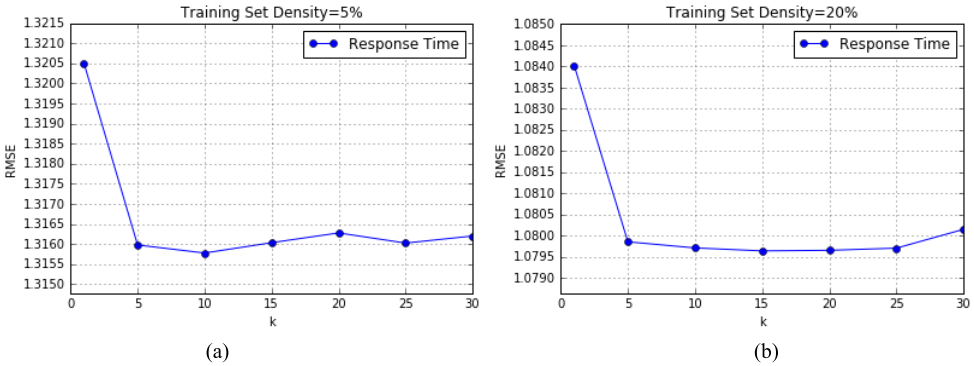
$$RMSE=\sqrt{\frac{1}{N}\sum_{i,j}\left(q_{ij}-\hat{q}_{ij}\right)^2} \quad (23)$$

where q_{ui} represents the real QoS value, $\hat{q}_{ui}$ represents the prediction value, and N is the number of values in the test set. A smaller RMSE value means higher prediction accuracy.

D. PERFORMANCE COMPARISON

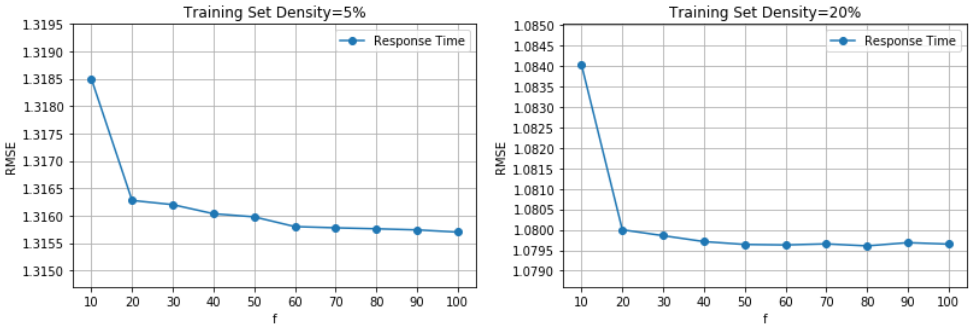
We compare our proposed model with the following well-known approaches that have been developed to predict QoS values, including:

1. UserMean: A user-oriented mean prediction method.
2. ItemMean: An item-oriented mean prediction method.
3. UPCC [29]: The user-based collaborative filtering algorithm using Pearson correlation coefficient (PCC).
4. IPCC [30]: The item-based collaborative filtering algorithm using PCC.
5. WSRec [28]: A hybrid model composed of UPCC and IPCC with confidence weight.



(a) (b)

FIGURE 6. Impact of k. (a) Training set density = 5% (b) Training set density = 20%.



(a) (b)

FIGURE 7. Impact of f. (a) Training set density = 5% (b) Training set density = 20%.

6. LACF [31]: LACF is short for *location-aware collaborative filtering*.
7. PMF [32]: PMF is short for *probabilistic matrix factorization*. PMF model has been explained in Section IV.
8. JLMF [33]: A collaborative recommendation framework containing three prediction models.
9. LRMF [25]: LRMF stands for *location and reputation-aware matrix factorization*.

For parameter setting, the number of users clusters k is set as 20, and the dimension f of latent feature vector is set as 30. Table 4 presents the RMSE values of all methods in four matrices with densities from 5% to 20%. CA-PMF (context-aware PMF) is our proposed method. Based on the results in Table 4, the following observations can be made.

1. The RMSE values of all methods decrease as the training set density increases, which indicates that more invocation records will promote the prediction performance.
2. The proposed method CA-PMF produces smaller RMSE than all compared approaches in all cases of training set densities, which indicates that our model outperforms other models under all different circumstances. It is worth noting that our model performs consistently better than PMF, clearly indicating that the clustering method employed and the use of network location information helps improve QoS prediction accuracy.

TABLE 4. Accuracy comparison.

Model	Training Set Density (TD) —— response time dataset			
	TD = 5%	TD = 10%	TD = 15%	TD = 20%
	RMSE	RMSE	RMSE	RMSE
UserMean	1.8473	1.8472	1.8388	1.8379
ItemMean	1.7765	1.7720	1.7565	1.7534
UPCC	1.5930	1.5741	1.5601	1.5214
IPCC	1.6851	1.6661	1.6517	1.6157
WSRec	1.5409	1.5090	1.4715	1.4245
LACF	1.4908	1.4567	1.4225	1.3655
PMF	1.4320	1.4016	1.3842	1.3468
JLMF	1.3746	1.3203	1.3072	1.2907
LRMF	1.4151	1.2576	1.2128	1.1422
CA-PMF	**1.3157**	**1.2296**	**1.1968**	**1.0796**

E. SENSITIVITY ANALYSIS OF PARAMETERS

1) IMPACT OF k

The parameter k determines the number of user cluster. If k is set to be 1, all users are in the same group. On the one hand, when k is relatively small, we mainly use historical invocation records of users and services to build the bias model, with

limited network location information. On the other hand, if k is quite large, it indicates that there are few users in the same cluster, including some clusters with only one user. In such a case, the bias model changes to the user bias model. Thus, k takes an important role in our QoS prediction method. To investigate the impact of k on the performance of CA-PMF model, the value of k was varied from 1 to 30.

Figure 6 shows the change of RMSE as the value of k varies from 1 to 30 for four different matrix densities settings. From Figures 6(a) and 6(b), it can be observed that the RMSE drops down sharply at beginning and then increases smoothly, which indicates that when k is small, the size of cluster will be large, leading to dissimilar users in the same cluster, which will hamper the performance of our model.

2) IMPACT OF f

In our proposed method, the parameter f determines the number of latent factors. In this section, we evaluate the sensitivity of our method to f which ranges from 10 to 100 with a constant increment of 10. Figure 7 shows the impact of the parameter f on RMSE under different matrix densities settings, where we can find that the prediction accuracy of our model increases with the increase in parameter f. However, it performs stably on RMSE when the parameter f is greater than 50.

VI. CONCLUSIONS AND FUTURE WORK

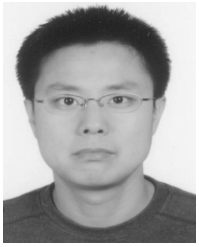
In this study, we propose a novel QoS prediction method to build an effective service recommendation system, which incorporates network location information and implicit associations between users and services. We propose a novel clustering algorithm by improving the initialization, integrating the user's network location information to find similar users. We also propose a novel PMF model. We conduct extensive experiments on a real-world dataset, and the results verify the effectiveness of our models.

In the future, we plan to investigate the performance of our models on more QoS properties, such as reputation and reliability. In addition, we are going to investigate ways of incorporating time factor into the existing models.

REFERENCES

[1] I. M. Delamer and J. L. M. Lastra, "Service-oriented architecture for distributed publish/subscribe middleware in electronics production," *IEEE Trans. Ind. Inform.*, vol. 2, no. 4, pp. 281–294, Nov. 2006.
[2] Y. Liu, A. H. Ngu, and L. Z. Zeng, "QoS computation and policing in dynamic web service selection," in *Proc. 13th Int. World Wide Web Conf. Alternate Track Papers Posters*, 2004, pp. 66–73
[3] Z. Yang, B. Wu, K. Zheng, X. Wang, and L. Lei, "A survey of collaborative filtering-based recommender systems for mobile Internet applications," *IEEE Access*, vol. 4, pp. 3273–3287, 2017.
[4] F. Ye and H. Zhang, "A collaborative filtering recommendation based on users' interest and correlation of items," in *Proc. IEEE Conf. Int. Conf. Audio, Lang. Image Process.*, Jul. 2016, pp. 515–520.
[5] Z. Zheng, H. Ma, M. R. Lyu, and I. King, "QoS-aware Web service recommendation by collaborative filtering," *IEEE Trans. Services Comput.*, vol. 4, no. 2, pp. 140–152, Apr./Jun. 2011.
[6] Z. Sharifi, M. Rezghi, and M. Nasiri, "A new algorithm for solving data sparsity problem based-on Non negative matrix factorization in recommender systems," in *Proc. ICCKE*, Oct. 2014, pp. 56–61.
[7] E. Aslanian, M. Radmanesh, and M. Jalili, "Hybrid recommender systems based on content feature relationship," *IEEE Trans. Ind. Inform.*, vol. 99, p. 1, 2016.
[8] R. Salakhutdinov and A. Mnih, "Probabilistic matrix factorization," in *Proc. NIPS*, 2007, pp. 1257–1264.
[9] T. Yu, Y. Zhang, and K.-J. Lin, "Efficient algorithms for Web services selection with end-to-end QoS constraints," *ACM Trans. Web*, vol. 1, no. 1, 2007, Art. no. 6.
[10] X. Luo, M. Zhou, Y. Xia, and Q. Zhu, "An efficient non-negative matrix-factorization-based approach to collaborative filtering for recommender systems," *IEEE Trans. Ind. Informat.*, vol. 10, no. 2, pp. 1273–1284, May 2014.
[11] N. L. Xuan, T. Vu, and T. D. Le, "Addressing cold-start problem in recommendation systems," in *Proc. Int. Conf. Ubiquitous Inf. Manage. Commun.*, 2008, pp. 208–211.
[12] X. Luo, Y. Lv, R. Li, Y. Chen, "Web service QoS prediction based on adaptive dynamic programming using fuzzy neural networks for cloud services," *IEEE Access*, vol. 3, pp. 2260–2269, 2017.
[13] D. Yu, Y. Liu, and Y. Xu, "Personalized QoS prediction for Web services using latent factor models," in *Proc. IEEE SCC*, Jun./Jul. 2014, pp. 107–114.
[14] D. Zhang, C.-H. Hsu, M. Chen, Q. Chen, N. Xiong, and J. Lloret, "Cold-start recommendation using bi-clustering and fusion for large-scale social recommender systems," *IEEE Trans. Emerg. Topics Comput.*, vol. 2, no. 2, pp. 239–250, Jun. 2017.
[15] G. Li and W. Ou, "Pairwise probabilistic matrix factorization for implicit feedback collaborative filtering," *Neurocomputing*, vol. 204, pp. 17–25, Sep. 2016.
[16] Q. Xie, S. Zhao, Z. Zheng, J. Zhu, and M. R. Lyu, "Asymmetric correlation regularized matrix factorization for Web service recommendation," in *Proc. IEEE Int. Conf. Web Services*, San Francisco, CA, USA, Jun./Jul. 2016, pp. 204–211.
[17] L. Shao, J. Zhang, Y. Wei, J. Zhao, B. Xie, and H. Mei, "Personalized QoS prediction for Web services via collaborative filtering," in *Proc. IEEE Int. Conf. Web Services*, Jul. 2007, pp. 439–446.
[18] D. Zhao, J. Xiu Y. Bai, and Z. Yang, "An improved item-based movie recommendation algorithm," in *Proc. Int. Conf. Cloud Comput. Intell. Syst.*, Aug. 2016, pp. 278–281.
[19] Y. Jiang, J. Liu, M. Tang, and X. Liu, "An effective Web service recommendation method based on personalized collaborative filtering," in *Proc. IEEE ICWS*, Jul. 2011, pp. 211–218.
[20] C. Desrosiers and G. Karypis, "A comprehensive survey of neighborhood-based recommendation methods," in *Recommender Systems Handbook*. New York, NY, USA: Springer, 2011, pp. 107–144.
[21] J. J. Sandvig, B. Mobasher, and R. D. Burke, "A survey of collaborative recommendation and the robustness of model-based algorithms," *IEEE Data Eng. Bull.*, vol. 31, no. 2, pp. 3–13, Jun. 2009.
[22] K. Qi, H. Hu, W. Song, J. Ge, and J. Lü, "Personalized QoS prediction via matrix factorization integrated with neighborhood information," in *Proc. IEEE SCC*, Jun./Jul. 2015, pp. 186–193.
[23] H. Wu, K. Yue, B. Li, B. Zhang, and C.-H. Hsu, "Collaborative QoS prediction with context-sensitive matrix factorization," *Future Gener. Comput. Syst.*, vol. 82, pp. 669–678, May 2017.
[24] W. Lo, J. Yin, S. Deng, Y. Li, and Z. Wu, "An extended matrix factorization approach for QoS prediction in service selection," in *Proc. IEEE SCC*, Jun. 2012, pp. 162–169.
[25] S. Li, J. Wen, F. Luo, T. Cheng, and Q. Xiong, "A location and reputation aware matrix factorization approach for personalized quality of service prediction," in *Proc. IEEE ICWS*, Jun. 2017, pp. 652–659.
[26] P. He, J. Zhu, Z. Zheng, J. Xu, and M. R. Lyu, "Location-based hierarchical matrix factorization for Web service recommendation," in *Proc. IEEE ICWS*, Jun./Jul. 2014, pp. 297–304.
[27] A. Fahad *et al.*, "A survey of clustering algorithms for big data: Taxonomy and empirical analysis," *IEEE Trans. Emerg. Topics Comput.*, vol. 2, no. 3, pp. 267–269, Sep. 2014.
[28] Z. Zheng, H. Ma, M. R. Lyu, and L. , "WSRec: A collaborative filtering based Web service recommender system," in *Proc. IEEE ICWS*, Jul. 2009, pp. 437–444.
[29] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl, "GroupLens: An open architecture for collaborative filtering of netnews," in *Proc. ACM Conf. Comput. Supported Cooperat. Work*, 1994, pp. 175–186.
[30] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, "Item-based collaborative filtering recommendation algorithms," in *Proc. 10th Int. Conf. World Wide Web*, 2001, pp. 285–295

[31] M. Tang, Y. Jiang, J. Liu, and X. Liu, "Location-aware collaborative filtering for QoS-based service recommendation," in *Proc. IEEE ICWS*, Jun. 2012, pp. 202–209.
[32] Y. Xu, J. Yin, W. Lo, and Z. Wu, "Personalized location-aware QoS prediction for Web services using probabilistic matrix factorization," in *Proc. WISE*, 2013, pp. 229–242.
[33] Y. Yin, S. Aihua, G. Min, X. Yueshen, and W. Shuoping, "QoS prediction for Web service recommendation with network location-aware neighbor selection," *Int. J. Softw. Eng. Knowl. Eng.*, vol. 26, no. 4, pp. 611–632, May 2016.



YUYU YIN (M'12) received the Ph.D. degree in computer science from Zhejiang University in 2010. He is currently an Associate Professor with the College of Computer, Hangzhou Dianzi University. He is also a Supervisor of master students with the School of Computer Engineering and Science, Shanghai University, Shanghai, China. During the past 10 years, he has published more than 40 papers in journals and refereed conferences, such as *Sensors*, *Entropy*, IJSEKE, *Mobile Information Systems*, ICWS, and SEKE. His research interests include service computing, cloud computing, and business process management. He is a member of the China Computer Federation (CCF) and the CCF Service Computing Technical Committee. He organized more than 10 international conferences and workshops, such as FMSC2011-2017 and DISA2012\2017-2018. He served as a Guest Editor for the *Journal of Information Science and Engineering* and the *International Journal of Software Engineering and Knowledge Engineering* and a Reviewer for the IEEE Transaction on Industry Informatics, the *Journal of Database Management*, and *Future Generation Computer Systems*.

LU CHEN received the bachelor's degree in computer science from Hangzhou Dianzi University, Zhejiang, China, where she is currently pursuing the master's degree in computer science with the Key Laboratory of Complex Systems Modeling and Simulation, Ministry of Education. Her research interests include recommendation system and machine learning.

YUESHEN XU received the Ph.D. degree from Zhejiang University. He was a co-trained Ph.D. student at the University of Illinois at Chicago. He is currently a Lecturer with the School of Computer Science and Technology, Xidian University. He has published more than 20 papers in international journals or conferences, such as ESWA, EAAI, WISE, WAIM, and *Sensors*. His research interests include recommender system, text mining, and service computing. He is a member of ACM and CCF. He is a Reviewer of several journals and conferences, including the IEEE TSC, KBS, the IEEE Access, JNCA, *Neurocomputing*, and ICDCS.

JIAN WAN received the Ph.D. degree in computer application technology from Zhejiang University, Zhejiang, China, in 1989. He is currently a Professor in software engineering with the Key Laboratory of Complex Systems Modeling and Simulation, Ministry of Education, and the School of Computer Science and Technology, Hangzhou Dianzi University. His research interests include grid computing, service computing, and cloud computing.