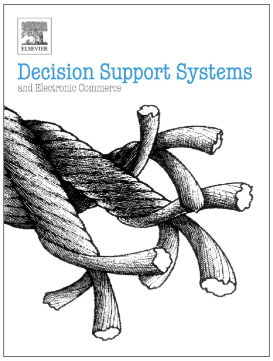


Time-aware cloud service recommendation using similarity-enhanced collaborative filtering and ARIMA model

Shuai Ding, Yeqing Li, Desheng Wu, Youtao Zhang, Shanlin Yang

PII: S0167-9236(17)30241-5
DOI: https://doi.org/10.1016/j.dss.2017.12.012
Reference: DECSUP 12913

To appear in: *Decision Support Systems*

Received date: 18 March 2017
Revised date: 8 December 2017
Accepted date: 25 December 2017

Please cite this article as: Shuai Ding, Yeqing Li, Desheng Wu, Youtao Zhang, Shanlin Yang , Time-aware cloud service recommendation using similarity-enhanced collaborative filtering and ARIMA model. The address for the corresponding author was captured as affiliation for all authors. Please check if appropriate. Decsup(2017), https://doi.org/10.1016/j.dss.2017.12.012

Time-aware Cloud Service Recommendation Using Similarity-enhanced Collaborative Filtering and ARIMA Model

Shuai Ding[a,b], Yeqing Li[a,b], Desehng Wu [c,*],Youtao Zhang[d], Shanlin Yang[a,b*]

[a] *School of Management, Hefei University of Technology, Anhui, Hefei 23009, China*

[b] *Key Laboratory of Process Optimization and Intelligent Decision-Making (Ministry of Education), Hefei University of Technology, Anhui, Hefei 23009, China*

[c] *Economics and Management School, University of Chinese Academy of Sciences, Beijing, 100190, China and Stockholm University*

[d] *Department of Computer Science, University of Pittsburgh, Pittsburgh 15213, PA, USA*

Corresponding author:

Both Desheng Wu and Shanlin Yang are Corresponding authors.

Desheng Wu, Email: dash@risklab.ca, dash.wu@gmail.com

Time-aware Cloud Service Recommendation Using Similarity-enhanced Collaborative Filtering and ARIMA Model

Abstract

The quality of service (QoS) of cloud services change frequently over time. Existing service recommendation approaches either ignore this property or address it inadequately, leading to ineffective service recommendation. In this paper, we propose a time-aware service recommendation (taSR) approach to address this issue. We first develop a novel similarity-enhanced collaborative filtering (CF) approach to capture the time feature of user similarity and address the data sparsity in the existing PITs (point in time). We then apply autoregressive integrated moving average model (ARIMA) to predict the QoS values in the future PIT under QoS instantaneity. We evaluate the proposed approach and compare it to the state-of-the-art. Our experimental results show that taSR achieves significant performance improvements over existing approaches.

Keywords: cloud service, time-aware recommendation, QoS, similarity-enhanced CF, ARIMA

1. Introduction

With the rapid development of cloud computing technology in the past decade, cloud services have prevailed in various application domains. While there are a large number of cloud services in commercial service markets, e.g., Apple APP store and Tencent application treasure, many of these services share similar or even overlapped functionalities. Recent studies have shown that, for either individual users or small and medium enterprise (SME) users, adopting appropriate cloud services can significantly reduce IT cost and increase operation efficiency, which has made cloud service recommendation and selection one of the most important tasks in cloud computing.

Given the difficulty in choosing appropriate services from a large set of services candidates that share same or similar functionalities, cloud users depend increasingly on recommendations from the cloud service vendors. In addition to the functionality information of cloud services, the cloud service vendors may collect non-functional information, such as response time, throughput, cost [1], referred to as QoS (quality of service) indicators, to better characterize the services. QoS-based service recommendation systems, e.g., kernel-based quantile estimator [2], clustering algorithm [3], and deviation-based neighborhood model [4], achieved better recommendation over the baseline that recommends services only using the functionality information.

However, recent studies revealed that QoS indicators exhibit strong instantaneity in the cloud computing environment [18], e.g., the response time of a service depends on the real-time network traffic as well as the computing load of the end users' equipment. They differ significantly from those in other recommendation systems, e.g., E-commence recommendation focuses on user-generated comments, blogs and discussion posts that remain stable for hours or even weeks. Recent advances in cloud service recommendation started to adopt time-aware approaches to address QoS instantaneity. Such as [18], which applies linear combination to fit the influence of time in QoS prediction. In a word, the CF-model based approaches [5] utilize attenuation function to solve the dynamic of user similarity caused by QoS changes, and the ARIMA-based approaches [17] captured the instantaneity in time-aware long-term QoS prediction. However, CF-based models are not effective in predicting QoS values at future PITs while ARIMA-based approaches fail to address the data sparsity in the real world.

In this paper, we propose a novel time-aware cloud service recommendation approach (taSR) for cloud service vendors. By better exploration of QoS instantaneity, the service vendors can recommend services that match users' demands better, which effectively addresses the limitations in existing methods. The followings summarize our contributions.

(1) TaSR, by combining CF model and ARIMA model, exploits the advantages of both models. TaSR adopts a CF method to replenish the missing QoS values such that the data series are ready for constructing effective ARIMA model. It then exploits ARIMA model to precisely capture the dynamics of QoS values and predict QoS values at future PITs (point in time). TaSR formulates the service recommendation as multi-criterion decision-making (MCDM) problem, which normalizes and weights in multiple QoS indicators, for better service recommendation.

(2) TaSR, by integrating user global similarity and user invocation similarity, improves the CF method for time-aware user similarity estimation. The proposed similarity-enhanced CF approach can comprehensively capture the dynamics of user similarity and accurately predict the missing QoS values at either a past PIT and the current PIT.

(3) We evaluate the proposed taSR approach and compare it to the state-of-the-art. The experimental results show that taSR achieves significant improvements over CF-based approaches and ARIMA-based approaches in various settings.

For the rest of the paper, Section 2 reviews the related background. Section 3 presents an overview of the proposed taSR approach. We elaborate the time-aware similarity estimation and the prediction model in Section 4 and Section 5, respectively. Section 6 discusses the experiments and analyzes the results. We summarize the paper in Section 7.

2. Literature review

In the last decade, QoS analysis based approaches have demonstrated their effectiveness in cloud service recommendation. Most cloud service recommendation systems adopt CF (collaborative filtering) based approaches, which can be divided into neighbor-based and model-based approaches [7]. The neighbor-based approaches may be further categorized into three kinds based on the type of neighbors: user-similarity based [8][9], item-similarity based [10][11], and hybrid-similarity based [12][13]. The first two predict the QoS values according to the values of their similar users and services, respectively, for improved prediction accuracy, and the last one integrates user similarity and service similarity in estimation. Meanwhile, model-based approaches have been used in QoS prediction. For examples, Silic et al. [14] presented CLUS to divide users/services into different groups based on the k-means algorithm. Yu et al. [15] applied trace norm regularized matrix factorization to predict the reliability of web services.

These traditional schemes adopt static prediction model and thus show good performance only for a specified PIT (point in time). Since the QoS values of cloud services change with different network connection and workload in different invocation time, it is necessary to take time into consideration in the estimation of user/service similarity. For example, two services that show high similarity one month ago may not be treated to be similar as the cloud hardware may have been updated over the time. Some CF-based approaches have proposed time-aware user similarity estimation to capture QoS instantaneity. Qi et al. [16] weighted the similarity according to the time span between the invocation time with an exponential decay function. Hu et al. [5] designed an exponential decay function to weight the similarity but according to the time span between invocation time to current.

However, very few CF-based approaches have the capacity to precisely characterize the temporal dynamics of QoS [17]. Currently, there are models proposed to find the correlation between different invocations. Wang et al. [18] predicted the current QoS by the linear combination of similar sequences and estimated the linear combination coefficients by Lasso. Ye et al. [6] proposed to integrate ARIMA model and Holt-Winters model for better long-term QoS performance. Chu et al. [19] proposed a time-aware Bayesian network to discover the time dependent quality of service relationships structure. Geebelen et al. [5] used kernel-based quantile estimator, a powerful non-linear black-box regressor, with online adaptation of the constant offset to predict future QoS values. Hu et al. [17] took the latest observation as a feedback to revise forecasts to future QoS values for each individual service with Kalman filtering.

In summary, the traditional CF approaches failed to precisely characterize the temporal dynamics of QoS, even though the construction of time-aware models need high quality QoS data. The existing time-aware QoS-based recommendation schemes cannot fully address the instantaneity of QoS values. In this article, we first propose novel similarity-enhanced CF approach to capture the time feature of similarity estimation and replenish the missing QoS value for further prediction. The ARIMA model is then applied to predict the QoS values in the near future for better accuracy. Finally, as a MCDM problem, the different indicators are combined to recommend cloud services.

3. taSR: Time-aware Cloud Service Recommendation

Clearly, it is crucial to address the instantaneity of QoS values for better service recommendation. In this paper, we propose a time-aware service recommendation approach that integrates the time-aware similarity-enhanced CF and the ARIMA model. The similarity-enhanced CF improves the calculation of user similarity for better QoS data filling, and the ARIMA model predicts the QoS values at a future PIT more accurately with high quality data. The procedure consists of three steps, as shown in Fig. 1.

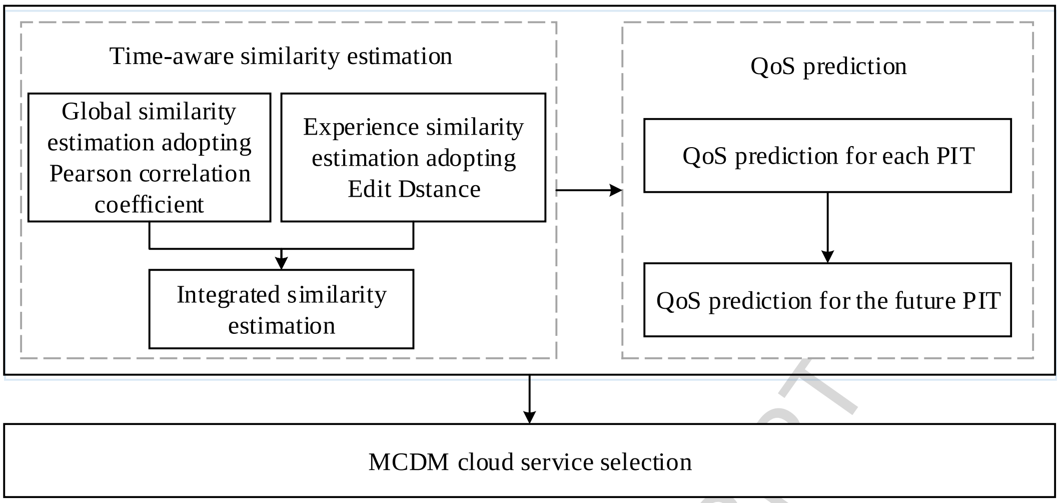

Time-aware similarity estimation
Global similarity estimation adopting Pearson correlation coefficient
Experience similarity estimation adopting Edit Dstance
Integrated similarity estimation
QoS prediction
QoS prediction for each PIT
QoS prediction for the future PIT
MCDM cloud service selection

Fig. 1. The taSR time-aware cloud service recommendation.

In the first step, taSR adopts a novel similarity metric to reconcile dynamic user similarity based on instantaneous QoS values. We use PCC (Pearson correlation coefficient) to calculate the global similarity based on QoS values， and adopt a custom attenuation function to adjust QoS prediction based on the user’s risk preference. We then evaluate the user invocation similarity based on the adoption of edit distance. We integrate the two similarity values in one metric using their geometric mean.

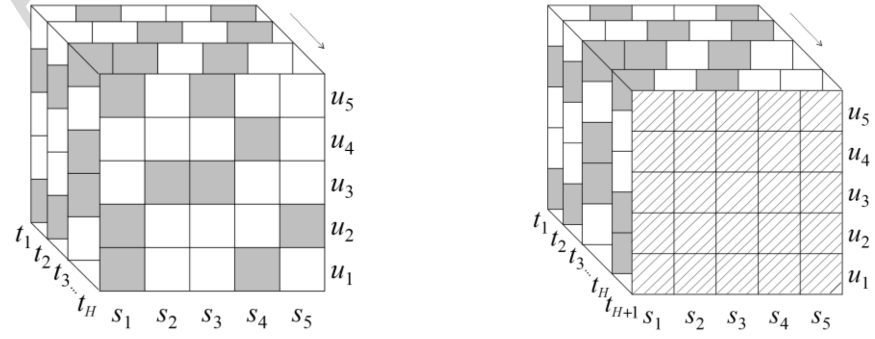
In the second step, taSR employs the similarity estimated in the first step and selects the users that are most similar to the target user to fill missing QoS values in the past and current PITs. Moreover, we adopt the ARIMA model to extend the QoS prediction to include not only the past and current PITs but also a future PIT with faithful description of QoS instantaneity.

The cloud service section is a multi-criterion decision-making (MCDM) problem when adopting different indicators to evaluate QoS performance. In the last step, with the filled user-service matrices, considering the inconsistency of different QoS indicators, we normalize the QoS values and weight the indicators to integrate the comprehensive QoS values. We then rank the QoS values and recommend the top-k candidate services.

4. Time-aware user similarity estimation

In this section, we elaborate the similarity estimation details in taSR. We discuss the model used in similarity estimation, present our similarity metric, and integrate the similarity estimation with edit distance.

The user-service matrix is a model for modeling the relationship between users and services, which is widely adopted in current cloud service recommendation studies. To model QoS instantaneity, the 2D matrix can be extended with time direction, resulting in time series user-service matrices, as shown in Fig. 2. In the figure, u_i ($i \in [1, I]$) and s_j ($j \in [1, J]$) denote different users, and cloud services, respectively. An entry in the matrix Q_{ij}^{h} denotes the observed/predicted QoS value to be used in similarity estimation (e.g., response-time, throughput). t_h (h=1 .. H, $H+1$) denotes different PITs (points in time). t_1 denotes the system start PIT while t_H and t_{H+1} denote the current PIT, and the next PIT in the future, respectively.



(a) The collected raw matrices (b) The predicted matrix for the future PIT

Fig. 2. Modeling the user, service, and PIT relationship.

4.1. User Global Similarity Estimation

Since user similarity analysis plays the key role in cloud service recommendation, we focus on better user similarity analysis approaches in this section. In particular, we adopt a novel similarity metric to address the dynamic nature of user similarity. Recent studies have revealed that, of all user similarity results at different PITS, those from recent PITs tend to have a larger impact [5].

The basic inter-user similarity is modeled by PCC (Pearson correlation coefficient) using Eq. (1). In the equation, the service set includes all the services with the similar function that the users used before. Q_{pj}^{h} and Q_{qj}^{h} denote the QoS values of service s_j invoked by users u_p and u_q, respectively, at PIT t_h. $\overline{Q}_{p}^{h}$ and $\overline{Q}_{q}^{h}$ denote the average QoS values of all service candidates invoked by u_p and u_q.

$$Sim_{pq}^{h} = \frac{\sum_{j=1}^{J}(Q_{pj}^{h} - \overline{Q}_{p}^{h})(Q_{qj}^{h} - \overline{Q}_{q}^{h})}{\sqrt{\sum_{j=1}^{J}(Q_{pj}^{h} - \overline{Q}_{p}^{h})^{2}}\sqrt{\sum_{j=1}^{J}(Q_{qj}^{h} - \overline{Q}_{q}^{h})^{2}}} \tag{1}$$

We then model the attenuation of similarity correlation over time. Studies have shown that the risk preference of cloud users greatly affects their behaviors [20]. There are three types of users: risk-averse, risk-neutral and risk-taking users. While risk-averse users pay more attention to the performance of recently invoked services, risk-taking users have less interest in recent PITs. A risk neutral user is often calm to the change of QoS values. Traditional methods, like questionnaire [21], expected-utility mode [22], have been applied to collect the risk preference information. In this paper, we assume cloud service vendors gain users' risk preference using history invocation information as a feedback.

There exist two main types of attenuation functions, i.e., logistic function and exponential function [23]. A logistic function is an S-curve function while an exponential function is a concave one. To precisely capture the risk preference and the evolution trend of user similarity, we propose a custom attenuation function as shown in Eq. (2).

$$f(h) = \frac{2}{1+(h/H)^{-\alpha}} \tag{2}$$

where H denotes the total number of PITs. In this way, we make h/H vary from 0 to 1, and α is a tunable parameter adjusted to indicate the risk preference of service similarity. Fig. 3 plots the function with different α values. From the figure, the correlation of inter-user similarity changes significantly when α falls below and exceeds 1.

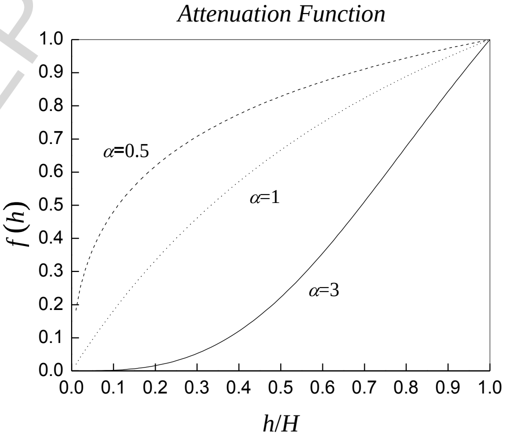


Fig. 3. The similarity attenuation function.

We then estimate the user global similarity at consecutive PITs t_1 .. t_H using Eq. (3).

$$Sim_{pq}^{Global} = \frac{\sum_{h=1}^{H}\left(f(h) * Sim_{pq}^{h}\right)}{\sum_{h=1}^{H} f(h)} \tag{3}$$

The user global similarity proposed in this article is a dynamic metric that is closely coupled with the attenuation function $f(h)$. α is the coefficient of users' risk preference. A risk-averse user may choose α being smaller than 1, which has slow attenuation rate and reflects the large impact of recent PITs; a risk-taking user may choose α being bigger than 1, which has fast attenuation rate such that the recommendation of cloud services depends less on recent PITs. Meanwhile, a risk neutral user chooses α being 1. We initialize α to 1.0 indicating every user is neutral at the beginning. In addition to recommending top N services under the current α, the vendor may supply additional m services using a smaller α and m services using a large α. m is much smaller than N. We update α dynamically according to the user's choice. Once a user selects the cloud service recommended by a larger α, we adjust the initial α to a larger one for the next time. Empathy, we adjust α to fit the users' risk preference.

4.2. User Invocation Similarity Estimation Adopting Edit Distance

To improve the accuracy of user similarity estimation, we next exploit users' service invocation history to enable additional similarity analysis among different users, similar as those in recent studies [12][24][25]. Two users are regarded as similar if they either invoke or do not invoke a service at a given time. In this paper, we convert the service invocation record of each user-service pair to a binary string with the values in the string sorted by their invocation times. Therefore, we quantify the similarity using edit distance (ED), a method that has been widely adopted to evaluate the similarity of different strings [26][27][28]. Adopting edit distance helps to capture not only the same invocation but also the same un-invocation, i.e., a service is not invocated by either user. The latter is often ignored in existing methods that estimate user invocation similarity [29].

We next elaborate the similarity estimation using invocation experiences of the set of services that the users used before with similar function. When a user u_p invokes service s_j at consecutive PITs t_1 .. t_H, $\{Q_{pj}^h, h=1..H\}$ denotes their raw QoS values. taSR first maps the QoS values into a binary string $B_{pj}=\{b_{pj}^h, h=1..H\}$, where $b_{pj}^h=1$ if $Q_{pj}^h \neq null$ and $b_{pj}^h=0$ otherwise. Then we conduct the |B_{pj}|+1 row and |B_{qj}|+1 column distance matrix D[|B_{pj}|+1,|B_{qj}|+1]=D[m,n], where D[m,n] is the $edit_{pq}^j(m,n)$ computed by:

$$\begin{cases} edit_{pq}^j(0,0)=0, & m=n=0 \\ edit_{pq}^j(m,0)=m, & m>0 \ and \ n=0 \\ edit_{pq}^j(0,n)=n, & m=0 \ and \ n>0 \\ edit_{pq}^j(m,n)=min\begin{cases} edit_{pq}^j(m,n-1)+1 \\ edit_{pq}^j(m-1,n)+1 \\ edit_{pq}^j(m-1,n-1)+f_{pq}^j(m,n) \end{cases} & ,m>0 \ and \ n>0 \end{cases} \tag{4}$$

where $f_{pq}^j(m,n)=1$ if the (m-1)th value in B_{pj} is equal to the (n-1)th value in B_{qj}, and $f_{pq}^j(m,n)=0$ otherwise.

Once m=n=H+1, we get the final edit distance between B_{pj} and B_{qj} --- $edit_{pq}^j(H+1,H+1)$. Then taSR calculates the invocation similarity between u_p and u_q for service s_j as follows.

$$Sim_{pq}^j = 1-\frac{edit_{pq}^j(H+1,H+1)}{H} \tag{5}$$

At last, taSR estimates the user invocation similarity by calculating the average similarity for cloud services invocated by user u_p or u_q, that is, $Sim_{pq}^{Exp}=\sum_{j=1}^{J} Sim_{pq}^j / N_{pq}$,where N_{pq} is the number of cloud services invocated by user u_p or u_q. Note that if two users have the same invocation experiences for all candidates, we have $Sim_{pq}^{Exp} = 1$. For instance, assume two users u_p and u_q invoke service s_j at four consecutive PITs t_{1} .. t_{4}, the collected response-time values (i.e., the QoS values) are {1.57, 2.31, null, null} and {0.98, null, 1.86, null} for two users, respectively. taSR shall transform them to the binary strings as $B_{pj} = \{1,\ 1,\ 0,\ 0\}$ and $B_{qj} = \{1,\ 0,\ 1,\ 0\}$.

Table 1

The constructed computation matrix

	b_{pj}	1	1	0	0
b_{qj}	0	1	2	3	4
1	1	/	/	/	/
0	2	/	/	/	/
1	3	/	/	/	/
0	4	/	/	/	/

Table 2

The edit distance between u_p and u_q

	b_{pj}	1	1	0	0
b_{qj}	0	1	2	3	4
1	1	**0**	1	2	3
0	2	1	1	1	2
1	3	2	1	2	2
0	4	3	2	1	**2**

Table 1 and Table 2 show how to fill in the distance matrix *D*[5,5]. The values in the first row and the first column are calculated by $edit_{pq}^{j}(m,0)=m$ and $edit_{pq}^{j}(0,n)=n$, respectively, as shown in Table 1. We then fill in other edit distance values in the distance matrix, as shown in Table 2. For example, *D*[2,2] is estimated as *min* {1+1,1+1,0+0} = 0, where $f_{pq}^{j}(2,2)=0$. Finally, the edit distance between B_{pj} and B_{qj} is $edit_{pq}^{j}(5,5)=2$.

4.3. Integrated similarity estimation

From the above discussion, the user global similarity extracts traditional user-service relationship and adopts attenuation function to emphasize the dynamic nature of user similarity; the user invocation similarity extracts the hidden information from invocation history from all PITs. To achieve better similarity estimation, taSR integrates the two similarity values in one time-aware similarity metric using geometric mean.

$$Sim_{pq}=\sqrt{Sim_{pq}^{Global}\times Sim_{pq}^{Exp}} \qquad (6)$$

Here, Sim_{pq} is within [0, 1] and a larger value stands for better user similarity.

5. Time-aware cloud service recommendation

5.1. QoS prediction for a past or the current PIT

Given the time series user-service matrices, we predict their missing QoS values at either a past or the current PIT using our similarity-enhanced CF. To prevent distraction from users with low similarity, we rank the users based on the measure of the similarity to the target user, i.e., using Eq. (6), and pick up the top *k* users. The missing QoS value $\hat{Q}_{pj}^{h}$ of sj for u_p at PIT t_h is calculated as follows.

$$\hat{Q}_{pj}^{h} = \bar{Q}_{p}^{h} + \frac{\sum_{q=1}^{k} Sim_{pq} \times (Q_{qj}^{h} - \bar{Q}_{q}^{h})}{\sum_{q=1}^{k} Sim_{pq}} \tag{7}$$

where $\{Q_{qj}^{h}, q = 1..k\}$ denotes the available QoS values of u_q in service s_j in PIT t_h. Sim_{pq} denotes the similarity between u_p and $u_{q,}$ which is computed from Eq. (6); $\bar{Q}_{p}^{h}$ denotes the average QoS value of u_p at PIT t_h. We adopt CF for QoS prediction at either a past or the current PIT because it shows high efficiency and stable performance in information filtering.

5.2. QoS Prediction for the future PIT

We next elaborate QoS value prediction for a future PIT. Given all QoS values are missing at a future PIT, we adopt the ARIMA model [30], a model that has been widely applied for time series based future value prediction. Comparing to existing ARIMA-based recommendation approaches [31][32], our proposed taSR approach improves service recommendation through better similarity analysis.

ARIMA model uses Box-Jenkins approach to predict future values, including model identification, parameter estimation, model checking. There are three equations in ARIMA model constructed as Eq.(8). The AR(ε) equation captures QoS value instantaneity, i.e., the QoS values at PIT t_{H+1} are affected by the QoS values at PITs $t_{H-\varepsilon+1}$ .. t_H and the autoregressive coefficient φ_ε. The MA(ξ) equation means Q_{pj}^{H+1} depends on the random errors r_{pj}^h ($h = H-\xi+1$ .. H) and the moving average coefficients coefficient θ_ξ. The ARMA(ε, ξ) equation indicates that Q_{pj}^{H+1} is both affected by the QoS values and random errors.

$$Q_{pj}^{H+1} = \begin{cases} \sum_{h=H-\varepsilon+1}^{H} \varphi_\varepsilon Q_{pj}^h + r_{pj}^H, & AR(\varepsilon), \text{ACF decays and PACF cuts off} \\ \sum_{h=H-\xi+1}^{H} \theta_\xi r_{pj}^h + r_{pj}^H, & MA(\xi), \text{ACF cuts off and PACF decays} \\ \sum_{h=H-\varepsilon+1}^{H} \varphi_h Q_{pj}^h + \sum_{h=H-\xi+1}^{H} \theta_\xi r_{pj}^h + r_{pj}^H, & ARMA(\varepsilon, \xi), \text{ACF and PACF decays} \end{cases} \tag{8}$$

where Q_{pj}^{H+1} denotes the predicted QoS value for user u_p on service s_j at the future PIT t_{H+1}.

To determine the equation to use, we firstly preprocess the QoS values to obtain the stationary data by difference equation; we then adopt Auto-Correlation Function (ACF) and Partial Auto-Correlation Function (PACF) [30] to identify the model according to the evolution characteristic of QoS values as shown in Table 3. Next, the least square is applied to estimate the model coefficient φ_ε and θ_ξ. Moreover, the Bayesian Information Criterion is utilized to check the model (the parameters ε and ξ). It is an iterative process to find the best model for QoS prediction.

Table 3 ACF and PACF of three models

	AR(ε)	MA(ξ)	ARMA(ε, ξ)
ACF	decay	cut off in ξ steps	decay
PACF	cut off in ε steps	decay	decay

Based on Table 3, we use the AR(ε) equation when the PACF curve cuts off in ε steps and the ACF curve decays. If the ACF curve cuts off in ξ steps and the PACF curve decays, we adopt the MA(ξ) to predict the QoS values. Once both of the ACF and PACF decay, we utilize the ARMA(ε, ξ) equation in prediction.

5.3 Cloud service selection

We recommend cloud services based on the predicted QoS values of the future PIT. Different QoS values may not always be consistent. For example, while we prefer lower response-time values and larger throughput values, one cloud service may have low response-time but also low throughput values. To achieve effective service recommendation, we next formulate the problem as an MCDM (multi-criteria decision making) [33]. At a given PIT, suppose Q_j^l ($l\epsilon[1,L]$, $j\epsilon[1,J]$) is the QoS value of indicator l of cloud service s_j. We first use the widely adopted extremum method to normalize QoS values.

$$\hat{Q}_j^l = \begin{cases} \dfrac{Q_j^l - minQ_j^l}{maxQ_j^l - minQ_j^l}, \ benefit\ indicator \\ \dfrac{maxQ_j^l - Q_j^l}{maxQ_j^l - minQ_j^l}, \ cost\ indicator \end{cases} \tag{15}$$

where $min\,Q_j^l$ and $max\,Q_j^l$ are the minimum and maximum QoS value of indicator l and $\hat{Q}_j^l$ is the normalized QoS value of indicator l of s_j in the future given PIT.

We then weight the indicators according to the variance as $w_l = S_l / \sum_{l=1}^{L} S_l$ [34], where S_l is the variance of normalized QoS values $\hat{Q}_j^l$ of all cloud services in a given PIT. Since users tend to pay more attention to QoS indicators that have large difference [34], this method assigns larger weights to indicators with larger variances. In our future work, we plan to enhance the weight assignment by taking the different preferences from the individual users.

We finally integrate the QoS values of different indicators of cloud service s_j as $Q_j = \sum_{l=1}^{L} w_l Q_j^l$, where Q_j is the aggregated QoS value of s_j for the future PIT. We rank the cloud service according to the aggregated QoS value and recommend the top cloud services.

6. Experiments

6.1. Data description

To study the effectiveness of our proposed taSR approach, we conducted experiments to compare its performance with the state-of-the-art time-aware service recommendation approaches. We adopted the open QoS dataset from WS-DREAM [12], which is the most representative dataset and has been widely adopted in QoS studies [35][36][37][38][39]. The dataset consists of 4532 distributed services collected from 142 distributed computers (i.e., users) located in 57 countries. Each computer invokes services (e.g., apps in Tencent Cloud platform) randomly with a time interval of 15 minutes such that one sequence contains at most 64 PITs, lasting for 16 hours. Two different QoS indicators, i.e., response-time (rt) and throughput (tp), are used to represent the user-side personalized QoS. In particular, the QoS values in this dataset exhibit instantaneity, as shown in [38][39], which makes the dataset an appropriate one in our experiment. We randomly extracted two 120*500*64 time-aware service-user matrices (response-time, throughput) from the original dataset for the experiments. Fig. 4 shows the distribution of response-time and throughput of QoS values in the dataset. In the experiments, t_{64} is treated as the future PIT so that all of its QoS values are to be predicted.

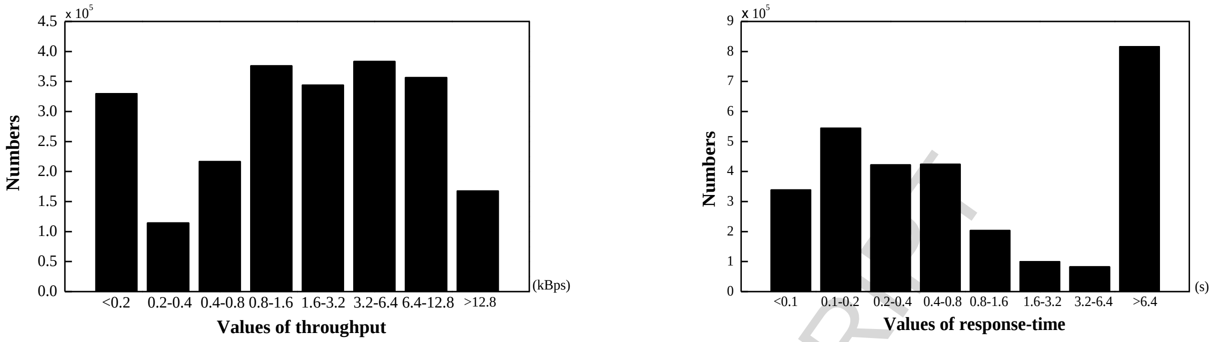


Fig.4. QoS value distributions.

6.2. Evaluation metrics

Matrix density. Given a test dataset, we randomly remove a subset of data values to simulate data sparsity, i.e., some QoS values are missing in the real world. The matrix density is defined as the percentage of available QoS values in the time-aware user-service matrix. As an example, suppose there are 10 users and 5 cloud services in 8 PITs. After removing 241 QoS values, we have the matrix density being 1-241/(10*5*8)=0.3975. For the dataset used in the experiments, we have the matrix density being *D*=1-*N*/(120*500*63), where N denotes the total number of missing QoS values. We vary the matrix density *D* from 0.05 to 0.5 with the step being 0.05.

MAE and RMSE. Given that taSR adopts rating-oriented CF, we used Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) to assess the prediction accuracy and to compare different approaches [40]. MAE and RMSE are defined as:

$$MAE = \frac{\sum_{p,j,h} \left| Q_{pj}^{h} - Q_{pj}^{h} \right|}{N} \tag{9}$$

$$RMSE = \sqrt{\frac{\sum_{p,j,h}(Q_{pj}^{h} - \overset{h}{Q}_{pj})^{2}}{N}} \tag{10}$$

where Q_{pj}^{h} and $\overset{h}{Q}_{pj}$ denote the observed and the predicted QoS values, respectively, for service s_j invoked by user u_p at PIT t_h. N is the total number of addition operations of the numerator, which equals to the total number of missing QoS values that we predicted. Note, MAE and RMSE decrease with increasing accuracy of QoS prediction.

NDCG. To evaluate the service recommendation, we adopted Normalized Discounted Cumulative Gain (NDCG) to assess the ranking accuracy of the *k* recommended services, i.e., the top-*k* candidates. NDCG is a gain-based evaluation metric focusing on the ranking prediction performance [41]. It calculates the performance according to the order of corresponding QoS values but not the values themselves. While throughput is a benefit-based QoS indicator, response-time is a cost-based QoS indicator. Therefore, we first normalized them to benefit indicators to obtain the rank of each QoS value. NDCG$_k$ is defined as follows.

$$NDCG_k = \frac{DCG_k}{IDCG_k}, \tag{11}$$

where DCG$_k$ and IDCG$_k$ denote the discounted cumulative gain on top-*k* ranked services according to the generated recommendation list and the ideal ranking, respectively. DCG$_k$ is computed as follows.

$$DCG_k = rel_1 + \sum_k \frac{rel_k}{\log_2 k}, \tag{12}$$

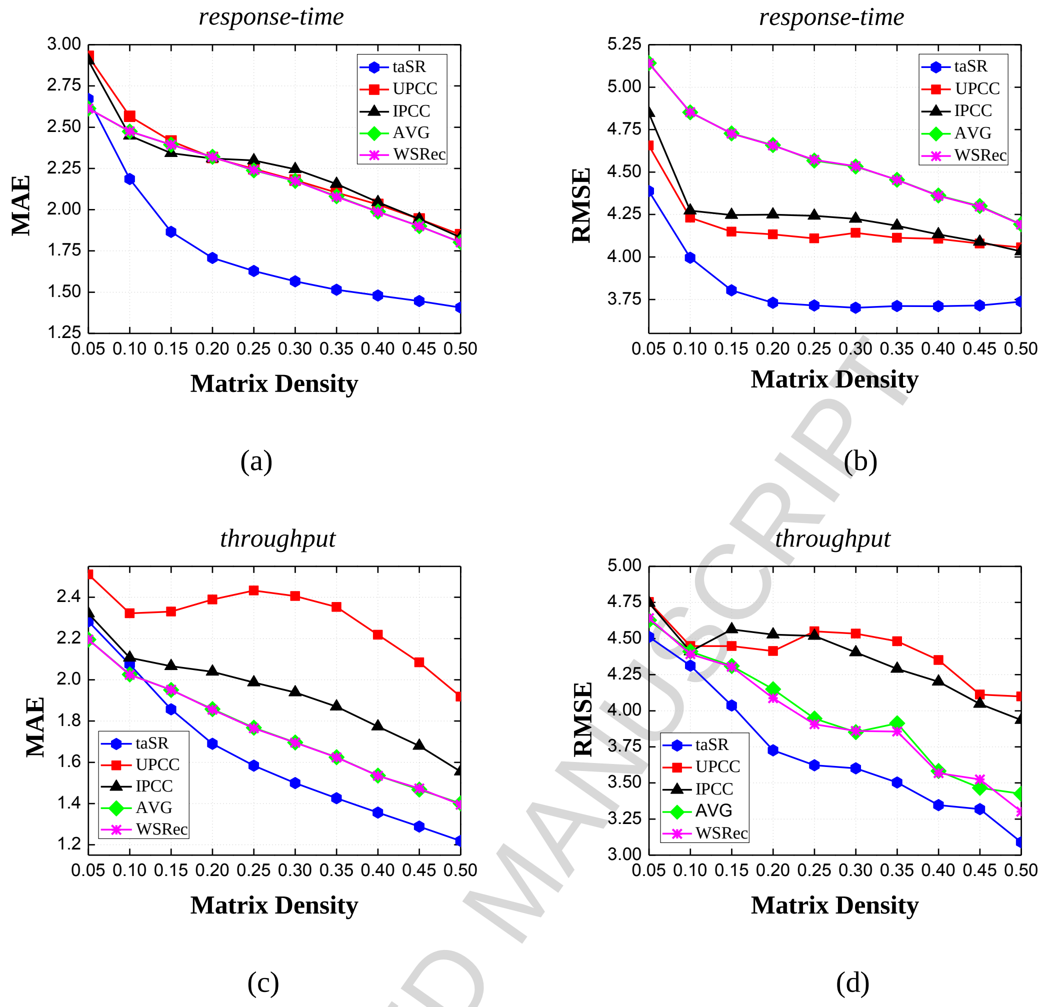
where rel_k denotes the real QoS values on service s_k at position k in the predicted ranking. If the generated recommendation is close to the ideal QoS ranking, DCG$_k$ approximates IDCG$_k$ such that NDCG$_k$ is close to 1. That is, the closer the value is to 1, the better performance the service recommendation approach has.

6.3. Results

In this section, we compared our proposed taSR approach with traditional rating-oriented CF-based approaches and a state-of-the-art time-ware ARIMA-based Kalman approach [17]. In these experiments, we set the parameter α to 1 for constructing the similarity attenuation function in Eq. (2); we select the top-10 similar neighbors for predicting missing QoS values.

6.3.1. Comparison with the rating-oriented approaches

We first compared taSR with three conventional rating-oriented CF-based approaches that are widely applied in recommendation --- user-based CF using PCC (UPCC) [42], item-based CF using PCC (IPCC) [43] and WSRec [12]. UPCC is a method that adopts user-based PCC to predict the missing QoS values based on the observed ones. IPCC is similar to UPCC but adopts item-based PCC. WSRec integrates UPCC and IPCC to predict the missing values. WSRec is often used as a baseline for performance comparison [17][18]. AVG is an approach that fills the user-service matrix with the average of the observed QoS values in the last 3 PITs [18]. In the experiments, we first filled in the 63 PITs with four approaches and our similarity-enhanced CF approach, and then predicted the future QoS values in PIT t_{64} with the ARIMA model.



(a)

(b)

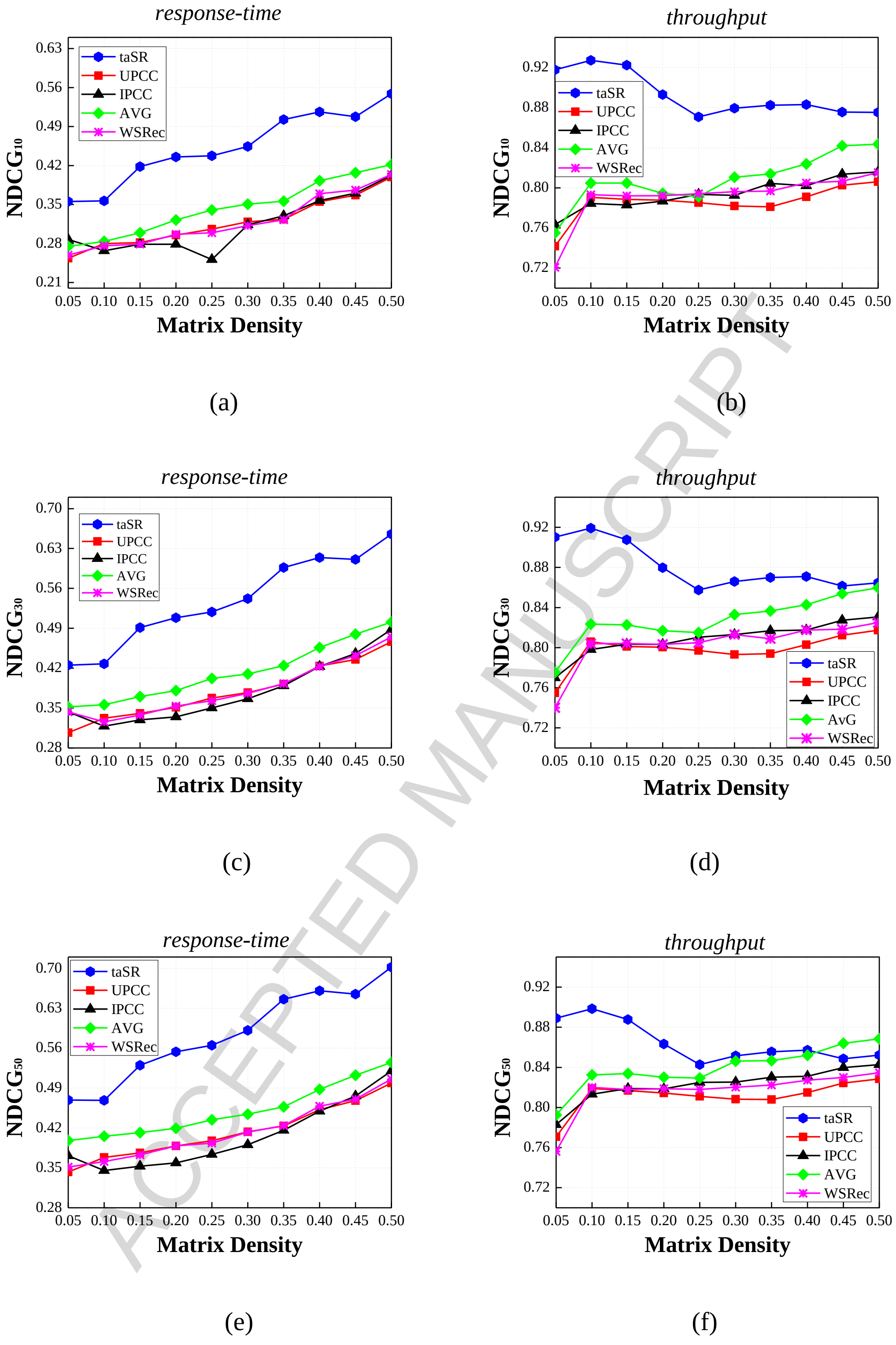
(c)

(d)

Fig. 5. Comparing MAE and RMSE from different approaches.

Fig. 5 summarizes the experimental results from different approaches under different matrix density settings. Figs. 5(a) and 5(b) present the comparison on response-time matrices while Figs. 5(c) and 5(d) present the comparison on throughput matrices. As shown in the figure, MAE and RMSE results decrease when the matrix density varies from 0.05 to 0.5. This is because dense matrices contain more useful information than sparse matrices. From the figure, we found that taSR outperforms all four other approaches on both response-time and throughput results, which demonstrates that, by better capturing the dynamic user similarity, we fill in the missing QoS values of the first 63 PITs more accurately. This leads to the better performance when adopting the ARIMA model in recommendation. Except for the value of RMSE in response-time, AVG and taSR generate better prediction than UPCC, IPCC approaches, indicating that time-aware approaches achieve better accuracy. Moreover, taSR outperforms AVG for the comprehensive consideration of the instantaneity.

To assess the accuracy of the service recommendation, we compared different recommendation approaches using the metrics NDCG_{10}, NDCG_{30} and NDCG_{50}. Fig.6 summarizes the ranking results based response-time and throughput, respectively. In general, the results from taSR are closer to 1.0 than those from other approaches. For response-time, the NDCG values increase when the density increases from 0.05 to 0.5, i.e., when there are more data. The NDCG values of response-time improve with the increasing numbers of the ranked cloud services, which indicates the robustness and stability of our approach. For throughput, taSR is clearly superior to other approaches when the density varies from 0.05 to 0.2 while the raw NDCG values remain stable with further increase of density. Both AVG and taSR generate better prediction than UPCC, IPCC and WSRec, indicating that time-aware approaches achieve better accuracy. Moreover, taSR outperforms AVG for the comprehensive consideration of the instantaneity except for NDCG_{50} with 0.45 and 0.5 density points. From the results, we concluded that taSR is effective for both similarity analysis and for the final service recommendation.



(a)

(b)

(c)

(d)

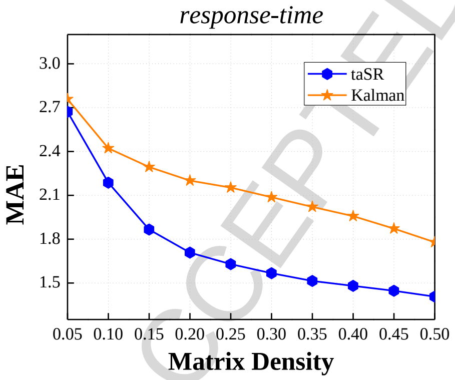
(e)

(f)

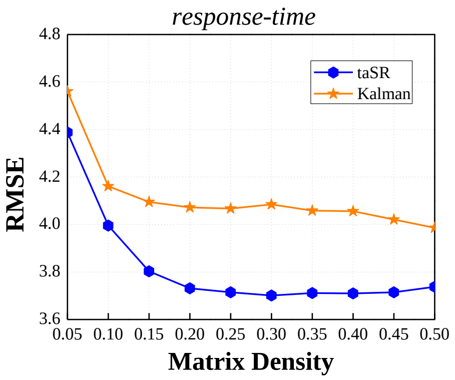
Fig. 6. Comparing NDCG$_{10}$, NDCG$_{30}$ and NDCG$_{50}$ from different approaches.

6.3.2. Comparison with the state-of-the-art time-aware approach

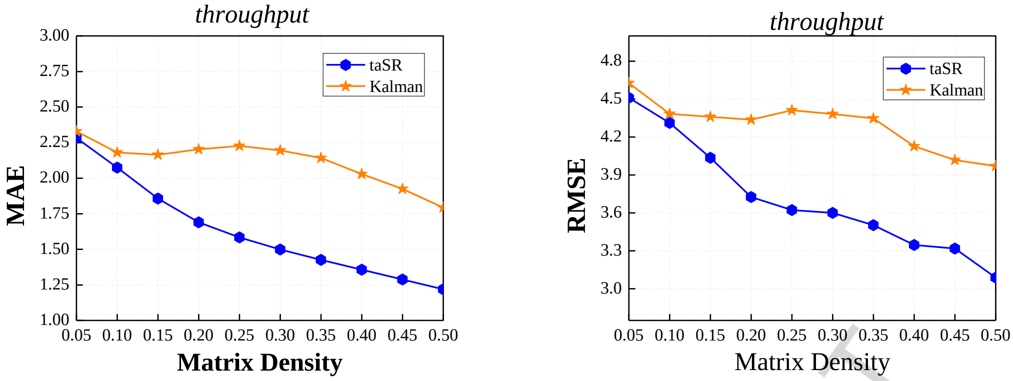
Next, we studied the prediction of QoS values at PIT t_{64}, i.e., for a future PIT, after adopting the ARIMA model. We compared taSR with the Kalman approach [17]. Hu et al. adopted Kalman to predict the QoS value for each of nine real-world web services with the full time-aware QoS sequences [17]. As a recursive approach, Kalman defines a state vector and a process noise vector according to the ARIMA model, and takes the new observation as a feedback. In this paper, we focused on the personalized QoS prediction, and compared the performance for taSR and Kalman. For the data preprocessing, we randomly removed a subset of data in the first 63 PITs to simulate the data sparsity in the real world, and filled in the missing data with traditional UPCC method for Kalman and with our similarity-enhanced CF method for taSR.



(a)



(b)



(c)

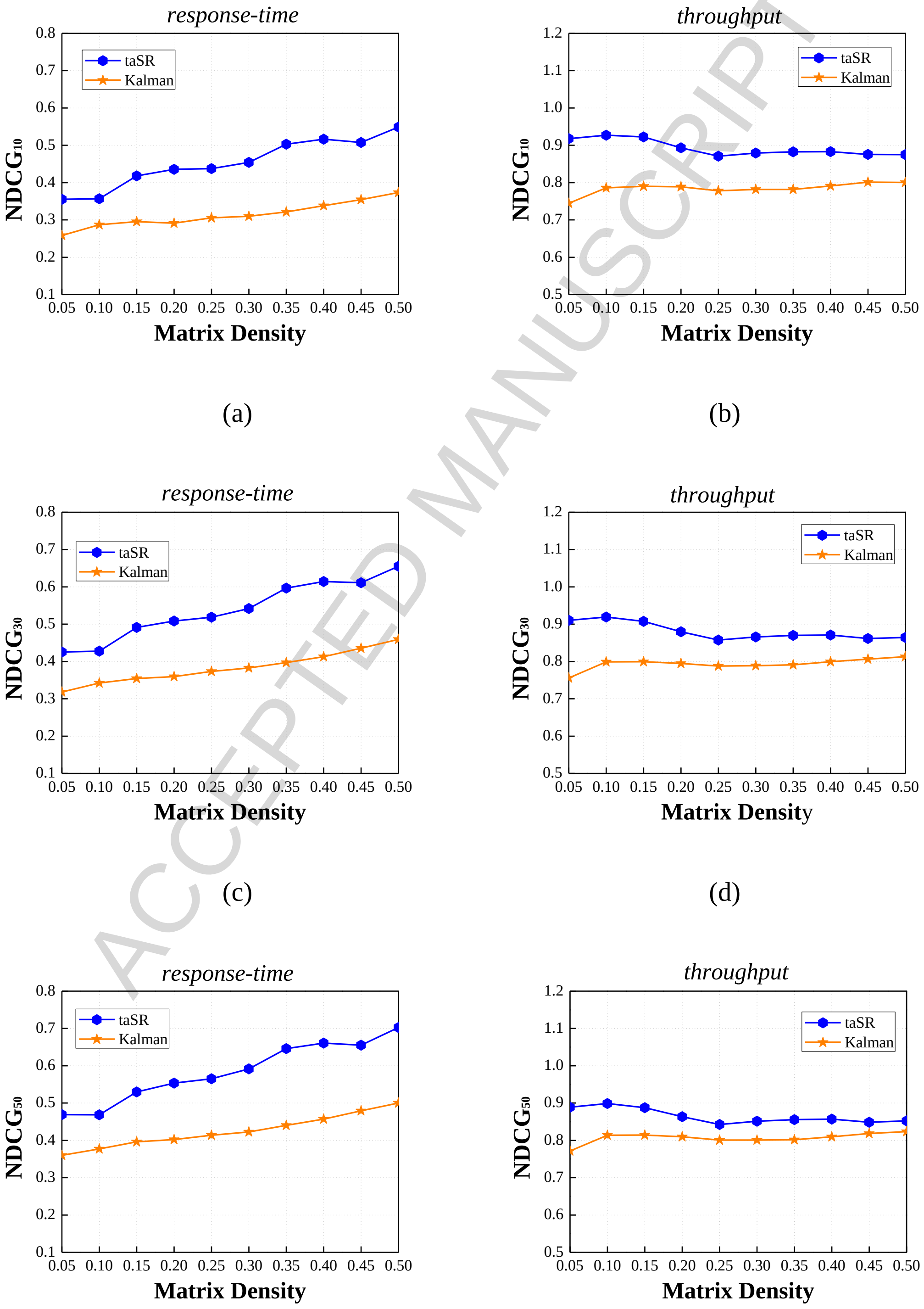
(d)

Fig. 7. Comparing MAE and RMSE from taSR and the Kalman approach.

Fig. 7 summarizes the MAE and RMSE results from taSR and the Kalman approaches. We used the the same dataset and matrix density. taSR and Kalman both predict the future QoS values in the future PIT(t_{64}). From the figure, taSR approach outperforms Kalman approach on both response-time and throughput. This is because the similarity estimation in the Kalman approach only used the QoS values of neighbor users at a specified PIT, even though the Kalman improves the ARIMA model for better stability.

We then compared the performance of taSR and the Kalman approach in making the optimal service recommendation. Fig. 8 summarizes the NDCG$_{10}$, NDCG$_{30}$ and NDCG$_{50}$ results from both approaches. From the figure, taSR is better than Kalman for all matrix densities and different QoS indicators. For response-time, taSR achieves better ranking with more data. For throughput, taSR generates good results, i.e., around 0.9, for NDCG$_{10}$, NDCG$_{30}$ and NDCG$_{50}$. tsSR achieves not only better stability but also better accuracy in throughput than those in response-time for the matrix density range that we evaluated.

Also from the figure, the Kalman approach is less sensitive to the matrix density as it uses the state transition matrix to predict the missing QoS value for the future. In the Kalman approach, the state transition matrix is predetermined so that it keeps unchanged across multiple iterations during prediction.



(a)

(b)

(c)

(d)

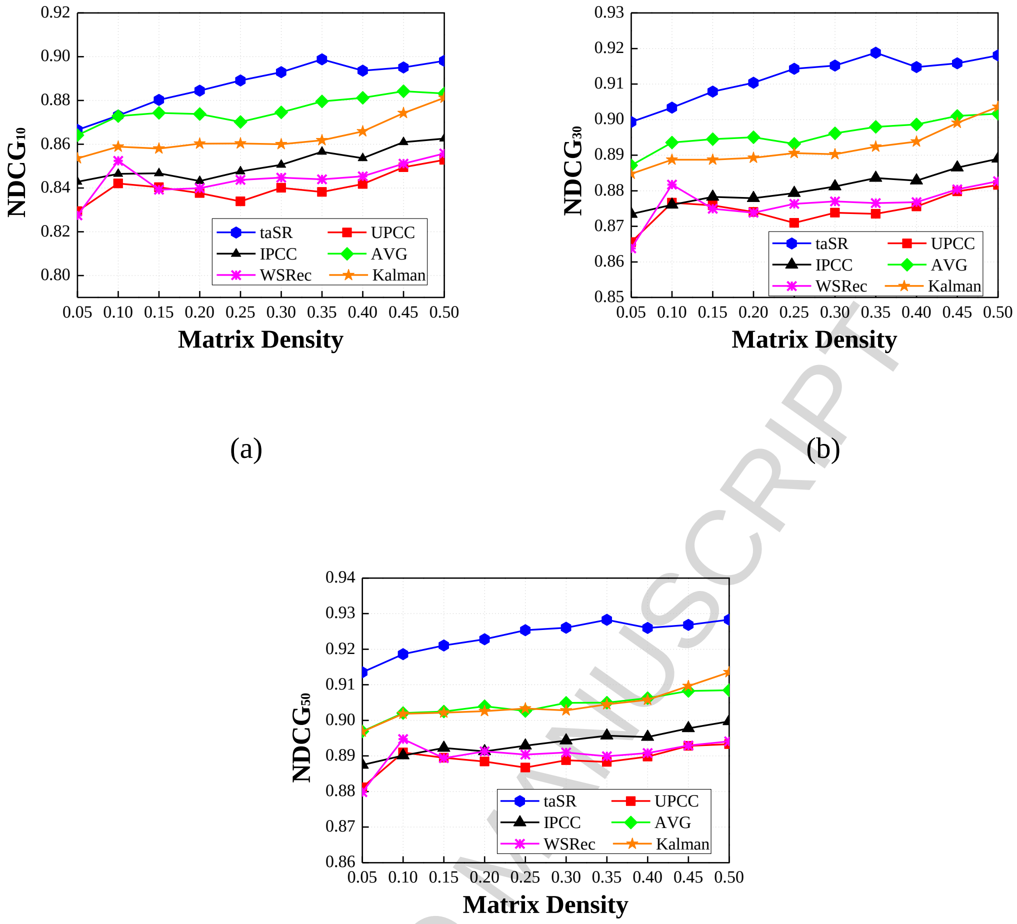
(e) (f)

Fig. 8. Comparing NDCG_{10}, NDCG_{30} and NDCG_{50} from taSR and the Kalman approach.

6.3.3 Comparison on cloud service selection

When adopting different indicators of QoS to evaluate the performance of cloud service, we formulate the selection as an MCDM problem and compute the aggregated QoS value as elaborated in Section 5.3.3.

Fig.9 compares the accuracy of cloud service selection based on the aggregated QoS values. We found that our approach ranks the cloud service more accurately for all NDCGs. The performance of taSR improves gradually when the density varies from 0.05 to 0.5, i.e., with more data information. When D=0.35, taSR produces the best ranking with the NDCG_{10}=0.899, NDCG_{30}=0.919 and NDCG_{50}=0.928. Based on the same dataset, filling the missing QoS values with separate UPCC in 63 PITs, Kalman is superior to UPCC, because Kalman improves the performance of ARIMA by correcting the prediction with new observation data. Furthermore, AVG produces a prediction rank better than UPCC, IPCC, WSRec and Kalman, as for the time-aware populating of data. Also from the figure, the accuracy of taSR increases with the increase of ranked services, demonstrating the stability and robustness of the proposed approach.



(a)

(b)

(c)

Fig.9. Comparing NDCG$_{10}$, NDCG$_{30}$ and NDCG$_{50}$ on cloud service selection.

6.3.4 Impact of k in recommendation

In our taSR approach, we choose top *k* similar users to the target user to filling the missing values in QoS matrices, which will influence the accuracy of prediction significantly. We tested *k*=2, 5, 10 and 30 to find the most appreciate *k* with the best performance. Tables 4, 5 and 6 show the performance of our approach with different *k* values. In general, our proposed approach achieves better performance with the *k*=5, except the NDCGs in throughput. TaSR has worse performance when *k*=2 or *k*=30. This is because it lacks sufficient information when *k*=2; and, when *k*=30, the aggregate effect of many less-similar users may distract the prediction of the choice of the target user. Therefore, we chose *k*=5 in the experiments to optimize the performance.

Table 4 summarizes the impact of *k* in response-time. In all density values, *k*=5 is clearly superior to others. In general, MAE and RMSE produce better prediction with smaller errors when *k*=5; $NDCG_{10}$, $NDCG_{30}$ and $NDCG_{50}$ perform better with the value closer to 1 when *k*=5. From the table, the best NDCGs for density D=0.05 appear when *k*=10 -- this is because, when density is very low, having more similar users helps to provide more information to improve prediction.

Table 5 summarizes the impact of *k* of throughput. The prediction of QoS values becomes more accurate when *k*=5 --- we observe smaller error of MAE and RMSE, except the RMSE in D=0.05. When *k*=10, the predicted rank of cloud services matches the real rank better than other *k* values for all density values we evaluated. Table 6 summarizes the impact of *k* in MCDM recommendation. We found that, for $NDCG_{10}$, $NDCG_{30}$ and $NDCG_{50}$, taSR achieves more precise recommendation when *k*=5 and the matrix density varies from 0.15 to 0.45. When D=0.05, taSR prefers to choose *k*=10 to recommend top 10 or 30 cloud services.

Table 4 impact of *k* in response-time

response-time	*k*	Matrix Density				
		0.05	0.15	0.25	0.35	0.45
MAE	2	2.673	2.312	1.823	1.680	1.465
	5	**2.668**	**1.858**	**1.635**	**1.522**	**1.448**
	10	2.694	2.130	1.787	1.593	1.480
	30	2.693	2.307	2.196	2.030	1.772
RMSE	2	4.461	3.953	3.815	3.820	3.788
	5	**4.386**	**3.798**	**3.728**	3.723	3.712
	10	4.407	3.973	3.787	**3.703**	**3.688**
	30	4.406	4.097	4.039	3.969	3.849
NDCG$_{10}$	2	0.215	0.375	0.348	0.439	0.465
	5	0.355	**0.418**	**0.438**	**0.503**	**0.508**
	10	**0.373**	0.381	0.418	0.447	0.503
	30	0.368	0.417	0.418	0.429	0.469
NDCG$_{30}$	2	0.397	0.423	0.462	0.498	0.524
	5	0.425	**0.491**	**0.519**	**0.597**	**0.616**
	10	**0.440**	0.454	0.499	0.537	0.611
	30	0.435	0.487	0.483	0.514	0.542
NDCG$_{50}$	2	0.457	0.483	0.527	0.539	0.586
	5	0.469	**0.530**	**0.565**	**0.646**	**0.660**
	10	**0.482**	0.498	0.540	0.584	0.655
	30	0.477	0.490	0.524	0.554	0.604

Table 5 impact of *k* in throughput

through-put	*k*	Matrix Density				
		0.05	0.15	0.25	0.35	0.45
MAE	2	2.268	1.867	1.743	1.672	1.406
	5	**2.257**	**1.843**	**1.589**	**1.426**	**1.287**
	10	2.264	1.989	1.722	1.544	1.385
	30	2.264	2.056	1.945	1.820	1.632
RMSE	2	4.543	4.269	3.819	3.604	3.407
	5	4.510	**3.931**	**3.698**	**3.452**	**3.323**
	10	**4.468**	4.214	3.822	3.578	3.433
	30	4.476	4.295	4.117	4.002	3.759
NDCG$_{10}$	2	0.897	0.921	0.901	0.873	0.869
	5	0.918	0.927	0.871	0.882	0.876
	10	**0.924**	**0.944**	**0.897**	**0.903**	**0.916**
	30	0.921	0.930	0.890	0.881	0.888
NDCG$_{30}$	2	0.889	0.894	0.843	0.867	0.856
	5	0.910	0.908	0.857	0.870	0.861
	10	**0.917**	**0.927**	**0.893**	**0.903**	**0.909**
	30	0.914	0.927	0.884	0.869	0.870
NDCG$_{50}$	2	0.873	0.857	0.845	0.832	0.841
	5	0.889	0.888	0.843	0.856	0.849
	10	**0.896**	**0.906**	**0.876**	**0.886**	**0.863**
	30	0.893	0.906	0.870	0.854	0.852

Table 6 impact of *k* in cloud service selection

selection	*k*	Matrix Density				
		0.05	0.15	0.25	0.35	0.45
$NDCG_{10}$	2	0.863	0.873	0.876	0.883	0.890
	5	0.867	**0.880**	**0.889**	**0.899**	**0.895**
	10	**0.868**	0.871	0.873	0.887	0.894
	30	0.867	0.871	0.882	0.875	0.889
$NDCG_{30}$	2	0.883	0.892	0.901	0.908	0.910
	5	0.899	**0.908**	**0.914**	**0.919**	**0.916**
	10	**0.900**	0.903	0.902	0.913	0.916
	30	0.900	0.902	0.909	0.903	0.906
$NDCG_{50}$	2	0.912	0.916	0.920	0.918	0.921
	5	**0.914**	**0.921**	**0.925**	**0.928**	**0.927**
	10	0.914	0.918	0.921	0.924	0.927
	30	0.914	0.917	0.923	0.917	0.924

8. Conclusion

While service recommendation and selection has become one of the most important tasks in cloud computing, it remains a major challenge to recommend the services that are most appropriate to match users' computing and service demands. In this paper, we propose taSR, a time-aware service recommendation approach that integrates similarity-enhanced CF based QoS prediction and time series analysis. taSR first enhanced similarity analysis by integrating user global similarity and invocation similarity. In particular, taSR adopts a time aware user similarity to describe the dynamic nature of user similarity. taSR then fills missing QoS values in the past PITs and the current PIT and adopts the ARIMA model to produce better recommendation for the future PIT.

With the fast advances of cloud computing paradigm, there exist a large amount of user-generated data in the cloud. It has become a major challenge for the research community to effectively exploit such data to improve service recommendation. In this paper, we made the effort to exploit the structured QoS data. In our future work, we will take advantage of unstructured data such as user comments, blogs, and discussion posts and adopt text-mining techniques [44] to further improve service recommendation and selection process.

Acknowledgements

The work is partially supported by the Ministry of Science and Technology of China under grant 2016YFC0503606, and fully supported by the National Natural Science Foundation of China (Nos. 71571058, 71131002, 71471055, 71501058, and 91546102), and a project funded by the China Postdoctoral Science Foundation grant no. 2015M570535.

References

[1] ZB Zheng, YL Zhang, MR Lyu, Distributed QoS Evaluation for Real-World Web Services, IEEE International Conference on Web Services. IEEE Computer Society, (2010) 83-90.

[2] D. Geebelen, K, Geebelen, E. Truyen, et al, QoS prediction for web service compositions using kernel-based quantile estimation with online adaptation of the constant offset, Information Sciences 268 (1) (2014) 397-424.

[3] K. Su, B. Xiao, B. Liu, et al, TAP: A personalized trust-aware QoS prediction approach for web service recommendation, Knowledge-Based Systems 115 (2017) 55-65.

[4] H. Wu, K. Yue, C.H. Hsu, et al, Deviation-based neighborhood model for context-aware QoS prediction of cloud and IoT services, Future Generation Computer Systems (2016), http://dx.doi.org/10.1016/j.future.2016.10.015.

[5] Y. Hu, Q. Peng, X. Hu, A time-aware and data sparsity tolerant approach for web service recommendation, IEEE International Conference on Web Services 2014, pp. 33-40.

[6] Z Ye, S.K Mistry, A Bouguettaya, et al, Long-term QoS-aware Cloud Service Composition using Multivariate Time Series Analysis, IEEE Transactions on Services Computing 9(3) (2016) 382-393.

[7] C.F. Tsai, C. Hung, Cluster ensembles in collaborative filtering recommendation, Applied Soft Computing 12 (4) (2012) 1417-1425.

[8] Y M Afify, I F Moawad, N L Badr, et al, Enhanced similarity measure for personalized cloud services recommendation, Concurrency & Computation Practice & Experience 29 (2017).

[9] C Yin, J Wang, J H Park, An Improved Recommendation Algorithm for Big Data Cloud Service based on the Trust in Sociology, Neurocomputing (2017).

[10] S Roy, R Bose, D Sarddar, A novel replica placement strategy using binary item-to-item collaborative filtering for efficient voronoi-based cloud-oriented content delivery network, Computer Engineering and Applications. IEEE, (2015) 603-608.

[11] M. Deshpande, G. Karypis, Item-based top-n recommendation algorithms, ACM Transaction on Information System 22 (1) (2004) 143–177.

[12] Z.B. Zheng, H. Ma, M.R. Lyu, et al, QoS-aware web service recommendation by collaborative filtering, IEEE Transactions on Services Computing 4 (2) (2011) 140–152.

[13] Y. Hu, Q. Peng, X. Hu, et al, Time aware and data sparsity tolerant web service recommendation based on improved collaborative filtering, IEEE Transactions on Services Computing 8 (5) (2015) 782-794.

[14] M. Silic, G. Delac, S. Srbljic, Prediction of atomic web services reliability for qos-aware recommendation, IEEE Transactions on Services Computing 8 (3) (2015) 425-438.

[15] Q Yu, Z Zheng, H Wang, Trace Norm Regularized Matrix Factorization for Service Recommendation, IEEE International Conference on Web Services (2013) 34-41.

[16] L Qi, X Xu, W Dou, et al, Time-Aware IoE Service Recommendation on Sparse Data, Mobile Information Systems (2016).

[17] Y. Hu, Q. Peng, X. Hu, et al, Web service recommendation based on time series forecasting and collaborative filtering, IEEE International Conference on Web Services 2015, pp. 233-240.

[18] X. Wang, J. Zhu, Z. Zheng, et al, A spatial-temporal QoS prediction approach for time-aware web service recommendation, ACM Transactions on the Web 10 (1) (2016)

[19] V W Chu, R K Wong, F Chen, et al, Web Service Recommendations Based on Time-Aware Bayesian Networks, IEEE International Congress on Big Data (2015) 359-366.

[20] E Borgonovo, V Cappelli, F Maccheroni, et al, Risk Analysis and Decision Theory: A Bridge, European Journal of Operational Research, 2017.

[21] D Ergu, G Kou, Questionnaire design improvement and missing item scores estimation for rapid and efficient decision making, Annals of Operations Research, 197(1) (2012) 5-23.

[22] J W Payne, J R Bettman, E J Johnson, Behavioral Decision Research: A Constructive Processing Perspective, Annual Review of Psychology, 43(1) (1992) 87-131.

[23] L Li, L Zheng, F Yang, et al, Modeling and broadening temporal user interest in personalized news recommendation, Expert Systems with Applications, 41(7) (2014) 3168-3177.

[24] Y Pan, D Wu, D L Olson. Online to offline (O2O) service recommendation method based on multi-dimensional similarity measurement, Decision Support Systems, 2017.

[25] S. Jimenez, F.A. Gonzalez, A. Gelbukh, Mathematical properties of Soft Cardinality: Enhancing Jaccard, Dice and cosine similarity measures with element-wise distance, Information Sciences 367 (2016) 373-389.

[26] S. Yabushita, A comparative analysis of the Tanimoto index and graph edit distance for measuring the topological similarity of trees, Applied Mathematics & Computation 259

(C) (2015) 242-250.

[27] J. Abreu, J.R. Rico-Juan, Characterization of contour regularities based on the Levenshtein edit distance, Pattern Recognition Letters 32 (10) (2011) 1421-1427.

[28] S.H. Ryu, B. Benatallah, Experts community memory for entity similarity functions recommendation, Information Sciences 379 (2017) 338-355.

[29] S Ding, C Y Xia, K L Zhou, et al. Decision support for personalized cloud service selection through multi-attribute trustworthiness evaluation, Plos One, 9(6) (2014).

[30] J C Chambers, S K Mullick, D D Smith, How to choose the right forecasting technique, Harvard Business Review, 49(4) (1971) 45-74.

[31] M Godse, U Bellur, R Sonar, Automating QoS Based Service Selection, IEEE International Conference on Web Services. IEEE, (2010) 534-541.

[32] A Amin, A Colman, L Grunske, An Approach to Forecasting QoS Attributes of Web Services Based on ARIMA and GARCH Models, International Conference on Web Services. IEEE, (2012) 74-81.

[33] H Ma, Z Hu, K Li, et al, Toward trustworthy cloud service selection: A time-aware approach using interval neutrosophic set, Journal of Parallel & Distributed Computing, 96 (2016) 75-94.

[34] O Kempthorne, The correlation between relatives on the supposition of mendelian inheritance,. American Journal of Human Genetics, 20(4) (1968) 402.

[35] C Wu, W Qiu, X Wang, et al, Time-Aware and Sparsity-Tolerant QoS Prediction Based on Collaborative Filtering, IEEE International Conference on Web Services. IEEE, (2016)

637-640.

[36] S Chen, Y Fan, W Tan, et al, Time-Aware Collaborative Poisson Factorization for Service Recommendation, IEEE International Conference on Web Services (2016) 196-203.

[37] X Wang, J Zhu, Z Zheng, et al, A Spatial-Temporal QoS Prediction Approach for Time-aware Web Service Recommendation, Acm Transactions on the Web 10(1) (2016) 7.

[38] Y Zhang, Z Zheng, M R Lyu. Real-Time Performance Prediction for Cloud Components, IEEE International Symposium on Object/component/service-Oriented Real-Time Distributed Computing Workshops, IEEE, (2012) 106-111.

[39] C Yu, L Huang. Time-Aware Collaborative Filtering for QoS-Based Service Recommendation, IEEE International Conference on Web Services. IEEE Computer Society (2014) 265-272.

[40] J. Abreu, J.R. Rico-Juan, Characterization of contour regularities based on the Levenshtein edit distance, Pattern Recognition Letters 32 (10) (2011) 1421-1427.

[41] K Rvelin, Kek, J Inen, Cumulated gain-based evaluation of IR techniques, Acm Transactions on Information Systems 20(4) (2002) 422-446.

[42] J S Breese, D Heckerman, C Kadie. Empirical analysis of predictive algorithms for collaborative filtering, Fourteenth Conference on Uncertainty in Artificial Intelligence, Morgan Kaufmann Publishers Inc (1998) 43-52.

[43] B Sarwar, G Karypis, J Konstan, et al. Item-based collaborative filtering recommendation algorithms, International Conference on World Wide Web. ACM, (2001) 285-295.

[44] L. Flory, K.M. Bryson, M. Thomas, A new web personalization decision-support artifact

for utility-sensitive customer review analysis, Decision Support Systems (2016), http://dx.doi.org/10.1016/j.dss.2016.11.003.

Shuai Ding is an associate professor at School of Management, Hefei University of Technology, China. He got his PhD from Hefei University of Technology.
Yeqing Li is a PhD candidate at School of Management, Hefei University of Technology, China.
Desheng Dash Wu is the Distinguished professor at University of Chinese Academy of Sciences, Professor of Stockholm University. His research interests focus on enterprise risk management in operations, performance evaluation in financial industry, and decision sciences. He has published more than 100 papers in refereed journals such as Production and Operations Management, Decision Support Systems, Decision Sciences, Risk Analysis, IEEE Transactions on Systems Man and Cybernetics etc. He is the editor of Springer book series titled "Computational Risk Management". He has served as associate editors/guest editors in such journals as IEEE Transactions on Systems Man and Cybernetics, Annals of Operations Research, Computers and Operations Research, International Journal of Production Economics, Omega etc. **He is a Senor Editor at Decision Support Systems.**

Youtao Zhang is an associate professor of Computer Science at the University of Pittsburgh. He received the Ph.D. degree in computer science from the University of Arizona, Tucson, AZ, USA, in 2002, and the B.S. and M.E. degrees from Nanjing University, Nanjing, China, in 1993 and 1996, respectively. Dr. Zhang has authored over 30 journal articles and more than 70 conference presentations in cloud computing, software engineering, memory systems and data intensive computing. Dr. Zhang was the recipient of the U.S. National Science Foundation Career Award in 2005, the Distinguished Paper Award of International Conference on Software Engineering 2003, and the Best Paper Award of International Symposium on Low Power Electronics and Design 2013. He is a member of ACM and IEEE.

Highlights

- We propose to integrate user global similarity and service invocation similarity in a novel time-aware similarity metric to address the instantaneity of QoS values.
- We propose to replenish missing QoS values for the past and current PITs through collaborative filtering (CF) and the predict QoS values in the future PIT with ARIMA model.
- Our experimental results showed that taSR achieves significantly improvements over existing approaches in various settings.