



# Temporal-Sparsity Aware Service Recommendation Method via Hybrid Collaborative Filtering Techniques

Shunmei Meng<sup>1,2</sup>(✉), Qianmu Li<sup>1</sup>(✉), Shiping Chen<sup>3</sup>, Shui Yu<sup>4</sup>,  
Lianyong Qi<sup>5</sup>, Wenmin Lin<sup>6</sup>, Xiaolong Xu<sup>7</sup>, and Wanchun Dou<sup>2</sup>

<sup>1</sup> Department of Computer Science and Engineering,  
Nanjing University of Science and Technology, Nanjing, China  
{mengshunmei, qianmu}@njust.edu.cn

<sup>2</sup> State Key Laboratory for Novel Software Technology, Nanjing University,  
Nanjing, China  
douwc@nju.edu.cn

<sup>3</sup> CSIRO Data61, Sydney, Australia  
shiping.chen@data61.csiro.au

<sup>4</sup> School of Software, University of Technology Sydney,  
Ultimo, NSW, Australia  
shui.yu@uts.edu.au

<sup>5</sup> School of Information Science and Engineering, Qufu Normal University,  
Jining, China  
lianyongqi@gmail.com

<sup>6</sup> Department of Computer Science, Hangzhou Dianzi University, Hangzhou,  
China  
linwenmin@hdu.edu.cn

<sup>7</sup> School of Computer and Software,  
Nanjing University of Information Science and Technology, Nanjing, China  
xlxu@nuist.edu.cn

**Abstract.** Temporal information has been proved to be an important factor to recommender systems. Both of user behaviors and QoS performance of services are time-sensitive, especially in dynamic cloud environment. Furthermore, due to the data sparsity problem, it is still difficult for existing recommendation methods to get the similarity relationships between services or users well. In view of these challenges, in this paper, we propose a temporal-sparsity aware service recommendation method based on hybrid collaborative filtering (CF) techniques. Specifically, temporal influence is considered into classical neighborhood-based CF model by distinguishing temporal QoS metrics from stable QoS metrics. To deal with the sparsity problem, a time-aware latent factor model based on a tensor decomposition model is applied to mine the temporal similarity between services. Finally, experiments are designed and conducted to validate the effectiveness of our proposal.

**Keywords:** Service recommendation · Temporal · Sparsity  
Collaborative filtering · CP decomposition

## 1 Introduction

With the rapid growth of public cloud offerings, service-oriented systems are becoming large-scale and complex. It becomes more difficult for users to find the appropriate services. Recommendation has been a hot research topic with the rapid growth of information. There have been many efforts done both in industry and academia on developing effective recommender systems [1, 2]. Collaborative filtering has been proved to be one of the most successful recommendation methods to deal with information overload in real world [3]. However, most of existing CF recommendation models provide recommendations based on rating models directly without considering context information, such as temporal influence. Actually, because of business competition and temporal dynamics of cloud environment, most cloud services are time-sensitive.

Recently, there have been some studies [4, 5] that investigate the importance of temporal influence in recommender systems, which mainly focus on the temporal dynamics on user preferences. Nevertheless, few works pay attention to temporal influence to Quality of Service (QoS). Compared with traditional internet services, QoS of cloud services is more sensitive to time due to the dynamics of cloud environment. The correlation among cloud services and user behaviors over long time may be weakened because of temporal dynamics. Thus it is important to consider the temporal impact on QoS of cloud services. Besides, the data sparsity problem is always a serious shortcoming to affect the performance of recommendation methods. Due to the sparsity problem, existing similarity models in collaborative methods fail to capture the similarity relationships between users or services effectively.

Based on these observations, in this paper, a temporal-sparsity aware service recommendation method via hybrid CF techniques is proposed. Firstly, temporal influence is considered into neighborhood-based CF recommendation model by distinguishing temporal QoS metrics from the stable QoS metrics. To address the data sparsity problem and mine the similarity relationships between services over time adaptively, a time-aware latent factor model based on tensor decomposition is presented. Finally, experiments based on real-world service dataset are designed and conducted to validate the effectiveness of our proposal.

## 2 Problem Formulation

In this section, some important concepts and definitions are presented. Firstly, to mining the temporal features of QoS, the QoS metrics of services are divided into two parts, i.e., stable QoS metrics and temporal QoS metrics:

**Stable QoS Metrics and Temporal QoS Metrics:** Stable QoS metrics are the metrics evolving at a rather slow speed, which are regular features (such as availability). Temporal QoS metrics are the metrics that have clear temporal features and present dynamic trends over time, which are dynamic features (such as price and duration).

To analyze the temporal influence to recommendation performance, the history time period of rating dataset is divided into  $K$  time slots, i.e.,  $\{T_1, T_2, \dots, T_K\}$ . The

temporal QoS metrics are distinguished from stable QoS metrics by measuring the fluctuation of the rating for each individual QoS metric over time. The fluctuation of QoS metric  $q_h$  is measured by the variance of QoS rating for  $q_h$  during the  $K$  time slots, which is defined in Eq. (1):

$$F(q_h) = \frac{1}{K \cdot |S|} \sum_{s=1}^{|S|} \sum_{k=1}^K (\overline{r}q_{hk}^s - \overline{r}q_h^s)^2 \quad (1)$$

Where  $|S|$  is the number of services in service set  $S$ ,  $\overline{r}q_{hk}^s$  is the average rating of metric  $q_h$  in time slot  $T_k$  of service  $s$ , and  $\overline{r}q_h^s$  is the overall average rating of  $q_h$  in total time period of service  $s$ .

A fluctuation threshold  $\delta_t$  is given to decide whether a QoS metric is stable or temporal. If  $F(q_h) > \delta_t$ , then  $q_h$  can be seen as a temporal QoS metric, otherwise  $q_h$  is considered as a stable metric. So the QoS metric vector  $\mathbf{Q} = [q_1, q_2, \dots, q_H]$  can be divided into two parts: stable QoS vector  $\mathbf{SQ} = [sq_1, sq_2, \dots, sq_a]$  and temporal QoS vector  $\mathbf{TQ} = [tq_1, tq_2, \dots, tq_b]$  ( $a + b = H$ ).

**Global Nearest Neighbors:** For a candidate service  $s$ , its global nearest neighbors are the services that have the most similar QoS ratings with service  $s$  at the whole time period of the history records.

**Temporal Nearest Neighbors:** The temporal nearest neighbors of service  $s$  are the services that have the most similar QoS ratings with service  $s$  at time slot  $T_k$  and its similar time slots, i.e.,  $T_{sim}(k)$ . (The definition of  $T_{sim}(k)$  is given in Sect. 3.1).

### 3 Temporal-Sparsity Aware Service Recommendation Method

In this paper, to make more appropriate recommendations from time-sensitive cloud services, a temporal-sparsity aware service recommendation method based on hybrid CF techniques is proposed. Our method can be designed as a three-phase process: (1) Time slot aggregation, (2) Similarity calculation and similarity prediction, (3) Rating prediction, which are described in detail in the following.

#### 3.1 Time Slot Aggregation

To make more accurate recommendations, we make predictions based on the QoS ratings at the target time slot. However, the related rating dataset of the target user at the target time slot maybe very spare. We provide an aggregation strategy to merge the similar time slots for the target time slot.

A temporal similar coefficient  $\varphi(T_i, T_j)$  is defined to measure the temporal closeness of the temporal QoS metrics between time slot  $T_i$  and  $T_j$ . As shown in Eq. (4),  $\varphi(T_i, T_j) \in [0, 1]$  is defined based on Pearson Correlation Coefficient (PCC). The larger  $\varphi(T_i, T_j)$  is, the closer the temporal features of candidate services between  $T_i$  and  $T_j$  is. Given a threshold  $\theta$  to determine whether two time slots are similar.

If  $\varphi(T_i, T_j) \geq \theta$ , then time slots  $T_i$  and  $T_j$  can be considered to be similar. The similar time slots of  $T_i$  is defined as  $T_{sim}(k)$  and  $T_k \subset T_{sim}(k)$ .

$$\varphi(T_i, T_j) = \frac{\sum_{s \in S(T_i) \cap S(T_j)} (\overline{\mathbf{RT}}_{is} - \overline{\mathbf{RT}}_i) \bullet (\overline{\mathbf{RT}}_{js} - \overline{\mathbf{RT}}_j)}{\sqrt{\sum_{s \in S(T_i) \cap S(T_j)} \|\overline{\mathbf{RT}}_{is} - \overline{\mathbf{RT}}_i\|^2} \cdot \sqrt{\sum_{s \in S(T_i) \cap S(T_j)} \|\overline{\mathbf{RT}}_{js} - \overline{\mathbf{RT}}_j\|^2}} \quad (2)$$

In Eq. (2),  $S(T_i) \cap S(T_j)$  is the set of coinvoled services by users at time slot  $T_i$  and  $T_j$ ,  $\overline{\mathbf{RT}}_{is}$  is the average temporal QoS-rating vector of service  $s$  at  $T_i$ ,  $\overline{\mathbf{RT}}_i$  is the average temporal QoS-rating vector of all candidate services in  $S(T_i) \cap S(T_j)$  at  $T_i$ .

### 3.2 Similarity Calculation and Similarity Prediction

As presented in Sect. 2, there are two kinds of nearest neighbors in our work, i.e., global nearest neighbors (denoted as  $S_{GNN}$ ) and temporal nearest neighbors denoted as  $S_{TNN}$ ). The global nearest neighbors of a service  $s$  can be decided by Eq. (3):

$$sim_{sv}^{GNN} = \frac{\sum_{u \in U(s) \cap U(v)} (\mathbf{RQ}_{us} - \overline{\mathbf{RQ}}_s) \bullet (\mathbf{RQ}_{uv} - \overline{\mathbf{RQ}}_v)}{\sqrt{\sum_{u \in U(s) \cap U(v)} \|\mathbf{RQ}_{us} - \overline{\mathbf{RQ}}_s\|^2} \times \|\mathbf{RQ}_{uv} - \overline{\mathbf{RQ}}_v\|^2}} \quad (3)$$

where  $U(s) \cap U(v)$  is the set of users that rated both service  $s$  and service  $v$  in the whole time period. Here, we give a preset similarity threshold  $\delta_{sim}$ , then the services that have similarity with service  $s$  no less than  $\delta_{sim}$  can be considered as global nearest neighbors of service  $s$ .

Similarly, the temporal nearest neighbors of service  $s$  can be determined by Eq. (4). The temporal nearest neighbors of service  $s$  at time slot  $T_k$  are the services that have similarity with service  $s$  no less than  $\delta_{sim}$  at time slot  $T_k$  and its similar time slots, i.e.,  $T_{sim}(k)$ .

$$sim_{sv}^{TNN}(T_k) = \frac{\sum_{u \in U_{sv}(T_k)} (\mathbf{RQ}_{us} - \overline{\mathbf{RQ}}_s) \bullet (\mathbf{RQ}_{uv} - \overline{\mathbf{RQ}}_v)}{\sqrt{\sum_{u \in U_{sv}(T_k)} \|\mathbf{RQ}_{us} - \overline{\mathbf{RQ}}_s\|^2} \times \|\mathbf{RQ}_{uv} - \overline{\mathbf{RQ}}_v\|^2}} \quad (4)$$

where  $U_{sv}(T_k) = \{u | u \in U(s) \cap U(v) \& t_{us} \in T_{sim}(k) \& t_{uv} \in T_{sim}(k)\}$ , which is the set of users that rated both service  $s$  and service  $v$  at  $T_{sim}(k)$ .

To solve sparsity problem further, a time-aware latent factor model based on CANDECOMP/PARAFAC (CP) decomposition [6] is applied to predict temporal similarity between services. The triadic relations among services, neighbors and time features are formulated as a three-dimensional similarity tensor  $\mathbf{Sim} \in \mathfrak{R}^{M \times M \times K}$ . The element in tensor  $\mathbf{Sim}$  is denoted as  $sim_{ijk}$ , which represents the temporal similarity of service  $i$  and service  $j$  at time slot  $T_k$ . The tensor  $\mathbf{Sim}$  is symmetric as  $sim_{ijk} = sim_{jik}$ . Then, based on the CP decomposition model, the tensor  $\mathbf{Sim} \in \mathfrak{R}^{M \times M \times K}$  can be decomposed into a sum of component rank-one tensors:

$$\mathbf{Sim} \approx \sum_{r=1}^R s_r \circ s_r \circ t_r \quad (5)$$

where  $R$  is actually the rank of tensor  $\mathbf{Sim}$ , which is defined as the smallest number of rank-one tensors.  $s_r$  and  $t_r$  represent the latent factor vectors associated with service and time, respectively.

Then the temporal similarity can be predicted by Eq. (6). The observed temporal similarity can be broken into two components: biases and service-neighbor-time interaction. The bias component contains the overall average similarity  $\mu$  and time bias  $b_{tk}$ . In similarity prediction, we didn't consider service bias.

$$\hat{sim}_{ijk} = \mu + b_{tk} + \sum_{r=1}^R s_{ir} \circ s_{jr} \circ t_{kr} \quad (6)$$

To learn the involved parameter  $b_{tk}$  and the involved vectors, i.e.,  $s_{ir}$ ,  $s_{jr}$  and  $t_{kr}$ , we minimize the regularized squared error function:

$$\min_{b, s, t} \sum_{(i,j,k) \in Train} \|sim_{ijk} - \hat{sim}_{ijk}\|^2 + \lambda W \quad (7)$$

where  $Train$  is the set of the  $(i, j, k)$  pairs for  $sim_{ijk}$ , which is known as the training set.  $sim_{ijk}$  is obtained by Eq. (4).  $W = b_{tk}^2 + \|s_{ir}\|^2 + \|s_{jr}\|^2 + \|t_{kr}\|^2$ , which is applied to regularize the learned parameters to avoid overfitting and the constant  $\lambda$  controls the extent of regularization. In this paper, we adopt stochastic gradient descent to solve Eq. (7) by looping through all similarity values in the training set.

### 3.3 Rating Prediction

The prediction of target user  $i$  on candidate service  $j$  at  $T_k$  (denoted as  $r_{ijk}$ ) is defined in Eq. (8). The prediction consists of two parts, i.e., prediction based on the global nearest neighbors and prediction based on the temporal nearest neighbors, which are combined by a weight coefficient  $\alpha$ . In our proposal,  $\alpha$  is set as  $a/H$ , and  $1 - \alpha = b/H$ .

$$r_{ijk} = \alpha \cdot \left( \bar{r}_j + \frac{\sum_{s \in S_{GNN}(j)} (r_{is} - \bar{r}_s) \cdot sim_{is}^{GNN}}{\sum_{s \in S_{GNN}(j)} |sim_{is}^{GNN}|} \right) + (1 - \alpha) \cdot \left( \bar{r}_j^k + \frac{\sum_{s' \in S_{TNN}(j, T_k)} (r_{is'} - \bar{r}_{s'}) \cdot sim_{is'}^{TNN}(T_k)}{\sum_{s' \in S_{TNN}(j, T_k)} |sim_{is'}^{TNN}(T_k)|} \right) \quad (8)$$

Where  $\bar{r}_j^k$  is the average rating of service  $j$  at the aggregated time slot  $T_{sim}(k)$ ,  $S_{GNN}(j)$  and  $S_{TNN}(j, T_k)$  respectively represent service  $j$ 's global and temporal nearest neighbor set where the services have been used by user  $i$ , and  $r_{is}$  represents the rating of user  $i$  on service  $s$ ,  $\bar{r}_s$  is the overall average rating of services in  $S_{GNN}(j)$ , and  $\bar{r}_{s'}$  is the overall average rating of services in  $S_{TNN}(j, T_k)$ .

## 4 Experiment

### 4.1 Experimental Setup

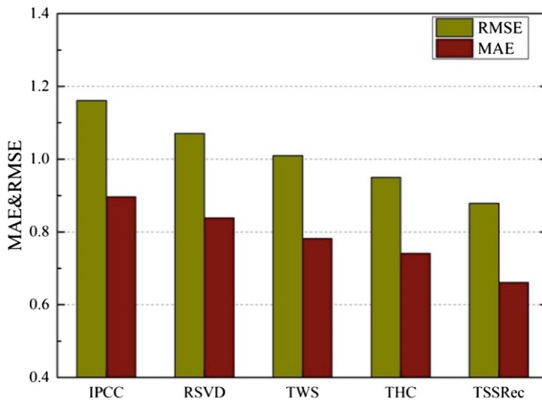
**Datasets:** We employ a real-world service dataset to simulate the history QoS ratings of services in the cloud market. The dataset is collected from a well-known travel review site ([www.tripadvisor.com](http://www.tripadvisor.com)). In our experiment, we use five-fold cross validation approach, and the dataset is split into 20% test data and 80% train data.

**Comparative Approaches:** To evaluate the effectiveness of our proposal, we compare our method with four other approaches: Item-based CF algorithm using PCC (IPCC) [7], regularized Singular Value Decomposition (RSVD) [8], a temporal QoS-aware web service recommendation method via tensor factorization (TWS) [9], and a time-aware hybrid collaborative recommendation method (THC) [10].

**Performance Metrics:** Four widely used metrics are applied to evaluate the statistical accuracy of recommendation approaches: mean absolute error (MAE), root-mean-square error (RMSE) [11], precision and recall [12].

### 4.2 Experimental Result

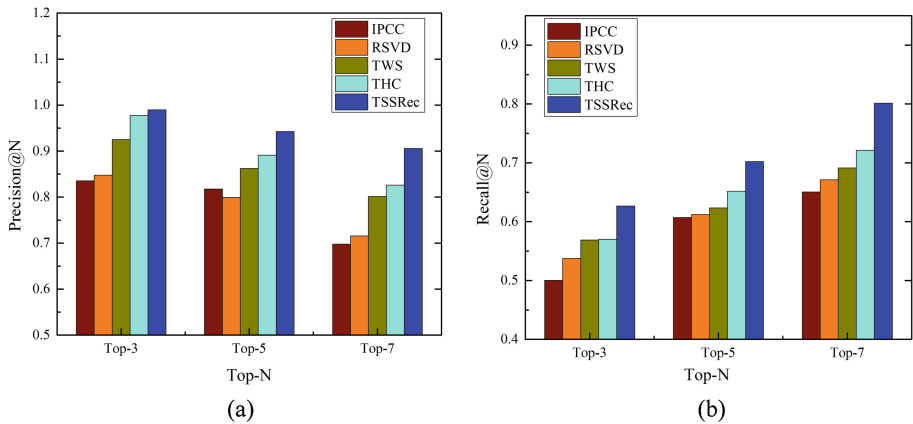
In our experiment, to evaluate the recommendation accuracy of our proposal, we compare our method (denoted as TSSRec) with four approaches. Figure 1 shows the best prediction performance (performance under the optimal parameter settings) of all methods in MAE and RMSE. We can see that both of MAE and RMSE of TSSRec are better than other four comparative approaches.



**Fig. 1.** Comparison of prediction performance in MAE and RMSE.

Figure 2 shows the performance of Top-N ( $N = 3, 5, 7$ ) recommendations. Figure 2(a) and (b) present the precision@N and recall@N performance of all methods,

respectively. It can be found that TSSRec also outperform other methods in Top-N ( $N = 3, 5, 7$ ) recommendation accuracy.



**Fig. 2.** Comparison of Top-N recommendation accuracy. (a) Comparison in precision@N. (b) Comparison in recall@N.

From the experimental results above, we can see that temporal influence is important to service recommendation in dynamic cloud environment. And our method achieves considerable improvement on recommendation accuracy compared to other comparative approaches.

## 5 Related Work

In the past, there has been some research work which integrates temporal influence into collaborative recommendation methods. Hu et al. [4] integrate temporal information into both similarity measurement and QoS prediction by considering the time gap between recommendation time and the occurring time of previous rating. The research [5] considers time information into recommendations based on probabilistic models. Wang et al. [13] adapt matrix factorization techniques to learn user-group affinity based on two different implicit engagement metrics. Recently, more and more influence factors, such as location influence, social influence, are considered into recommender systems. Lian et al. [14] propose a collaborative location recommendation framework to exploit the relations between users, activities and locations. The literature [15] focuses on the problem of joint modeling user check-in behaviors for real-time POI recommendation. Wang et al. [16] propose a spatial-temporal QoS prediction method where the temporal QoS prediction is formulated as a generic regression problem and a zero-mean Laplace prior distribution assumption is made on the residuals of QoS prediction.

## 6 Conclusion

In this paper, a temporal-sparsity aware service recommendation method based on hybrid CF techniques is proposed. Specifically, temporal influence is considered into classical neighborhood-based CF model by distinguishing temporal QoS metrics from stable QoS metrics. Accordingly, stable nearest neighbors and temporal nearest neighbors are defined. Then a time-aware latent factor model based on CP decomposition is integrated into neighborhood model to mine the temporal similarity relationships between services to address the data sparsity problem. Finally, experiments based on real-world service dataset are conducted to demonstrate the effectiveness of our proposal. In our future work, we will do further research in collaborative recommendation models based on multi-model integration and consider more context information in dynamic cloud environment.

**Acknowledgment.** This paper is partially supported by the National Science Youth Foundation of China under Grant No. 61702264, the Open Research Project of State Key Laboratory of Novel Software Technology (Nanjing University) under Grant No. KFKT2017B07.

## References

1. Yang, J., Liu, C., Teng, M., Chen, J., Xiong, H.: A unified view of social and temporal modeling for B2B marketing campaign recommendation. *IEEE Trans. Knowl. Data Eng.* **30**(5), 810–823 (2018)
2. Zhang, Y., et al.: Personalized quality centric service recommendation. In: Maximilien, M., Vallecillo, A., Wang, J., Oriol, M. (eds.) *ICSOC 2017. LNCS*, vol. 10601, pp. 528–544. Springer, Cham (2017). [https://doi.org/10.1007/978-3-319-69035-3\\_39](https://doi.org/10.1007/978-3-319-69035-3_39)
3. Meng, S., Dou, W., Zhang, X., Chen, J.: KASR: a keyword-aware service recommendation method on mapreduce for big data applications. *IEEE Trans. Parallel Distrib. Syst.* **25**(12), 3221–3231 (2014)
4. Hu, Y., Peng, Q., Hu, X., Yang, R.: Time aware and data sparsity tolerant web service recommendation based on improved collaborative filtering. *IEEE Trans. Serv. Comput.* **8**(5), 782–794 (2015)
5. Zhang, J.D., Chow, C.Y.: TICRec: a probabilistic framework to utilize temporal influence correlations for time-aware location recommendations. *IEEE Trans. Serv. Comput.* **9**(4), 633–646 (2016)
6. Kolda, T.G., Bader, B.W.: Tensor decompositions and applications. *SIAM Rev.* **51**(3), 455–500 (2009)
7. Linden, G., Smith, B., York, J.: Amazon. com recommendations: item-to-item collaborative filtering. *IEEE Internet Comput.* **7**(1), 76–80 (2003)
8. Koren, Y., Bell, R., Volinsky, C.: Matrix factorization techniques for recommender systems. *Computer* **42**(8), 42–49 (2009)
9. Zhang, W., Sun, H., Liu, X., Guo, X.: Temporal QoS-aware web service recommendation via non-negative tensor factorization. In: *Proceedings of the 23rd International Conference on World Wide Web*, pp. 585–596. ACM (2014)
10. Meng, S., et al.: A temporal-aware hybrid collaborative recommendation method for cloud service. In: *2016 IEEE International Conference of Web Services (ICWS)*, pp. 252–259. IEEE (2016)



11. Chai, T., Draxler, R.R.: Root mean square error (RMSE) or mean absolute error (MAE)?—arguments against avoiding RMSE in the literature. *Geosci. Model. Dev.* **7**(3), 1247–1250 (2014)
12. Carullo, G., Castiglione, A., De Santis, A., Palmieri, F.: A triadic closure and homophily-based recommendation system for online social networks. *World Wide Web* **18**(6), 1579–1601 (2015)
13. Wang, X., Donaldson, R., Nell, C., Gorniak, P., Ester, M., Bu, J.: Recommending groups to users using user-group engagement and time-dependent matrix factorization. In: AAAI, pp. 1331–1337 (2016)
14. Lian, D., et al.: Scalable content-aware collaborative filtering for location recommendation. *IEEE Trans. Knowl. Data Eng.* **30**(6), 1122–1135 (2018)
15. Yin, H., Cui, B., Zhou, X., Wang, W., Huang, Z., Sadiq, S.: Joint modeling of user check-in behaviors for real-time point-of-interest recommendation. *ACM Trans. Inf. Syst. (TOIS)* **35**(2), 11 (2016)
16. Wang, X., Zhu, J., Zheng, Z., Song, W., Shen, Y., Lyu, M.R.: A spatial-temporal QoS prediction approach for time-aware web service recommendation. *ACM Trans. Web (TWEB)* **10**(1), 7 (2016)