Neurocomputing 467 (2022) 454–464

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Real-time POI recommendation via modeling long- and short-term user preferences

Xin Liu [a], Yongjian Yang [a], Yuanbo Xu [a], Funing Yang [a,*], Qiuyang Huang [a], Hong Wang [b]

[a] College of Computer Science and Technology, Jilin University, Changchun 130012, China
[b] Army Academy of Armored Forces, Changchun 130000, China

A R T I C L E I N F O

Article history:
Received 3 January 2021
Revised 12 August 2021
Accepted 25 September 2021
Available online 4 October 2021
Communicated by Zidong Wang

Keywords:
Real-time POI recommendation
Periodic trends
Public preferences
Category filter

A B S T R A C T

Recently, Next Point-of-Interest (POI) Recommendation which proposes users for their next visiting locations, has gained increasing attention. A timely and accurate next POI recommendation can improve users' efficient experiences. However, most existing methods typically focus on the sequential influence, but neglect the user's real-time preference changing over time. In some scenarios, users may need a real-time POI recommendation, for example, when using Take-away Applications, users need recommending the appropriate restaurants at the specific moment. Hence, how to mine users' patterns of life and their current preferences becomes an essential issue for the real-time POI recommendation. To address the issues above, we propose a real-time preference mining model (RTPM) which is based on LSTM to recommend the next POI with time restrictions. Specifically, RTPM mines users' real-time preferences from long-term and short-term preferences in a uniform framework. For the long-term preferences, we mine the periodic trends of users' behaviors between weeks to better reflect users' patterns of life. While for the short-term preferences, trainable time transition vectors which represent the public preferences in corresponding time slots, are introduced to model users' current time preferences influenced by the public. At the stage of recommendation, we design a category filter to filter out the POIs whose categories are unpopular in corresponding time slots to reduce the search space and make recommendation fit current time slot better. Note that RTPM does not utilize users' attributes and their current locations for recommendation, which makes great contributions to users' privacy protection. Extensive experiments on two real-world datasets demonstrate that RTPM outperforms the state-of-the-art models on Recall and NDCG.

1. Introduction

In recent years, Location-Based Social Networks (LBSNs), such as Foursquare,[1] Yelp[2] and GoWalla,[3] have gained rapid development and produce abundant check-in data. These data offer a great opportunity to explore users' mobility patterns and preferences for POI recommendation. Besides, to better apply POI recommendation to real-life scenarios, next POI recommendation, a natural extension to general POI recommendation, has been attracting more attention from researchers [1–3]. It leverages users' historical check-in sequences to recommend the most probable POIs which users will visit next, which benefits both users and service providers.

While next POI recommendation has achieved great success, some limitations are still existing. 1) Although most next POI recommendation models [4,5] take dynamic preferences into consideration to suggest next POIs, they ignore users' real-time preferences with time restrictions [6]. For example, in Fig. 1, a user goes to the hotel between 9:00 and 10:00 and then visits the library and supermarket in sequence. If this user needs to be recommended the next POI between 14:00 and 15:00, these models may perform weakly due to lack of considering the preference in this specific time slot. 2) Many existing next POI recommendation models just try to mine users' mobility patterns from specific POI transition regularities [7–9]. Nevertheless, they ignore the influence of the POI category which reflects the purpose of the POI. Some POI categories may be unpopular in some specific time slots according to people's patterns of life. To illustrate that clearly, we specially calculate the percentage of the number of check-ins in each time slot with respect to the two POI categories, museum and nightlife spot, from the dataset TKY (detailed in Section 6.1) as examples and present the results in Fig. 2. Nobody approaches

* Corresponding author.
E-mail address: yfn@jlu.edu.cn (F. Yang).

[1] https://foursquare.com/.
[2] https://www.yelp.com/.
[3] https://blog.gowalla.com/.

https://doi.org/10.1016/j.neucom.2021.09.056
0925-2312/

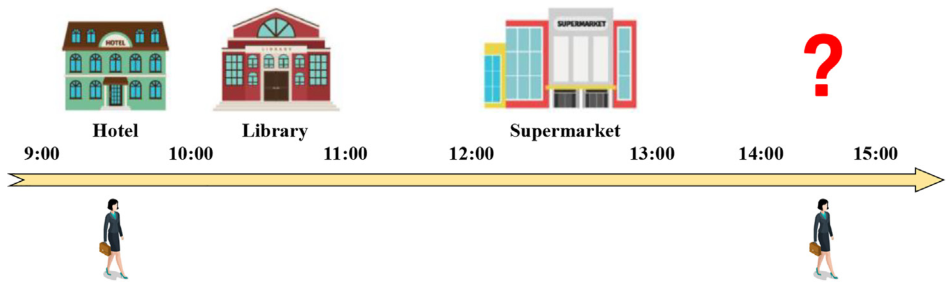


Fig. 1. The scenario of real-time POI recommendation.

to the museum from the time slot 15 to the time slot 21 as shown in Fig. 2a and few people go to the nightlife spot from the time slot 1 to the time slot 7 as shown in Fig. 2b. Hence, it is meaningless to recommend the POIs belonging to the two categories in these time slots. 3) Some next POI recommendation models [10] leverage users' personal attributes, social relationships or their current locations to make a recommendation, which will expose users' privacy.

To overcome the limitations mentioned above, we propose a model named RTPM for real-time POI recommendation. It utilizes users' check-in records and current time (without current locations) to generate POI recommendation lists where POIs are considered users are interested in at present. First, we mine the periodic trends of users' patterns of life between weeks to model users' long-term preferences. Then, for the short-term preferences, in addition to considering the sequential influence, we consider the preferences with time restrictions, which is essential for the real-time recommendation. Specifically, time transition vectors are introduced to depict the users' current time preferences influenced by the public. In addition, we filter out the POIs whose categories users have few interest in at the moment according to people's living habits before recommendation. In this way, the categories can be made better use of to make the recommendation in line with users' patterns of life and help to improve recommendation accuracy effectively. It's worth noting that users' privacy gains protection to some extent in our model, due to no use of users' personal attributes and their current locations.

The main contributions of this paper are summarized as follows:

- We propose a real-time preference mining model named RTPM to formulate both users' long-term and short-term preferences with time restrictions for the real-time POI recommendation.
- We introduce explicit weighting functions for each day to better describe the periodic trends of users' behaviors between weeks for users' long-term preferences. Trainable time transition vectors are deployed to depict the influence of the public preferences on users' short-term preferences in different time slots. A category filter is used to filter out the POIs whose categories is unpopular in current time slot, which makes the final recommendation lists fit users' living habits in current time slot better.
- The model makes POI recommendation only leveraging users' behavior patterns and making no use of users' personal attributes and their current locations which makes for users' privacy protection.

The rest of the paper is organized as follows: in Section 2, we review the related works. Basic definitions and the problem definitions are given in Section 3. Section 4 describes the overall framework and the components of our model. Then, we introduce our model detailedly in Section 5. In Section 6, we conduct experiments to evaluate our proposed model. Finally, we conclude the paper in Section 7.

2. Related works

2.1. POI recommendation

POI recommendation as a hot academic issue has attracted extensive attention from researchers in recent years. Early POI recommendation mainly adopts Collaborative Filtering (CF) based methods to explore users' preferences. Ye et al. [11] incorporate social and geographical influence with user-based CF for recom-

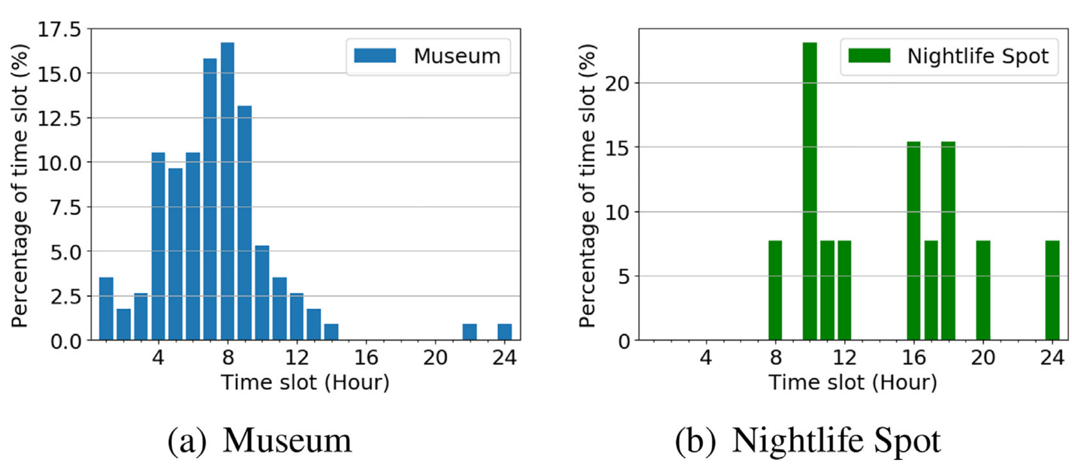


Fig. 2. Temporal distribution characteristics of check-ins for different POI categories: (a) Museum; (b) Nightlife Spot.

mendation. Aware of the importance of temporal influence, Yuan et al. [12] try to mine the temporal behaviors of users from their historical check-in records and then integrate that as well as spatial behaviors with the collaborative recommendation model. Aliannejadi et al. [13] propose a two-phase collaborative ranking algorithm that incorporates the geographical influence of POIs and is regularized based on the variance of POIs popularity and users' activities over time. Matrix Factorization (MF) technique is also exploited widely in POI recommendation. Lian et al. [14] propose augmenting users' and POIs' latent factors with activity area vectors of users and influence area vectors of POIs to capture the spatial clustering phenomenon in human mobility behaviors, and then incorporate the spatial clustering phenomenon into MF for POI recommendation. Liu et al. [15] employ MF to predict users' preference transitions over location categories. Considering the relationship hidden among the content features, Zhang et al. [16] present an optimization model for extracting the relationship hidden in content features by considering user preferences. With the development of neural networks (NN), massive researches on POI recommendation based on NN occur. He et al. [17] present a NN architecture to model latent features of users and items and a general framework for CF based on NN. Liu et al. [18] incorporate geographic features with generative adversarial networks to make a recommendation. However, conventional POI recommendation methods don't consider the sequential effect, which limits their performance in the real world scenarios.

2.2. Next POI recommendation

Next POI recommendation, different from conventional POI recommendation, fixes more attention on the sequence dependency. Early researches exploit the Markov chain to recommend the next probable POI. Cheng et al. [19] propose a novel matrix factorization method to incorporate personalized Markov chain with localized region constraint for recommendation. Chen et al. [20] devise a next location predictor with Markov model, considering both individual and collective movement patterns and suited to different time periods. In recent years, deep learning achieves great success in many fields, e.g., Natural Language Processing (NLP) and Computer Vision (CV). In this context, a large number of recommendation methods based on deep learning appear. Chang et al. [21] propose a Word2Vec based POI embedding model utilizing users' check-in sequence and text context about POIs for successive POI recommendation. Zheng et al. [22] propose an attention-based dynamic preference model for next POI recommendation. Liu et al. [23] propose a Recurrent Neural Networks (RNN) based model incorporating time-specific transition matrices and distance-specific transition matrices to model time intervals and geographical distances, respectively. Long Short-Term Memory (LSTM), as a variant of RNN, is also widely applied to recommend the next POI. Sun et al. [24] devise a model based on LSTM, consisting of a nonlocal network for long-term preference modeling and a geo-dilated RNN for short-term preference learning. Yu et al. [25] propose a category-aware deep model that incorporates POI categories and geographical influence to reduce search space to overcome data sparsity.

However, when facing the scenario of real-time POI recommendation, some next POI recommendation models may perform weakly due to the lack of attention on users' real-time preferences. They don't model the preferences changing over time and ignore users' behavior habits in different time slots. By contrast, our RTPM model not only focuses on users' sequential preferences, but also takes users' real-time preferences with time restrictions into consideration, which is useful for understanding users' current intentions. Besides, we exploit the statistical regularities for user's check-ins on various POI categories to filter out the unpopular POI categories in corresponding time slots. This way can better describe users' patterns of life and further enhance the effect of recommendation.

3. Preliminaries

Let $U = \{u_1, u_2, \ldots, u_{|U|}\}$be a set of LBSN users, $L = \{l_1, l_2, \ldots, l_{|L|}\}$be a set of POIs, and $C = \{c_1, c_2, \ldots, c_{|C|}\}$be a set of POI categories. Each POI is associated with its coordinates $(longitude, latitude)$, i.e., (lon_l, lat_l), and belongs to one of the POI categories in C. Each user $u \in U$ has a trajectory sequence represented by $S = \{S_1, S_2, \ldots, S_n\}$, where n is the index of the current day's trajectory. Each trajectory $S_m = \{l_1, l_2, \ldots, l_{|S_m|}\}$denotes a sequence of POIs visited consecutively by the user in the m-th day, where $S_m \in S$. We specially represent the current day's check-in POI sequence as $S_n = \left\{l^1, l^2, \ldots, l^{p-1}\right\}$, where p is the index of the present time and $p-1$ is the index of the latest check-in time.

The real-time POI recommendation is defined thus: given a user's historical trajectory sequence $\{S_1, S_2, \ldots, S_{n-1}\}$, current day's trajectory $S_n = \left\{l^1, l^2, \ldots, l^{p-1}\right\}$and present time t^p, the real-time POI recommendation provides user u with top-K POIs where user u would like to go at the present time t^p.

4. Framework

Our RTPM model is presented in Fig. 3. It mainly consists of four components, i.e., the long-term preference, the short-term preference, the probability calculation and the recommendation module. The long-term preference section intends to mine the periodic trends contained in the past days. The short-term preference section combines user's own sequential influence with the public preferences at the present time. And the last two parts calculate the probabilities of all POIs and leverage the category filter to generate the final recommendation lists which better match people's living habits.

Long-term preference: given a user's trajectory sequence, this part first inputs these POI embedding vectors into the LSTM to obtain the preference vectors related to each check-in. Then, it weighs each preference vector above according to the time interval and geographical distance between each historical check-in and the latest check-in. After that, we further integrate them to make up the everyday preferences. Finally, everyday preferences are given weights by a specially designed function of time and distance to describe the periodic trends of daily life. And then, the long-term preference are formulated by everyday preferences.

Short-term preference: given a user's preference vector related to the latest check-in, the time of his latest check-in and the present time, the short-term preference is formulated by the user's own sequential influence and the public influence. The sequential influence is expressed in the preference vector related to the latest check-in. The preference influenced by the public is calculated by introducing two time transition vectors, one for the latest check-in time and another for the current time, which reflect the public preferences in corresponding time slots. At last, the short-term preference is modeled by integrating the public preference and current day's check-in sequence influence.

Probability calculation and recommendation: in the first two parts, users' long-term and short-term preferences have been modeled. We concatenate the long-term preference vector and the short-term preference vector to form a new one and utilize it to calculate the probabilities of each POI. Before generating the final recommendation list, we filter out the POI categories which are unpopular in current time and remain the POIs belonging to the

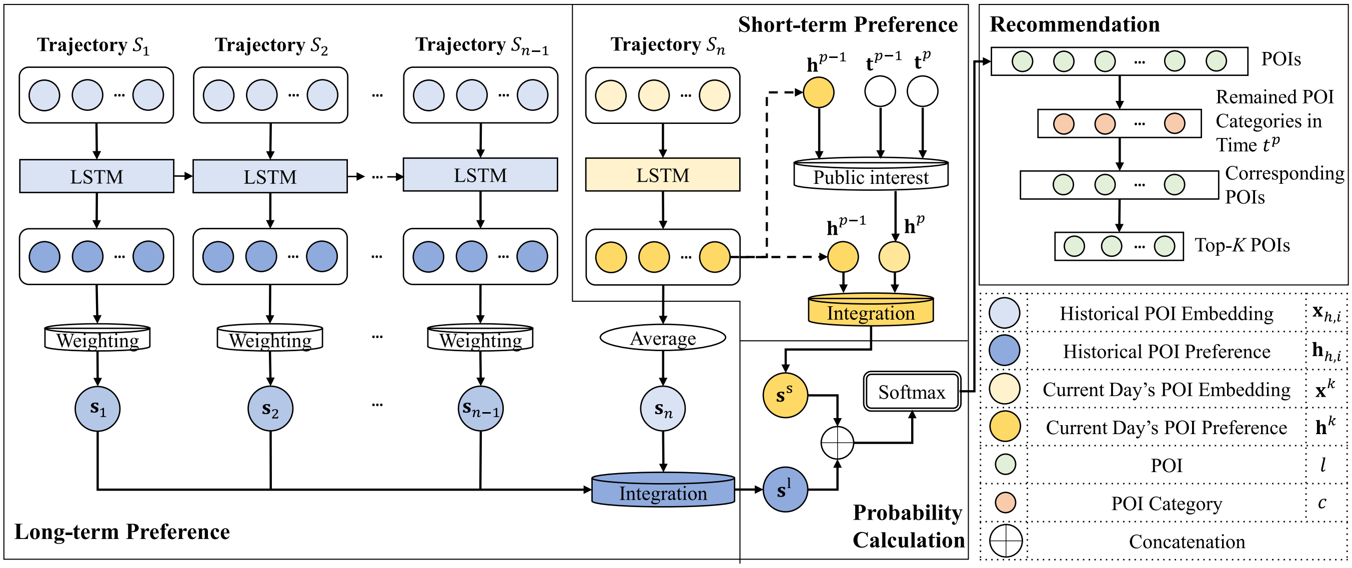


Fig. 3. The overview of our RTPM model.

rest categories. Finally, the model generates the recommendation list from the rest of POIs according to the probabilities.

5. Methodology

In this section, we introduce our RTPM model in detail. The four components, i.e., the long-term preference, the short-term preference, the probability calculation and the recommendation module will be introduced in order.

5.1. Long-term preference

This section is applied to mine users' long-term preferences from both spatial and temporal aspects. The main idea is to mine the relationship between users' historical preferences and the current preferences according to the spatial and temporal similarities. The more similarities the historical behaviors shared with the present's, the more effect the historical preferences may have on the present's. We measure the spatial and temporal similarities by the distance of time and space.

First of all, given a user u's historical trajectory sequence $\{S_1, S_2, \ldots, S_{n-1}\}$, we input each check-in POI vector into the LSTM to obtain the preference vector $\mathbf{h}_{h,i}$ related to each check-in:

$$\mathbf{h}_{h,i} = LSTM(\mathbf{x}_{h,i}, \mathbf{h}_{h,i-1}), \quad i \in \{1, 2, \ldots, |S_h|\}, \quad h \in \{1, 2, \ldots, n-1\}, \tag{1}$$

where $\mathbf{h}_{h,i}$ is the hidden state of LSTM which is regarded as the user u's preference related to $\mathbf{x}_{h,i}$. $\mathbf{x}_{h,i} \in \mathbb{R}^{d \times 1}$ is the d-dimensional embedding vector of i-th POI l_i in the h-th trajectory S_h, which is randomly initialized and trained in the network. Then, the preferences $\{\mathbf{h}_{h,1}, \mathbf{h}_{h,2}, \ldots, \mathbf{h}_{h,|S_h|}\}$related to historical check-ins are formulated.

Next, we integrate each check-in preference to formulate each historical day's preference by considering spatial and temporal factors. The main idea is that the shorter spatio-temporal distance between the historical check-in and the latest check-in is, the more effect the corresponding check-in preference may have on the present's and it should be given a large weight. Specially, we divide one week into 48 time slots (24 slots for hours on weekdays and 24 slots for hours on weekends) and each check-in time is represented by the time slot. Then the h-th day's preference $\mathbf{s}_h$ is defined as follows:

$$\mathbf{s}_h = \sum_{i=1}^{|S_h|} (\alpha \omega^{t}_{h,i} \mathbf{h}_{h,i} + (1-\alpha)\omega^{s}_{h,i} \mathbf{h}_{h,i}), h \in \{1, 2, \ldots, n-1\} \tag{2}$$

$$\omega^{t}_{h,i} = \frac{\exp(-\mathrm{abs}(t_{h,i} - t^{p-1}))}{\sum_{j=1}^{|S_h|} \exp(-\mathrm{abs}(t_{h,j} - t^{p-1}))} \tag{3}$$

$$\omega^{s}_{h,i} = \frac{\exp(-d_{l_{h,i}, l^{p-1}})}{\sum_{j=1}^{|S_h|} \exp(-d_{l_{h,j}, l^{p-1}})} \tag{4}$$

$$d_{l_{h,i}, l^{p-1}} = \sqrt{(lon_{l_{h,i}} - lon_{l^{p-1}})^2 + (lat_{l_{h,i}} - lat_{l^{p-1}})^2} \tag{5}$$

where $\omega^{t}_{h,i}$ and $\omega^{s}_{h,i}$ are the time weight and distance weight of the preference $\mathbf{h}_{h,i}$ which model the importance of the preference $\mathbf{h}_{h,i}$ from temporal and spatial distance, respectively. The time weight $\omega^{t}_{h,i}$ is calculated by Eq. (3), where $t_{h,i}$ is the time slot of the i-th check-in in the h-th day and t^{p-1} is the time slot of the latest check-in. The distance weight $\omega^{s}_{h,i}$ is calculated by Eq. (4), where $d_{l_{h,i}, l^{p-1}}$, the distance between the i-th check-in location in the h-th historical day $l_{h,i}$ and the latest check-in location l^{p-1}, can be calculated by the Euclidean distance as Eq. (5). $\alpha \in (0, 1)$ is a hyper-parameter to trade off the importance of temporal and spatial factors. We can find that the check-in which has shorter time interval with the latest check-in time slot and the shorter distance with the latest check-in location, takes more effect on modeling the h-th day's preference $\mathbf{s}_h$. Then, we obtain every historical day's preference $\{\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_{n-1}\}$.

For current day's preference, we use a separate LSTM to model the preference $\mathbf{h}^k$ related to each current day's check-in:

$$\mathbf{h}^k = LSTM(\mathbf{x}^k, \mathbf{h}^{k-1}), k \in \{1, 2, \ldots, p-1\} \tag{6}$$

where $\mathbf{h}^k$ is the hidden state of LSTM which is regarded as the preference related to $\mathbf{x}^k$ and $\mathbf{x}^k \in \mathbb{R}^{d \times 1}$ is the d-dimensional embedding vector of the k-th POI in current day's trajectory S_n. In this way, we obtain the preference $\{\mathbf{h}^1, \mathbf{h}^2, \ldots, \mathbf{h}^{p-1}\}$related to current day's check-ins. Further, the current day's preference is defined as follows:

$$\mathbf{s}_n = \frac{1}{|S_n|}\sum_{k=1}^{|S_n|}\mathbf{h}^k \tag{7}$$

where we employ the average of the preference vector of all current day's check-ins $\mathbf{h}^k$ is out of considering each current day's check-in is equally important for modeling the current day's preference $\mathbf{s}_n$.

After obtaining everyday preferences $\{\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_n\}$, we aim to incorporate historical days' preferences with the current day's preference by their spatio-temporal relationships to model the long-term preference. Considering there exists the periodic trends in people's daily life with time going by, different time interval from the past day to the present may generate different degrees of influence on the present. In addition, different day's range of activities is also different and the preferences in the days with similar range of activities with the present's, have more abilities to reflect the present day's preference. As a result, to distinguish the different importance of historical days' preference, we specially designed a weight ψ_h for each historical day in consideration of the influence factors mentioned above:

$$\psi_h = \frac{\exp(\tau_h(\mathbf{s}_h)^\top \mathbf{s}_n)}{\sum_{m=1}^{n-1}\exp(\tau_m(\mathbf{s}_m)^\top \mathbf{s}_n)}, h \in \{1, 2, \ldots, n-1\} \tag{8}$$

where $(\mathbf{s}_h)^\top \mathbf{s}_n$ is a preference matching score [24] between the h-th day and the current day, which is used to measure the similarity between the h-th day's preference and the current day's preference. And τ_h is introduced from Flashback architecture [26] to capture the periodicity property of the time influence made by historical days' preference as well as the spatial influence, and we further adjust the formulation of τ_h as follows:

$$\tau_h = \tau_h^{\mathrm{t}} \times \tau_h^{\mathrm{s}} \tag{9}$$

$$\tau_h^{\mathrm{t}} = \frac{\cos(\frac{2\pi}{7}\triangle T_{h,n}) + 1}{2} \cdot \exp(-\gamma \triangle T_{h,n}) \tag{10}$$

$$\tau_h^{\mathrm{s}} = \exp(-\delta \triangle D_{h,n}) \tag{11}$$

where τ_h^{t} and τ_h^{s} is employed to depict temporal and spatial factors respectively. τ_h^{t} is calculated by Eq. (10) where $\triangle T_{h,n}$ is time interval between the h-th day and the current day n with respect to the day and γ is the temporal decay rate controlling the decay speed of τ_h^{t} over time interval $\triangle T_{h,n}$. In order to see the effect of τ_h^{t} directly, we present its function graph in Fig. 4. We adjust the period of τ_h^{t} to 7 which reflects that people's activities usually take one week (i.e., 7 days) as a period, e.g., people may have the same activities this Monday as those in last Monday. Besides, as time goes by, the past will have less and less influence on the present. τ_h^{s} is calculated by Eq. (11) where $\triangle D_{h,n}$ is the distance between the central coordinates of the h-th day and the current day n and δ is the spatial decay rate controlling the decay speed of $\tau_{h,s}$ over spatial distance $\triangle D_{h,n}$. The central coordinate of each day's trajectory $S_h \in \{S_1, S_2, \ldots, S_n\}$is defined as follows:

$$lon_{S_h} = \frac{lon_{l_1} + lon_{l_2} + \ldots + lon_{l_{|S_h|}}}{|S_h|} \tag{12}$$

$$lat_{S_h} = \frac{lat_{l_1} + lat_{l_2} + \ldots + lat_{l_{|S_h|}}}{|S_h|} \tag{13}$$

which represents the range of activities in the corresponding day. The days sharing the more similar range of activities with the current day's may have more impact on the current day's preference.

Finally, the long-term preference $\mathbf{s}^{\mathrm{l}}$ incorporating each historical day's preferences $\{\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_{n-1}\}$is formulated as follows:

$$\mathbf{s}^{\mathrm{l}} = \sum_{h=1}^{n-1}\psi_h \mathbf{s}_h \tag{14}$$

5.2. Short-term preference

This section mines users' short-term preferences by considering the check-in sequence influence and the current time preference influenced by the public.

In Eq. (6), We have obtained $\{\mathbf{h}^1, \mathbf{h}^2, \ldots, \mathbf{h}^{p-1}\}$, the preference vectors related to each current day's check-in. The POI that the user wants to visit right now from the short-term aspect mainly depends on the user's recent mobile pattern and the preference influenced by the public in current time. The user's recent mobile pattern, i.e., the check-in sequence influence, has been modeled in the latest check-in's preference vector $\mathbf{h}^{p-1}$. As for the preference of the public, we introduce the d-dimensional time transition vector $\mathbf{t}^t \in \mathbb{R}^{d\times 1}$ for each time slot $t \in \{1, 2, \ldots, 48\}$ to represent the public preference in time slot t. We further define the user's current time preference $\mathbf{h}^p$ influenced by the public as follows:

$$\mathbf{h}^p = \mathbf{t}^{p-1}\mathbf{h}^{p-1} + \mathbf{t}^p\mathbf{h}^{p-1} \tag{15}$$

where $\mathbf{t}^{p-1}$ and $\mathbf{t}^p$ are the trainable time transition vectors with regard to the time slots of the latest check-in and the present. In

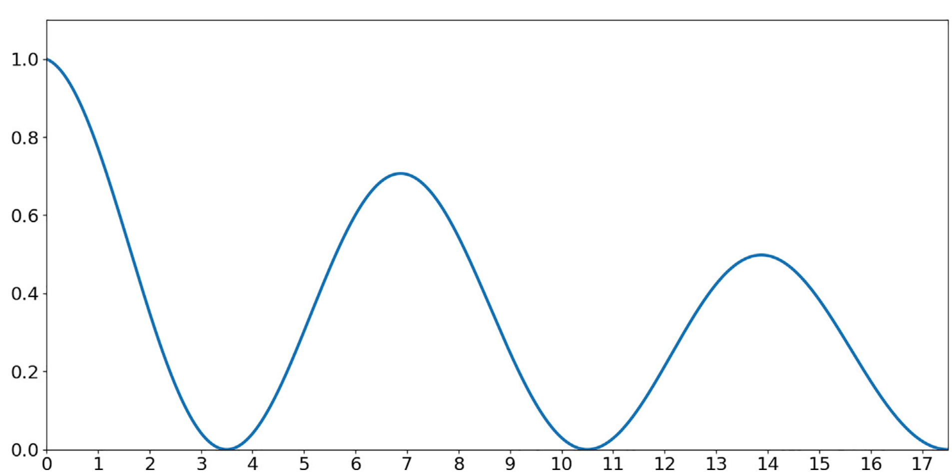


Fig. 4. Function graph of τ_h^{t}.

Eq. (15), the first term represents the user's preference in the latest check-in time slot t^{p-1} influenced by the public. And the second term represents corresponding preference influenced by the public in current time slot t^p.

Then, the short-term preference incorporating the check-in sequence influence with the current time preference influenced by the public is represented as follows:

$$\mathbf{s}^{s} = \eta \mathbf{h}^{p-1} + (1-\eta)\mathbf{h}^{p} \tag{16}$$

where η is a hyper-parameter to trade off the importance of the check-in sequence influence $\mathbf{h}^{p-1}$ and the current time's preference influenced by the public $\mathbf{h}^{p}$.

5.3. Probability calculation

In this section, we combine the long-term preference with the short-term preference to calculate the probability distribution $\mathbf{p}$ over all POIs $l \in L$ as follows:

$$\mathbf{p} = softmax(\mathbf{W}_p(\mathbf{s}^{l} \oplus \mathbf{s}^{s})) \tag{17}$$

where $\oplus$ is the concatenation of the long-term preference $\mathbf{s}^{l}$ and the short-term preference $\mathbf{s}^{s}$, and $\mathbf{W}_p \in \mathbb{R}^{|L| \times 2d}$ is a trainable projection matrix for all POIs. The POI with higher probability $p \in \mathbf{p}$ has more possibilities to be visited by the user at present. Then the objective function is formulated as the log likelihood:

$$\mathcal{L} = -\sum_{r=1}^{N} log(p_r) \tag{18}$$

where $p_r \in \mathbf{p}$ is the probability of the ground truth POI for the r-th training sample and N is the total number of all training samples. We aim to minimize the value of the objective function.

5.4. Recommendation

Because people may not visit the POIs belonging to some categories in specific time slots according to people's patterns of life as shown in Fig. 2, the existence of these POIs may affect the effect of the recommendation. As a result, we do not directly use the probabilities calculated by the Eq. (17) to generate the recommendation list like general methods. Instead, we first filter these categories and the POIs belonging to them on the basis of the relationship between the category popularity and the time slots by Eq. (19):

$$\mathbf{p}_f = \mathbf{p} \odot \mathbf{M}_f \tag{19}$$

where $\mathbf{M}_f$ is the filter matrix for the specific time slot which is actually a 0–1 matrix. It is constructed by counting up the number of check-ins for each POI categories in different time slots. If the amount is more than zero in current time slot, the position of POIs belonging to corresponding categories in $\mathbf{M}_f$ is set to 1, otherwise is set to 0. After the filter operation, the probability of POIs belonging to the unpopular categories in current time slot in $\mathbf{p}_f$ is set to 0 and others remain unchanged. Finally, recommendation lists are generated according to the probabilities $\mathbf{p}_f$ calculated by the Eq. (19). In this way, the POIs belonging to those categories which do not match people's patterns of life in current time slot won't be recommended and the recommendation performance will be promoted because of reducing the search space.

6. Experiments

6.1. Data descriptions

We evaluate our RTPM model on two real datasets: the Foursquare check-ins in New York and Tokyo [25] from 12 April 2012 to 16 February 2013. They are denoted as NYC and TKY, respectively. For both of them, we eliminate unpopular POIs visited by less than 10 users. A trajectory consists of a user's all check-ins in one day. We further remove inactive users with less than 5 trajectories. After preprocessing, the first 80% of each dataset is split into the training set and the rest serves as the testing set. The statistics of both datasets after preprocessing are shown in Table 1.

6.2. Evaluation metrics

We adopt *Rec@K* and *NDCG@K* to evaluate the performance. *Rec@K* is the ratio of the correct POIs among the top K recommended POIs to the groundtruth. *Rec@K* is defined as follows:

$$Rec@K = \frac{TP}{TP + FN} \tag{20}$$

where TP is the true positive and FN is the false negative. *NDCG@K* measures the quality of the top-K ranking list and can be calculated by the follows:

$$NDCG@K = \frac{1}{IDCG}\sum_{i=1}^{N}\frac{2^{rel_i}-1}{log(1+i)} \tag{21}$$

where $IDCG$ stands for the maximum possible DCG for a given recommendation list, and rel_i is 1 if the POI at position i in the recommendation list is visited and 0 otherwise. N is the number of correctly recommended POIs. In this paper, we choose $K = \{1, 3, 5\}$ for evaluation.

6.3. Baselines

We compare our RTPM model with the following six methods:

NCF [17]: This framework leverages a multi-layer perceptron to learn the user-POI interaction function and can express and generalize matrix factorization based on neural networks.

TCF [27]: This time-based collaborative filtering algorithm incorporates users' global similarity during a long period and local similarity within a short time interval into the collaborative filtering algorithm.

RNN [28]: This method uses a standard recurrent structure to model users' behavior sequences and make a prediction.

LSTM [29]: As a variant of RNN model, it contains a memory cell and three specially designed gates to better learn long-term dependency.

GRU [30]: Similar to LSTM, it is equipped with two gates to control the information flow to handle sequential data in a relatively low computational cost.

LSTPM [24]: This method models users' long-term and short-term preferences respectively by a nonlocal network and a geo-dilated RNN, incorporating spatio-temporal factors.

Table 1
Statistics of the evaluation datasets.

Datasets	#user	#POI	#category	#check-in	sparsity
NYC	741	12120	240	58361	99.72%
TKY	2032	22218	224	288847	99.76%

6.4. Results

The results of the baselines introduced above and our model are reported in Table 2. As we see, RTPM achieves the best performance compared with all baselines in terms of *Rec@K* and *NDCG@K* on both NYC and TKY datasets.

We observe that NCF and TCF perform worst in the scenario of next POI recommendation. Because they model users' preferences in a static manner and don't take sequential dependencies into account which is crucial to next POI recommendation. Considering the sequential dependencies, RNN improves the performance in the experiments. LSTM and GRU introduce the gating mechanism on the basis of RNN and perform better in the next POI recommendation issue. It means LSTM and GRU have more abilities to handle long sequences than RNN. LSTPM models both users' long-term preferences and short-term preferences and considers the spatio-temporal relationships among check-ins. Hence, it achieves a great success in the next POI recommendation. Besides, its performance reveals the importance of spatio-temporal factors. For our RTPM model, on the basis of LSTPM, we further think about the periodic trends of people's behaviors and design better weight setting methods. What's more, we consider time restrictions in RTPM for real-time recommendation. Specifically, we take both the influence of the public preference in different time slots and the popularity of POI categories changing over time into consideration, which are not considered in LSTPM and make RTPM have better real-time performance. As a result, our RTPM model achieves the best performance among these baselines.

6.5. Analysis on key components in RTPM

To verify the effectiveness of several key components in our model, we conduct more experiments on three simplified versions of our model.

The following two simplified versions of our model are designed to evaluate the long-term and short-term components. And in order to eliminate the influence of the category filter, we remove the category filter for both these two versions.

- **RTPM**$_{NC-L}$: This version removes the short-term component and the category filter in the recommendation module, and reserves the long-term component.
- **RTPM**$_{NC-S}$: This one removes the long-term component and the category filter in the recommendation module, and reserves the short-term component.

To verify the effectiveness of the category filter, we implement the following version:

- **RTPM**$_{NC}$: This version removes the category filter in the recommendation module of the RTPM and reserves the rest.

Table 3 presents the experimental results of the three simplified versions of RTPM. As we see, RTPM$_{NC-L}$ outperforms RTPM$_{NC-S}$ on more evaluation metrics. It demonstrates that the long-term preference component can better mine users' patterns of life and take more effect on the real-time POI recommendation. Even so, the short-term preference component is still indispensable. We can see that incorporating the long-term preference and the short-term preference, RTPM$_{NC}$ achieves a better performance. This indicates that although RTPM$_{NC-S}$ performs a little weak relatively, it also makes a contribution to the real-time POI recommendation. Compared with RTPM$_{NC}$, RTPM further improves the experimental results. This shows the effectiveness of the category filter operation and proves that the recommendation lists generated after the category filter operation fit users' real-time preferences and patterns of life in corresponding time slots better than the recommendation lists generated directly by the probabilities.

6.6. Analysis on the category filter

To further illustrate the necessity and the practical effect of the category filter, we count up the specific number of categories filtered out and the POIs belonging to these categories in each time slot. Then, we further calculate the percentage of those filtered out in the whole categories and POIs and present the results in Fig. 5. From Fig. 5a, We can find that at least 20% categories are filtered in each time slots which means there are at least 20% categories which few people visit in the corresponding time slots. And more than 70% POIs are filtered in each time slot so that the search space is greatly reduced.

6.7. Impact of parameters

We conduct some experiments to evaluate the impact of parameters α, η, γ and δ on NYC and the results are presented in Fig. 6–9 respectively. The results on TKY are similar and we omit them due to space limit. Besides, the results of *NDCG@*1 are the same as *Rec@*1, so the results of *NDCG@*1 are also omitted. In

Table 2
Baseline performance on NYC and TKY.

	NYC						TKY					
	Rec@1	Rec@3	Rec@5	NDCG@1	NDCG@3	NDCG@5	Rec@1	Rec@3	Rec@5	NDCG@1	NDCG@3	NDCG@5
NCF	0.0303	0.1088	0.1584	0.0303	0.0751	0.0955	0.0440	0.0861	0.1208	0.0440	0.0680	0.0820
TCF	0.0239	0.0460	0.0555	0.0239	0.0370	0.0409	0.0685	0.1261	0.1608	0.0685	0.1016	0.1158
RNN	0.1515	0.2522	0.2941	0.1515	0.2102	0.2275	0.1740	0.2934	0.3527	0.1740	0.2436	0.2680
LSTM	0.1595	0.2867	0.3507	0.1595	0.2329	0.2593	0.1976	0.3392	0.4063	0.1976	0.2797	0.3073
GRU	0.1625	0.2757	0.3260	0.1625	0.2284	0.2492	0.2083	0.3483	0.4117	0.2083	0.2897	0.3158
LSTPM	0.1836	0.3087	0.3707	0.1836	0.2559	0.2814	0.2088	0.3492	0.4135	0.2088	0.2902	0.3168
RTPM	**0.1944**	**0.3182**	**0.3752**	**0.1944**	**0.2663**	**0.2898**	**0.2143**	**0.3504**	**0.4151**	**0.2143**	**0.2934**	**0.3201**

Table 3
Performance of different versions of RTPM.

	NYC						TKY					
	Rec@1	Rec@3	Rec@5	NDCG@1	NDCG@3	NDCG@5	Rec@1	Rec@3	Rec@5	NDCG@1	NDCG@3	NDCG@5
RTPM$_{NC-L}$	0.1189	0.2617	0.3320	0.1189	0.2011	0.2301	0.1319	0.2916	0.3675	0.1319	0.2243	0.2556
RTPM$_{NC-S}$	0.1517	0.2468	0.2850	0.1517	0.2067	0.2224	0.1581	0.2632	0.3173	0.1581	0.2190	0.2414
RTPM$_{NC}$	0.1882	0.3139	0.3714	0.1882	0.2616	0.2852	0.2139	0.3499	0.4146	0.2139	0.2930	0.3196
RTPM	**0.1944**	**0.3182**	**0.3752**	**0.1944**	**0.2663**	**0.2898**	**0.2143**	**0.3504**	**0.4151**	**0.2143**	**0.2934**	**0.3201**

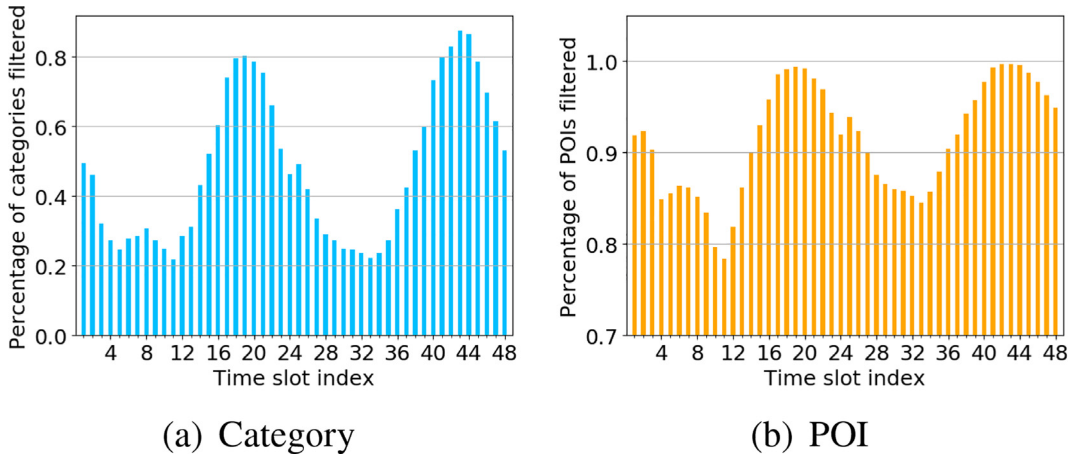


Fig. 5. The percentage of (a) Category and (b) POI filtered in different time slots.

Fig. 6–9, the performance of *Rec@K* and *NDCG@K* is different with these parameters changing. We aim to achieve better recommendation performance with less POIs recommended and thus we focus more on *Rec*@1 and *NDCG*@1.

α is used to trade off the importance of temporal and spatial influence when modeling everyday preferences. In Fig. 6a, we can see $\alpha = 0.6$ performs the best, which means temporal factors are a little more important than spatial factors when modeling everyday preferences.

η is the parameter to adjust the importance of the sequential influence and the current time's preference influenced by the public when modeling the short-term preference. In Fig. 7a, When $\eta = 0.7$, the performance is the best. So, when modeling the short-term preference, the sequential influence is more important than the current time's preference influenced by the public which means the patterns of life of oneself affect his plan more than others.

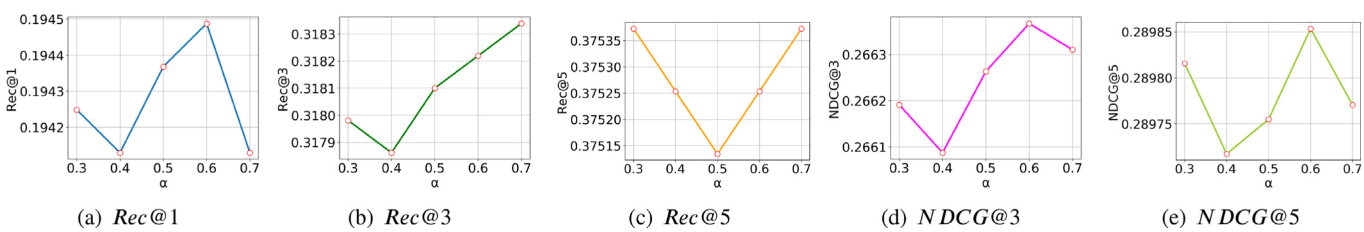


Fig. 6. Performance of *Rec@K* and *NDCG@K* w.r.t. parameter α on NYC.

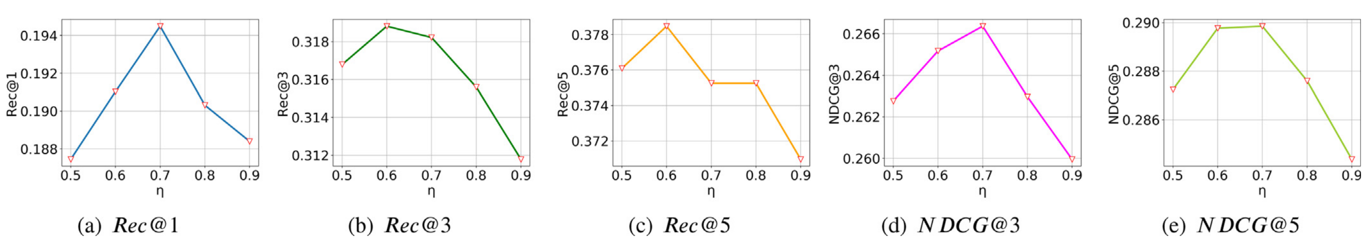


Fig. 7. Performance of *Rec@K* and *NDCG@K* w.r.t. parameter η on NYC.

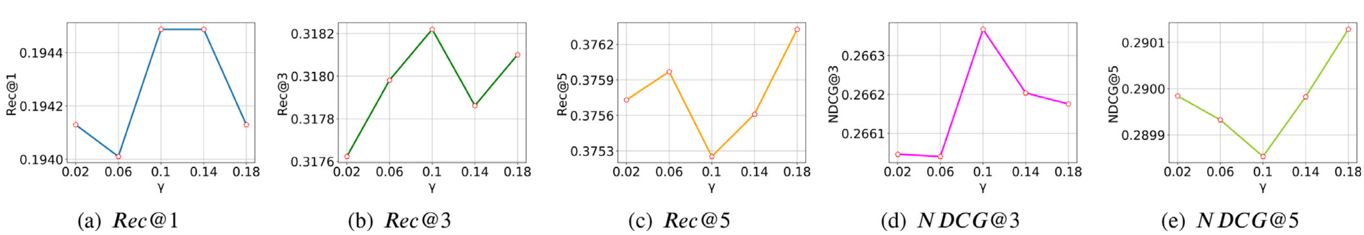
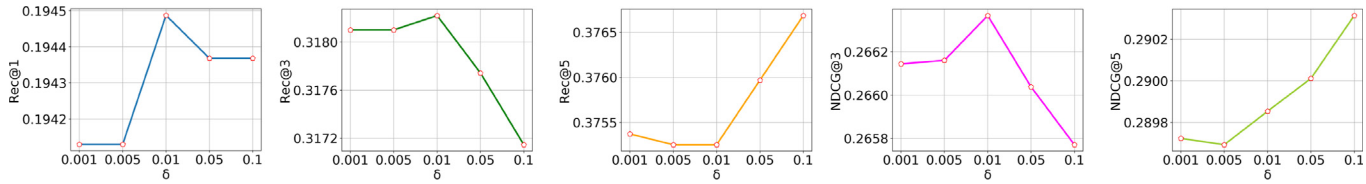


Fig. 8. Performance of *Rec@K* and *NDCG@K* w.r.t. parameter γ on NYC.



(a) *Rec*@1 (b) *Rec*@3 (c) *Rec*@5 (d) *NDCG*@3 (e) *NDCG*@5

Fig. 9. Performance of *Rec*@*K* and *NDCG*@*K* w.r.t. parameter δ on NYC.

γ and δ control the decay speed of the time interval and the distance from the historical day to the present respectively when modeling the long-term preference. As shown in Fig. 8a and Fig. 9a, they should not be set to very large or small values. Because, if they are too large, some historical information may be ignored. On the contrary, if they are too small, the model can not distinguish the importance of the past days. As a result, we should set them to the appropriate values, i.e., $\gamma = 0.1$ and $\delta = 0.01$.

7. Conclusion

In this paper, we propose a model named RTPM for real-time POI recommendation without utilizing the users' personal attributes and their current locations. RTPM mines users' real-time preferences with time restrictions from long-term and short-term preferences. On the one hand, it formulates the periodic trends of users' behaviors by spatio-temporal factors in the long-term preference. On the other hand, it considers the influence of the public preference in current time slot and the users' own sequential influence for the short-term preference. Moreover, we design a category filter to eliminate the POI categories which few people visit in current time slot to further reduce the search space and make the final recommendation fit users' living habits in current time slot better. Finally, we conduct extensive experiments on two real datasets to evaluate our model. The experimental results verify the effectiveness of our model and demonstrate that RTPM outperforms the baselines.

In future work, we may take users' regional transfer regularities into account to depict users' behavior patterns more comprehensively. And the randomness of users' preferences will also be considered.

CRediT authorship contribution statement

Xin Liu: Conceptualization, Methodology, Formal analysis, Software. **Yongjian Yang:** Supervision, Project administration. **Yuanbo Xu:** Formal analysis, Writing – review & editing. **Funing Yang:** Writing – review & editing. **Qiuyang Huang:** Software, Visualization. **Hong Wang:** Data curation.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work is supported by the National Natural Science Foundations of China under Grant No. 61772230, No. 6197-2450, National Natural Science Foundations of China for Young Scholars No. 61702215, No. 62002123 and China Innovation Postdoc Foundations No. BX20190140.

References

[1] Lu Zhang, Zhu Sun, Jie Zhang Yu, Chen Li Lei, Ziqing Wu, Horst Kloeden, Felix Klanner, An interactive multi-task learning framework for next POI recommendation with uncertain check-ins, Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI 2020 (2020) 3551–3557.

[2] Shanshan Feng, Lucas Vinh Tran, Gao Cong, Lisi Chen, Jing Li, and Fan Li. HME: A hyperbolic metric embedding approach for next-poi recommendation. In Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval, SIGIR 2020, Virtual Event, China, July 25-30, 2020, pp. 1429–1438, 2020..

[3] Qinyong Wang, Hongzhi Yin, Tong Chen, Zi Huang, Hao Wang, Yanchang Zhao, and Nguyen Quoc Viet Hung. Next point-of-interest recommendation on resource-constrained mobile devices. In WWW '20: The Web Conference 2020, Taipei, Taiwan, April 20-24, 2020, pp. 906–916, 2020..

[4] Ranzhen Li, Yanyan Shen, Yanmin Zhu. Next point-of-interest recommendation with temporal and multi-level context attention. In IEEE International Conference on Data Mining, ICDM 2018, Singapore, November 17–20, 2018, pp. 1110–1115, 2018..

[5] Jing He, Xin Li, Lejian Liao, Dandan Song, and William K. Cheung. Inferring a personalized next point-of-interest recommendation model with latent behavior patterns. In Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, February 12-17, 2016, Phoenix, Arizona, USA, pp. 137–143, 2016..

[6] Hao Wang, Huawei Shen, Wentao Ouyang, and Xueqi Cheng. Exploiting poi-specific geographical influence for point-of-interest recommendation. In Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13–19, 2018, Stockholm, Sweden, pp. 3877–3883, 2018..

[7] Tongcun Liu, Jianxin Liao, Zhigen Wu, Yulong Wang, and Jingyu Wang. A geographical-temporal awareness hierarchical attention network for next point-of-interest recommendation. In Proceedings of the 2019 on International Conference on Multimedia Retrieval, ICMR 2019, Ottawa, ON, Canada, June 10-13, 2019, pp. 7–15, 2019..

[8] Jing He, Xin Li, Lejian Liao, and Mingzhong Wang. Inferring continuous latent preference on transition intervals for next point-of-interest recommendation. In Machine Learning and Knowledge Discovery in Databases - European Conference, ECML PKDD 2018, Dublin, Ireland, September 10–14, 2018, Proceedings, Part II, volume 11052, pp. 741–756, 2018..

[9] Nicholas Lim, Bryan Hooi, See-Kiong Ng, Xueou Wang, Yong Liang Goh, Renrong Weng, and Jagannadan Varadarajan. STP-UDGAT: spatial-temporal-preference user dimensional graph attention network for next POI recommendation. In CIKM '20: The 29th ACM International Conference on Information and Knowledge Management, Virtual Event, Ireland, October 19-23, 2020, pp. 845–854, 2020..

[10] Xi Xiong, Shaojie Qiao, Nan Han, Fei Xiong, Bu. Zhan, Rong-Hua Li, Kun Yue, Guan Yuan, Where to go: an effective point-of-interest recommendation framework for heterogeneous social networks, Neurocomputing 373 (2020) 56–69.

[11] Mao Ye, Peifeng Yin, Wang-Chien Lee, Dik-Lun Lee, Exploiting geographical influence for collaborative point-of-interest recommendation, in: Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval, 2011, pp. 325–334.

[12] Quan Yuan, Gao Cong, Zongyang Ma, Aixin Sun, and Nadia Magnenat Thalmann. Time-aware point-of-interest recommendation. In Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 363–372, 2013..

[13] Mohammad Aliannejadi, Dimitrios Rafailidis, Fabio Crestani, A joint two-phase time-sensitive regularized collaborative ranking model for point of interest recommendation, IEEE Trans. Knowl. Data Eng. 32 (6) (2020) 1050–1063.

[14] Defu Lian, Cong Zhao, Xing Xie, Guangzhong Sun, Enhong Chen, and Yong Rui. Geomf: Joint geographical modeling and matrix factorization for point-of-interest recommendation. In Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 831–840, 2014..

[15] Xin Liu, Yong Liu, Karl Aberer, Chunyan Miao, Personalized point-of-interest recommendation by mining users' preference transition, in: Proceedings of the 22nd ACM International Conference on Information & Knowledge Management, 2013, pp. 733–738.

[16] Haijun Zhang, Yanfang Sun, Mingbo Zhao, Tommy W.S. Chow, Q.M. Jonathan Wu, Bridging user interest to item content for recommender systems: an optimization model, IEEE Trans. Cybern. 50 (10) (2020) 4268–4280.
[17] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. Neural collaborative filtering. In Proceedings of the 26th International Conference on World Wide Web, pp. 173–182, 2017..
[18] Wei Liu, Zhi-Jie Wang, Bin Yao, Jian Yin, Geo-alm: Poi recommendation by fusing geographical information and adversarial learning mechanism, in: Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19, 2019, pp. 1807–1813.
[19] Chen Cheng, Haiqin Yang, Michael R. Lyu, Irwin King, Where you like to go next: successive point-of-interest recommendation, in: Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence, 2013, pp. 2605–2611.
[20] Meng Chen, Yang Liu, and Xiaohui Yu. Nlpmm: A next location predictor with markov modeling. In Vincent S. Tseng, Tu Bao Ho, Zhi-Hua Zhou, Arbee L.P. Chen, and Hung-Yu Kao, editors, Advances in Knowledge Discovery and Data Mining, pp. 186–197, 2014..
[21] Buru Chang, Yonggyu Park, Donghyeon Park, Seongsoon Kim, Jaewoo Kang, Content-aware hierarchical point-of-interest embedding model for successive poi recommendation, in: Proceedings of the 27th International Joint Conference on Artificial Intelligence, 2018, pp. 3301–3307.
[22] Chenwang Zheng and Dan Tao. Attention-based dynamic preference model for next point-of-interest recommendation. In Wireless Algorithms, Systems, and Applications - 15th International Conference, WASA 2020, Qingdao, China, September 13–15, 2020, Proceedings, Part I, vol. 12384, pp. 768–780, 2020..
[23] Qiang Liu, Shu Wu, Liang Wang, and Tieniu Tan. Predicting the next location: A recurrent model with spatial and temporal contexts. In Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, February 12-17, 2016, Phoenix, Arizona, USA, pp. 194–200, 2016..
[24] Ke Sun, Tieyun Qian, Tong Chen, Yile Liang, Quoc Viet Hung Nguyen, and Hongzhi Yin. Where to go next: Modeling long- and short-term user preferences for point-of-interest recommendation. In The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7–12, 2020, pp. 214–221, 2020..
[25] Fuqiang Yu, Lizhen Cui, Wei Guo, Xudong Lu, Qingzhong Li, and Hua Lu. A category-aware deep model for successive POI recommendation on sparse check-in data. In WWW '20: The Web Conference 2020, Taipei, Taiwan, April 20-24, 2020, pp. 1264–1274, 2020..
[26] Dingqi Yang, Benjamin Fankhauser, Paolo Rosso, Philippe Cudré-Mauroux, Location prediction over sparse user mobility traces using rnns: Flashback in hidden states!, in: Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI, 2020, pp 2184–2190.
[27] Jun Zeng, Xin He, Feng Li, Yingbo Wu, A recommendation algorithm for point of interest using time-based collaborative filtering, Int. J. Inf. Technol. Manag. 19 (4) (2020) 347–357.
[28] Yuyu Zhang, Hanjun Dai, Chang Xu, Jun Feng, Taifeng Wang, Jiang Bian, Bin Wang, and Tie-Yan Liu. Sequential click prediction for sponsored search with recurrent neural networks. In Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence, July 27 -31, 2014, Québec City, Québec, Canada, pp. 1369–1375, 2014..
[29] Sepp Hochreiter, Jürgen Schmidhuber, Long short-term memory, Neural Comput. 9 (1997) 1735–1780.
[30] Kyunghyun Cho, Bart van Merrienboer, Çaglar Gülçehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using RNN encoder-decoder for statistical machine translation. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25–29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL, pp. 1724–1734, 2014..

Xin Liu received the B.E. degree in resources exploration engineering from Jilin University, Changchun, China, in 2018. He is currently pursuing the M.E. degree in the College of Computer Science and Technology, Jilin University, Changchun, China. His research interests include recommender system and mobile computing.

Yongjian Yang received his B.E. degree in automatization from Jilin University of Technology, Changchun, Jilin, China, in 1983; and M.E. degree in Computer Communication from Beijing University of Post and Telecommunications, Beijing, China, in 1991; and his Ph.D. in Software and theory of Computer from Jilin University, Changchun, Jilin, China, in 2005. He is currently a professor and a PhD supervisor at Jilin University, Director of Key lab under the Ministry of Information Industry, Standing Director of Communication Academy, member of the Computer Science Academy of Jilin Province. His research interests include: Theory and software technology of network intelligence management; Key technology research of wireless mobile communication and services. He participated 3 projects of NSFC, 863 and funded by National Education Ministry for Doctoral Base Foundation. He has authored 12 projects of NSFC, key projects of Ministry of Information Industry, Middle and Young Science and Technology Developing Funds, Jilin provincial programs, Shenzhen, Zhuhai, and Changchun.

Yuanbo Xu received his B.Sc. degree and M.S. degree in the College of Computer Science and Technology, Jilin University, Changchun, China. He is pursuing his Ph.D. degree in Key Laboratory of Symbol Computation and Knowledge Engineering of the Ministry of Education, Jilin University, Changchun, China. His research interests include applications of data mining, recommender system, and mobile computing.

Funing Yang was born in Jilin City of Jilin Province, China, in 1987. She received her B.E. degree in college of software engineering from Jilin University, Changchun, in 2010, and the master's degree from the school of computer science, Beijing University of Posts and Telecommunications, Beijing, China, in 2013. She is currently a teacher in Jilin University, Changchun. Her current research interests include management, integration and mining of massive traffic data.

Qiuyang Huang received his B.E. degree in software engineering from Jilin University, Chang-chun, in 2013, his M.E. degree in software engineering from Jilin University, Changchun, in 2016. He is currently a Ph.D. in the computer science and technology at Jilin University, Changchun. He is also a visiting scholar at the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore. His research interests include applications of data mining, urban computing, and intelligent transportation systems. He has published some research results on journals such as Scientific Data, Neurocomputing.

Hong Wang graduated from Jilin Normal University with a master's degree, in 2008. She is currently a lecturer of Army Academy of Armored Forces. Her research interests include: Probability and Mathematical Statistics.