Weighted multi-information constrained matrix factorization for personalized travel location recommendation based on geo-tagged photos

Dandan Lyu[1] · Ling Chen[1] (ORCID) · Zhenxing Xu[1] · Shanshan Yu[1]

Abstract

Given travel history, travel location recommendation can automatically suggest users where to visit. Huge efforts have been devoted to introducing different additional information (e.g., sequential, textual, geographical, and visual information) for enhancing recommendation performance. However, existing methods only consider limited additional information and treat different information equally. In this paper, we present Weighted Multi-Information Constrained Matrix Factorization (WIND-MF) for personalized travel location recommendation based on geo-tagged photos. On one hand, photos (visual information), users' visit sequences (sequential information), and textual tags (textual information) are leveraged to comprehensively profile users and travel locations. On the other hand, visual, sequential, and textual similarities as well as geographical distance based co-visit probabilities are assigned with different weights to constrain the factorization of the original user-travel location matrix. We experimented on a dataset of six cities in China, and the experiment results verify the superiority of the proposed method. The code and dataset is available at https://github.com/revaludo/WIND-MF.

Keywords Geo-tagged photos · Personalized travel location recommendation · Matrix factorization · Multi-information · Similarity regularization

1 Introduction

With the popularity of smart devices and the support of online sharing platforms, people can share their footprints on the Internet anytime and anywhere (e.g., check-in at or upload the geo-tagged photos of visited locations). Generally, users can rely on manual search from these large amounts of information to find some locations satisfying their preferences, which is usually much cumbersome and time-consuming. Location recommendation systems [1] can mine users' preferences from

their history and automatically provide suitable locations, which would provide a huge convenience for users.

In recent years, location recommendation has attracted the attention of the research community. According to the types of data sources used by the researchers, location recommendation studies can be divided into three categories: 1) full GPS trajectory based location recommendation, 2) check-in based location recommendation, and 3) geo-tagged photo based location recommendation. For the early full GPS trajectory based location recommendation, researchers often use full GPS trajectory data to mine interesting locations and users' movement patterns. Then, they employ the similarities derived from users' history to provide personalized location recommendation [2–4]. For check-in based location recommendation, researchers usually consider social relationship to provide more appropriate recommendation results [5–8]. For geo-tagged photo (usually attached with a time-stamp and a coordinate, indicating when and where the photo was taken) based location recommendation, researchers usually first extract travel locations exploiting the geotags of photos, and then take users' preferences into account to recommend locations [9–11].

✉ Ling Chen
   lingchen@cs.zju.edu.cn

   Dandan Lyu
   revaludo@zju.edu.cn

   Zhenxing Xu
   xuzhenxing@zju.edu.cn

   Shanshan Yu
   shshyu@zju.edu.cn

[1] College of Computer Science and Technology, Zhejiang University, Hangzhou 310027, China

To improve recommendation performance, different additional information (e.g., sequential [12–14], textual [7, 15–17], geographical [5, 18, 19], and visual [20–23] information) has been introduced into location recommendation methods. Existing studies can be divided into two categories: The former learns features from additional information and users' history simultaneously to make personalized recommendation [24, 25], which usually needs lots of training examples to learn good representations of users and locations; The latter usually first extract features from additional information and then train a recommendation model based on users' history by using these features as priors [26, 27], which can integrate different additional information flexibly and achieve comparable performance under fewer training examples. However, there exists no method that exploits all of the above-mentioned additional information, and the significance of different additional information is not well studied.

To address the above-mentioned problem, we propose Weighted Multi-Information Constrained Matrix Factorization (WIND-MF) for personalized travel location recommendation based on geo-tagged photos. We firstly exploit multi-information to profile users and travel locations, and then assign different weights to corresponding similarities for personalized travel location recommendation.

The main contributions of this paper are as follows:

1) Propose WIND-MF for personalized travel location recommendation based on geo-tagged photos, which can exploit photos, users' visit sequences, and textual tags, to profile users and locations comprehensively.
2) Assign different weights to visual, sequential, and textual similarities as well as co-visit probabilities, which can capture the significance of different additional information.
3) Conduct extensive experiments to study the impact of different additional information. The results reveal that visual and sequential information contributes most to improving recommendation performance.

The rest of the paper is organized as follows. Section 2 reviews the related work. Section 3 gives the definitions of some basic concepts and then formally defines the problem. Section 4 introduces the proposed method. Section 5 presents the experiment results, and Section 6 concludes the paper and gives a brief outlook on future work.

2 Related work

In this section, we demonstrate existing studies related to our work, including location recommendation exploiting different types of data sources and location recommendation exploiting different additional information.

2.1 Location recommendation exploiting different types of data sources

Early location recommendation studies are mostly based on full GPS trajectories [2–4, 28]. Zheng et al. [2, 3] leveraged GPS trajectory data recorded by multiple users to mine and recommend interesting locations as well as classical sequences within a given geospatial region. Due to privacy issues, it is difficult for the researchers to obtain a large amount of full GPS trajectory data [29]. Therefore, researchers seek to extract partial GPS trajectories from other easily accessed data sources, e.g., check-ins and geo-tagged photos.

Nowadays, with the development of Location-Based Social Networks (LBSNs) [29], people can share their locations with others and make comments on their visited locations, which causes check-in based location recommendation. Early studies mainly take social relationship into account to provide more appropriate recommendation results [30, 31]. Long and Joshi [31] proposed a Hypertext Induced Topic Search (HITS) based Point of Interest (POI) recommendation algorithm, which recommends POIs based on users' social network links and check-in behaviour. Gao et al. [30] proposed a social-historical model to explore users' check-in behaviour, which integrates social and historical effects, and assesses the role of social correlation in users' check-in behaviour. To improve recommendation performance, different additional information (e.g., sequential, textual, geographical, and visual information) has been introduced, which we will introduce in the following subsection.

Geo-tagged photos provide rich location-based data, which can be exploited for location recommendation. Early studies mainly take users' preferences into account to recommend locations [9–11]. Clements et al. [9] firstly defined a similarity between the geotag distributions of two users based on a Gaussian kernel convolution, and predicted a user's favourite locations in a new city based on the rankings of the most similar users in the target city. Mamei et al. [10] proposed to recommend interesting locations in a new city to a user by exploiting instance-based Pearson collaborative filtering (CF). Popescu and Grefenstette [11] firstly mined users' location visiting records to build a user-user similarity matrix, and recommended interesting locations in a new destination based on the experience of like-minded users who already visited that destination. To improve recommendation performance, different additional information (e.g., sequential, textual, geographical, and visual information) has been introduced, which we will introduce in the following subsection.

2.2 Location recommendation exploiting different additional information

Sequential information Since human movement has strong sequential patterns, i.e., the probability of the transition from

one location to another is non-uniform [32], researchers tend to exploit sequential information. Markov chain models, Recurrent Neural Networks (RNNs), and word2vec have been used to model location sequences in the previous works. Kurashima et al. [14] used a first-order Markov chain model to predict the visit probability of a next location dependent on the previous location. Yang et al. [12] employed RNN and Gated Recurrent Unit (GRU) models to capture short and long term sequential information in mobile trajectories, respectively. Liu et al. [13] leveraged word2vec to learn the latent representation of a location by capturing the influence of its context locations. Inspired by the success of word2vec to capture sequential information, we exploit doc2vec [33] (an extension of word2vec, which can learn the vectors of documents) to get the sequential representations of users and locations.

Textual information Since locations visited by the same user tend to be semantically similar [34], researchers tend to exploit textual information. Term Frequency-Inverse Document Frequency (TF-IDF), topic models, and sentiments have been used to capture textual information in the previous works. Majid et al. [15, 16] and Memon et al. [35] leveraged TF-IDF to process the textual tags of geo-tagged photos. Jiang et al. [17] extracted the topics of users' preferences from the textual tags of geo-tagged photos via an author topic model. Gao et al. [7] studied the relationship between users' check-in behaviour and sentiment indications extracted from their tips. Inspired by the success of topic models to capture textual information, we exploit Latent Dirichlet Allocation (LDA) [36] to get the textual representations of users and locations.

Geographical information Since there is a strong correlation between users' check-in behaviour and geographical distance, researchers tend to exploit geographical information. One major research direction assumes that the distances of visited locations follow a power law distribution derived from the whole check-in history of all users. Yuan et al. [18] and Ye et al. [5] fused CF with geographical information modeled by a power law distribution for POI recommendation. Another research direction firstly clusters the whole check-in history of all users to find the most popular locations as centers, and assumes that the distances between visited locations and their centers follow a multi-center Gaussian model. Cheng et al. [19] fused geographical information modeled by a multi-center Gaussian model into a generalized matrix factorization framework for POI recommendation. Inspired by the success of the power law distribution to capture geographical information, we exploit it to model the co-visit probabilities of locations.

Visual information In addition to the above well-studied information, researchers have attempted to exploit visual information. Originally, users' attributes extracted via face recognition have been exploited to build users' profiles. Cheng et al. [22] extracted users' attributes (e.g., gender, age, and race) from photo contents, which was extended by further considering travel group types (e.g., family, friends, and couple) [23]. Recently, deep neural networks have been exploited to extract visual features. Wang et al. [21] used a VGG16 model to extract visual features from images and incorporated them into a probabilistic model for learning the latent features of users and POIs. Zhang et al. [37] used a Bayesian stacked convolutional auto-encoder to extract visual features from images and incorporated them into a pair-wise ranking model. Inspired by the success of auto-encoders to capture visual information, we exploit Variational Auto-Encoder (VAE) [38] to get the visual representations of users and locations.

Multi-information Recently, researchers also have exploited multi-information for location recommendation. Existing studies can be divided into two categories. The former learns features from additional information and users' history simultaneously to make personalized recommendation [24, 25], which usually needs lots of training examples to learn good representations of users and locations. Zhou et al. [25] proposed a multi-context trajectory embedding framework, which embeds user-level, trajectory-level, location-level, and temporal contexts into a shared low-dimension space. Yang et al. [24] developed PACE, which jointly predicts user's preferences over POIs, user's friends, and POI's nearby POIs, and ensures that users or POIs sharing more similar friends or nearby POIs will have closer embeddings.

The latter usually first extract features from additional information and then train a recommendation model based on users' history by using these features as priors [26, 27], which can integrate different additional information flexibly and achieve comparable performance under fewer training examples. Xu et al. [27] firstly exploited gender and age information extracted from photos, POI category distribution, temporally fine-grained users' preferences, etc., to profile users and travel locations, and then calculated user-user and travel location-travel location similarities to constrain the factorization of user-travel location matrix. Ding and Chen [39] proposed RecNet, which firstly factorizes co-visiting, geographical proximity, and categorical correlation matrices to obtain the embeddings of POIs and users, and then feeds the embedded POIs and users into a deep neural network to adaptively learn high-order interactions between them. However, there exists no method that exploits all of the above-mentioned additional information, and the significance of different additional information is not well studied.

3 Preliminaries and problem definition

In this section, we give the definitions of some basic concepts and terms, and then formally define the problem.

Definition 1 (Geo-tagged photo) A geo-tagged photo can be defined as $p = (u, g, t, T)$, where u is the user who contributed the photo, g is the coordinate where the photo was taken, t is the time-stamp when the photo was taken, and T is the textual tag set of the photo.

Definition 2 (User) A user is a person who has taken and shared geo-tagged photos online, and can be defined as $u = (P, T)$, where P is the set of geo-tagged photos taken by the user, and T is the set of textual tags belonging to all the photos taken by the user. A set of users can be defined as $U = \{u_1, u_2, \cdots, u_{|U|}\}$.

Definition 3 (Travel location) A travel location can be defined as $l = (c, g, P, T)$, where c is the city the location is in, g is its coordinate, P is the set of geo-tagged photos taken at the location, and T is the set of textual tags belonging to all the photos taken at the location. A set of travel locations can be defined as $L = \{l_1, l_2, \cdots, l_{|L|}\}$.

Definition 4 (Visit) A visit can be defined as $v = (u, l, t)$, which denotes that user u has visited travel location l at time t.

Definition 5 (User-travel location matrix) User-travel location matrix can be defined as M, whose element M_{ij} ($0 < i \leq |U| \wedge 0 < j \leq |L|$) represents the visit frequency of user u_i to travel location l_j.

Definition 6 (Visit sequence). Given time threshold ΔT, the j th visit sequence of user u_i is $S_i^j = \left[(l_1, t_1), (l_2, t_2), \cdots, \left(l_{|S_i^j|}, t_{|S_i^j|} \right) \right]$, where $0 < t_{k+1} - t_k \leq \Delta T$ $\left(1 \leq k < |S_i^j| \right) \wedge |S_i^j| \geq 2$. $SS_i = \left\{ S_i^1, S_i^2, \cdots, S_i^{|SS_i|} \right\}$ denotes the visit sequence set of the user.

Our research problem can be formulated as: Given target user u and query city c where the user wants to travel for the first time, i.e., the query is $q = (u, c)$, our purpose is to recommend a set of travel locations in city c that user u would be interested in.

4 Method

The framework of the proposed method is shown in Fig. 1. Firstly, the geo-tagged photos are clustered to find travel locations. Based on the visits identified by consecutive photos taken during a threshold time period by a same user at a same travel location, we build the original user-travel location matrix M and extract users' visit sequences. After that, we profile users and travel locations by exploiting the visual, sequential, and textual information of geo-tagged photos via VAE, doc2vec, and LDA, respectively. In addition, the co-visit

probabilities of travel locations are modeled by a power-law distribution based on their geographical distances. Afterwards, different weights are assigned to visual, sequential, and textual similarities as well as co-visit probabilities to obtain weighted user-user and travel location-travel location similarities, which are then used as regularization terms to constrain the factorization of M. Finally, we get the completed user-travel location matrix R, according to which, we can recommend some suitable travel locations to the target user.

4.1 Find travel locations

Since people usually take a lot of photos at travel locations, finding travel locations can be regarded as a problem of recognizing places photographed frequently. Researchers have used clustering algorithms to extract travel locations, e.g., mean shift [14, 22, 23], OPTICS [2], and DBSCAN [40, 41]. Generally speaking, DBSCAN has the following advantages compared to other clustering algorithms: 1) need minimum domain knowledge to determine the parameters (do not need to determine the number of clusters in advance) and be able to identify clusters of arbitrary shapes, 2) can filter out abnormal points, 3) maintains high efficiency when dealing with large-scale data. However, DBSCAN fails to meet our requirements, as it holds a uniform density threshold for all clusters, while clusters that we would like to extract from geo-tagged photos may have different sizes and densities. In order to solve this problem, Kisilevich et al. [42] proposed P-DBSCAN algorithm, which extends the definition of directly density-reachable by using adaptive density. Therefore, we use P-DBSCAN algorithm to find travel locations from geo-tagged photos, i.e., obtaining a set of travel locations L.

4.2 Build the original matrix and extract visit sequences

The visit frequency of a user to a travel location indirectly reflects the degree that the user prefers the location. Therefore, we first get the visit frequency of each user-travel location pair. Like [43], first of all, we use geo-tagged photos taken by different users at different travel locations to identify each visit. Specifically, given a user-travel location pair, we first sort the geo-tagged photos according to their taken time. Since a user might take more than one geo-tagged photo within one visit, if the taken time interval of two consecutive photos is less than visit duration threshold t_{thr}, we presume that these two photos belong to the same visit. The mean taken time of the photos belonging to a same visit is regarded as the visit time. After processing all users' visit history, we can count the visit frequency of each user-travel location pair, to get the original user-travel location matrix M.

Considering that users' visit order of travel locations usually can reflect their travel preferences to a certain extent, we
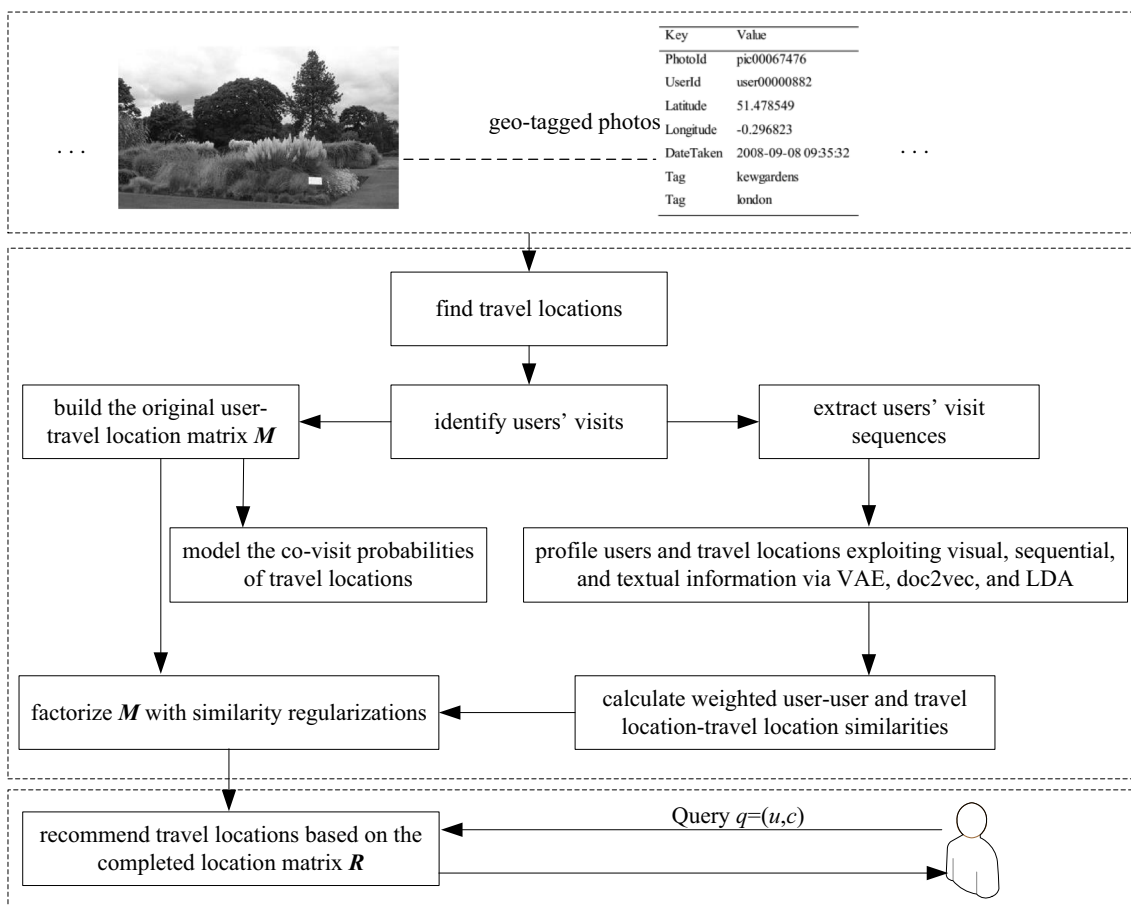
Fig. 1 The framework of the proposed method

extract users' visit sequences based on their visit history. Specifically, we firstly sort the visit history of user u_i according to visit time. Then, according to time threshold ΔT, we segment u_i's visit history to obtain the visit sequence set of the user, i.e., SS_i.

4.3 Profile users and travel locations

In this section, we consider visual, sequential, and textual information to profile users and travel locations.

4.3.1 Visual information modeling

Photos taken by users at travel locations contain a large amount of visual information. In order to make full use of it, we use VAE to get important visual characteristics from photos.

VAE [38] is an unsupervised learning method to learn complicated distributions of data, whose loss function is given by Eqs. 1–4. Given all the geo-tagged photos $X \in R^{r \times 32 \times 32 \times 3}$, where r is the number of photos, we assume that X is generated by a directed graphical model $P(X|z)$, and the encoder is learning an approximation $Q(z|X)$ to the posterior distribution

$P(z|X)$.

$$L_{\mathrm{VAE}} = E_{z \sim Q}[\log P(X|z)] - D_{KL}\Big(Q(z|X)\|P(z)\Big) \quad (1)$$

$$P(z) = N(z|\mathbf{0}, \mathbf{I}) \quad (2)$$

$$Q(z|X) = N\big(z|\boldsymbol{\mu}, \boldsymbol{\sigma}^2 * \mathbf{I}\big) \quad (3)$$

$$P(X|z) = N\Big(X|\boldsymbol{\mu}', \boldsymbol{\sigma}'^2 * \mathbf{I}\Big) \quad (4)$$

where the latent variable z obeys normal distribution, and is viewed as the visual features of photos X. Specifically, the visual feature of photo X_k ($0 \le k < r$) is z_k. $\mathbf{I}$ is an identity matrix. $Q(z|X)$ gives a distribution over z values that are likely to produce X. $P(X|z)$ measures the amount of information required to reconstruct X from z under an ideal encoding. $\boldsymbol{\mu}$, $\boldsymbol{\sigma}$, $\boldsymbol{\mu}'$, and $\boldsymbol{\sigma}'$ are arbitrary deterministic functions that can be learned from data, and are implemented via neural networks.

The visual features of all the photos taken at travel location l_j are averaged to get the visual representation of the travel location, denoted as v_{l_j}; meanwhile, the visual features of all the photos taken by user u_i are averaged to get the visual representation of the user, denoted as v_{u_i}.

4.3.2 Sequential information modeling

Since the order how users visit travel locations can reflect their travel preferences to a certain extent, we use doc2vec to get the sequential representations of users and travel locations.

Doc2vec [33] was proposed to learn the vector of variable length text (e.g., sentences, paragraphs, and documents) and is based on word2vec [44]. It can be divided into Distributed Memory version of Paragraph Vector (PV-DM) and Distributed Bag of Words version of Paragraph Vector (PV-DBOW). Since PV-DM considers the concatenation of the document vector with the word vectors of context words to predict the next word in a text window, while PV-DBOW ignores context words, we use PV-DM.

An illustrative framework of PV-DM is shown in Fig. 2. Treating user u_i's visit sequence set SS_i as a document, PV-DM predicts the next travel location l_j by considering the user and context travel locations l_{j-w} to l_{j+w}, whose objective function is to maximize the average log probability in Eqs. 5–7:

$$L(SS_i) = \frac{1}{|SS_i|} \sum_{l_j \in SS_i} p\left(l_j | u_i, l_{j-w}, \cdots, l_k \cdots, l_{j+w}\right) \quad (5)$$

$$p\left(l_j | u_i, l_{j-w}, \cdots, l_k \cdots, l_{j+w}\right) = \frac{\exp\left(\mathbf{s}_j^{\mathrm{T}} \bullet \mathbf{v}\right)}{\sum_{l_{j'} \in L} \exp\left(\mathbf{s}_{l_{j'}}^{\mathrm{T}} \bullet \mathbf{v}\right)} \quad (6)$$

$$\mathbf{v} = \left(\mathbf{s}_{u_i} + \mathbf{s}_{l_{j-w}} + \cdots + \mathbf{s}_{l_k} \cdots + \mathbf{s}_{l_{j+w}}\right)/(2w+1) \quad (7)$$

where w is the size of context window. $l_{j'}$ $\left(l_j \neq l_{j'}\right)$ is one of the travel locations in the travel location set, and l_k $(j - w < k < j + w, k \neq j)$ is one of the context travel locations. $\mathbf{s}_{l_j}$, $\mathbf{s}_{l_{j'}}$, and $\mathbf{s}_{l_k}$ are the embeddings of travel locations l_j, $l_{j'}$, and l_k, respectively. $\mathbf{s}_{u_i}$ is the embedding of user u_i.
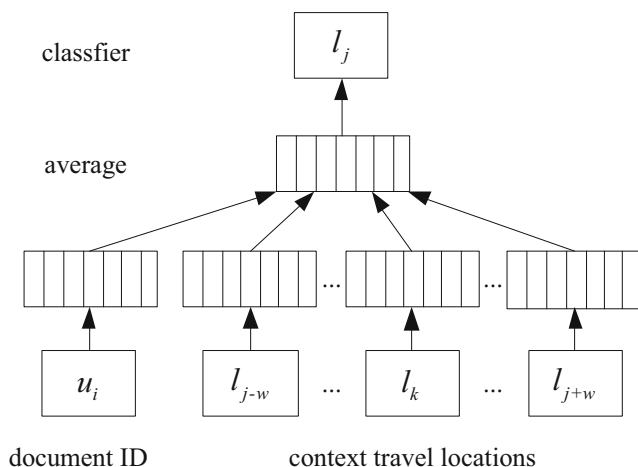
4.3.3 Textual information modeling

In addition to the rich visual information contained in geo-tagged photos, their corresponding metadata also hold a large amount of textual information, which is also very important to profile users and travel locations [14, 16].

LDA is a representative topic model [45], which was first put forward and applied in the field of natural language processing. In LDA, each document is a probability distribution of a series of topics, and each word's presence is attributable to one of the document's topics.

Treating a textual tag as a word, and the textual tag set of a user as a document, the generation process is as follows (as shown in Fig. 3):

1) For $i \in [1, k]$

Sample $\varphi_i \sim \mathrm{Dir}(\beta)$, where φ_i is the word distribution of the i th topic, k is the number of topics, and $\mathrm{Dir}(\beta)$ is the Dirichlet distribution of parameter β:

2) For $i \in [1, |U|]$

Sample $\theta_{u_i} \sim \mathrm{Dir}(\alpha)$, where θ_{u_i} is the topic distribution of user u_i's textual tag set, and $\mathrm{Dir}(\alpha)$ is the Dirichlet distribution of parameter α.

3) For each textual tag in u_i's textual tag set:

Sample topic $z \sim Multinomial(\theta_{u_i})$.
Sample word $w \sim Multinomial(\varphi_z)$.

Similarly, treating a textual tag as a word, and the textual tag set of a travel location as a document, we can obtain the topic distributions of travel locations. Specifically, the topic distribution of travel location l_j's textual tag set is θ_{l_j}.

4.4 Calculate user similarity and travel location similarity

We use cosine similarity, which is one of the most popular similarity measures [27, 46, 47], to measure user-user and travel location-travel location similarities based on visual,
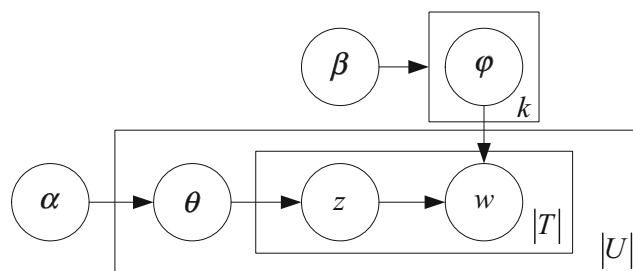

Fig. 2 The framework of PV-DM

Fig. 3 The graph model of LDA topic model

sequential, and textual information. For example, visual similarity between two users u_i and u_j can be calculated by Eq. 8:

$$\text{Sim}\left(\boldsymbol{v}_{u_i}, \boldsymbol{v}_{u_j}\right) = \frac{\boldsymbol{v}_{u_i} \bullet \boldsymbol{v}_{u_j}}{\|\boldsymbol{v}_{u_i}\|_F^2 \times \|\boldsymbol{v}_{u_j}\|_F^2} \tag{8}$$

where $\|\boldsymbol{v}_{u_i}\|_F^2$ and $\|\boldsymbol{v}_{u_j}\|_F^2$ represent the Frobenius norms of $\boldsymbol{v}_{u_i}$ and $\boldsymbol{v}_{u_j}$, respectively. Similarly, sequential and textual similarities between u_i and u_j, as well as visual, sequential, and textual similarities between two travel locations l_i and l_j can be calculated, denoted by, $\text{Sim}\left(\boldsymbol{s}_{u_i}, \boldsymbol{s}_{u_j}\right)$, $\text{Sim}\left(\boldsymbol{\theta}_{u_i}, \boldsymbol{\theta}_{u_j}\right)$, $\text{Sim}\left(\boldsymbol{v}_{l_i}, \boldsymbol{v}_{l_j}\right)$, $\text{Sim}\left(\boldsymbol{s}_{l_i}, \boldsymbol{s}_{l_j}\right)$, and $\text{Sim}\left(\boldsymbol{\theta}_{l_i}, \boldsymbol{\theta}_{l_j}\right)$, respectively.

In addition, based on the coordinates, the geographical distance between l_i and l_j can be obtained, denoted as $\text{dis}(l_i, l_j)$. Then, we use a power law distribution [5] to model the co-visit probability of l_i and l_j, which is given by Eq. 9:

$$y\left(l_i, l_j\right) = a \times \text{dis}\left(l_i, l_j\right)^b \tag{9}$$

where a and b are the parameters of the power-law distribution, which can be learned by linear regression.

Finally, weighted user-user and travel location-travel location similarities can be calculated by Eq. 10 and Eq. 11, respectively:

$$\begin{aligned} SimU\left(u_i, u_j\right) = {} & w_1 \times \text{Sim}\left(\boldsymbol{v}_{u_i}, \boldsymbol{v}_{u_j}\right) + w_2 \\ & \times \text{Sim}\left(\boldsymbol{s}_{u_i}, \boldsymbol{s}_{u_j}\right) + (1-w_1-w_2) \\ & \times \text{Sim}\left(\boldsymbol{\theta}_{u_i}, \boldsymbol{\theta}_{u_j}\right) \end{aligned} \tag{10}$$

$$\begin{aligned} SimL\left(l_i, l_j\right) = {} & w_3 \times \text{Sim}\left(\boldsymbol{v}_{l_i}, \boldsymbol{v}_{l_j}\right) + w_4 \times \text{Sim}\left(\boldsymbol{s}_{l_i}, \boldsymbol{s}_{l_j}\right) \\ & + w_5 \times \text{Sim}\left(\boldsymbol{\theta}_{l_i}, \boldsymbol{\theta}_{l_j}\right) + (1-w_3-w_4-w_5) \\ & \times y\left(l_i, l_j\right) \end{aligned} \tag{11}$$

where w_1, w_2, w_3, w_4, and w_5 are similarity weights.

4.5 Factorize the original matrix

Matrix factorization has been successfully used in recommender systems [48–51]. It can find the latent variables between users and items, which can reflect the characteristics of users and items to a certain extent. Through matrix factorization, the original user-travel location matrix M can be approximated by the product of two factorized matrices U and L, which is given by Eq. 12.

$$M \approx U^T L = R \tag{12}$$

The objective function of matrix factorization is given by Eq. 13.

$$\begin{aligned} V = {} & \min_{U,L} \frac{1}{2} \sum_{0 < i \le |U|, 0 < j \le |L|} \left(M_{ij} - U_i^T L_j\right)^2 \\ & + \frac{\lambda}{2}\left(\|U\|_F^2 + \|L\|_F^2\right) \end{aligned} \tag{13}$$

where $\|U\|_F^2$ and $\|L\|_F^2$ represent the Frobenius norms of matrices U and L, respectively, and λ is a regularized parameter that is used to prevent over-fitting.

Since visit frequency is a kind of implicit feedback, we exploit weighted matrix factorization [52], whose objective function is given by Eq. 14:

$$\begin{aligned} V = {} & \min_{U,L} \frac{1}{2} \sum_{0 < i \le |U|, 0 < j \le |L|} C_{ij}\left(P_{ij} - U_i^T L_j\right)^2 \\ & + \frac{\lambda}{2}\left(\|U\|_F^2 + \|L\|_F^2\right) \end{aligned} \tag{14}$$

where C_{ij} measures our confidence in observing M_{ij}, which can be calculated by Eq. 15. $P_{ij} = 1$, if $M_{ij} > 0$; otherwise $P_{ij} = 0$.

$$C_{ij} = 1 + \gamma \times M_{ij} \tag{15}$$

where γ is the ratio to balance the zero and non-zero elements in M.

In order to leverage the profiles of users and travel locations, we introduce user-user and travel location-travel location similarities to constrain the matrix factorization process, which is inspired by some former works [19, 27]. The objective function is transformed correspondingly, which is given by Eq. 16.

$$V = \min_{U,L} \frac{1}{2} \sum_{0 < i \le |U|, 0 < j \le |L|} C_{ij}\left(P_{ij} - U_i^T L_j\right)^2 + \frac{\lambda}{2}\left(\|U\|_F^2 + \|L\|_F^2\right)$$

$$+ \frac{\beta}{2}\left(\sum_{0 < i,j \le |U|} SimU\left(u_i, u_j\right)\|U_i - U_j\|_F^2 + \sum_{0 < i,j \le |L|} SimL\left(l_i, l_j\right)\|L_i - L_j\|_F^2\right) \tag{16}$$

where β is a parameter to control the significance of user-user and travel location-travel location similarities.

We update U and L alternately. Specifically, we compute the gradient of Eq. 16 with respect to U when fixing L to update U, which is given by Eq. 17. Similarly, we update L by Eq. 18:

$$U \leftarrow U + \alpha\left(C*\left(P - U^T L\right)L^T - \beta \sum_{0 < i,j \le |U|} SimU\left(u_i, u_j\right)\left(U_i - U_j\right) - \lambda U\right) \tag{17}$$

$$L \leftarrow L + \alpha U^T\left(C*\left(P - U^T L\right) - \beta \sum_{0 < i,j \le |L|} SimL\left(l_i, l_j\right)\left(L_i - L_j\right) - \lambda L\right) \tag{18}$$

where α is the learning rate, and $*$ denotes the bit-wise product.

4.6 Travel location recommendation

After minimizing Eq. 16, we can get optimized U and L. By multiplying matrices U and L, we can get the completed user-travel location matrix R, which can recover the missing values in M, and R_{ij} is the preference score of user u_i for travel location l_j.

Given target user u and query city c where the user wants to travel (i.e., the query is $q = (u, c)$), we first obtain the user's preference scores for all the travel locations in city c from the completed matrix R. Based on preference scores, we return top n travel locations as the results of the query.

5 Experiments

5.1 Dataset

We use the public API of Flickr to collect 736,383 geo-tagged photos that were taken in six cities in China between 1 January 2001 and 1 July 2011 [16]. We use the method introduced in Section 4.1 to find travel locations in these six cities. Table 1 shows the corresponding numbers of users and travel locations in each city. After using the method introduced in Section 4.2 to identify users' visits, we get 4386 visits for 882 users at 1514 travel locations, which results in a user-travel location matrix with 99.67% sparsity.

5.2 Experiment settings

In this section, we first give the settings of some important parameters.

1) The parameters of P-DBSCAN: We set $minPts = 50$ photos, $\varepsilon = 100$ m, and density ratio $\omega = 0.5$ for P-DBSCAN referring to [16]. An example of the clustering results in Hangzhou is shown in Fig. 4.
2) Visit duration threshold t_{thr}: We set $t_{thr} = 6$ hours referring to [16].

Table 1 The numbers of users and travel locations in each city

City	# Users	# Travel Locations
Beijing	294	414
Chengdu	68	49
Guangzhou	40	55
Hangzhou	133	94
Hongkong	210	419
Shanghai	314	483

3) The network structure of VAE: We construct VAE with Keras, and the network structure is shown in Table 2.
4) The parameters in matrix factorization: We vary the values of β, λ, and α over [0.0001, 0.0003, 0.001, 0.003, 0.01, 0.03, 0.1, 0.3], and find that the optimal values are $\beta = 0.0003$, $\lambda = 0.001$, and $\alpha = 0.3$. We tune parameters γ and d with experiments in Section 5.3.
5) Time threshold ΔT in visit sequence: We tune the parameter with experiments in Section 5.4.
6) Topic number k in topic model: We tune the parameter with experiments in Section 5.5.
7) Similarity weights: We vary the values of similarity weights over [0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9], and find that the optimal values are $w_1 = 0.5$, $w_2 = 0.1$, $w_3 = 0.4$, $w_4 = 0.2$, and $w_5 = 0.3$.

In the following experiments, we select users who have visited at least three cities. For each user, we sort the visited cities according to visit time. The data of the last visited city is chosen for testing, while the data of the second to last visited city is chosen to tune parameters, and the rest are used for training. According to the above dividing strategy, we split the dataset $\mathcal{D}$ into training set $\mathcal{D}_{train}$, validation set $\mathcal{D}_{valid}$, and test set $\mathcal{D}_{test}$.

To evaluate recommendation performance, we employ MAP@n as the performance metric, which is calculated by Eqs. 19–20:

$$\text{MAP}@n = \left(\sum_{i=1}^{m} \text{AP}@n \right) / m \tag{19}$$

$$\text{AP}@n = \left(\sum_{i=1}^{n} \left(\sum_{j=1}^{i} f_j \right) / i \right) / n \tag{20}$$

where n indicates the number of travel locations to be recommended, and m represents the number of target users. f_j is 1 if the user has actually visited the j th location in the returned results, otherwise f_j is 0.

5.3 The effects of ratio γ and latent variable number d in matrix factorization

Ratio γ controls the rate of confidence increase. If the value of γ is too small, the confidence levels of a same user's different visit frequencies cannot be well distinguished; on the contrary, the confidence levels may be extremely biased, as visiting a travel location can also be the result of factors different from preferring it [52]. Latent variable number d may affect the performance of the proposed method. If the value of d is too small, it would be difficult to make a clear distinction between users and travel locations; on the contrary, the computational complexity would increase dramatically. We use $\mathcal{D}_{train}$ to build models and evaluate on $\mathcal{D}_{valid}$ to study the effects of ratio γ and latent variable number d. Specifically, we increase
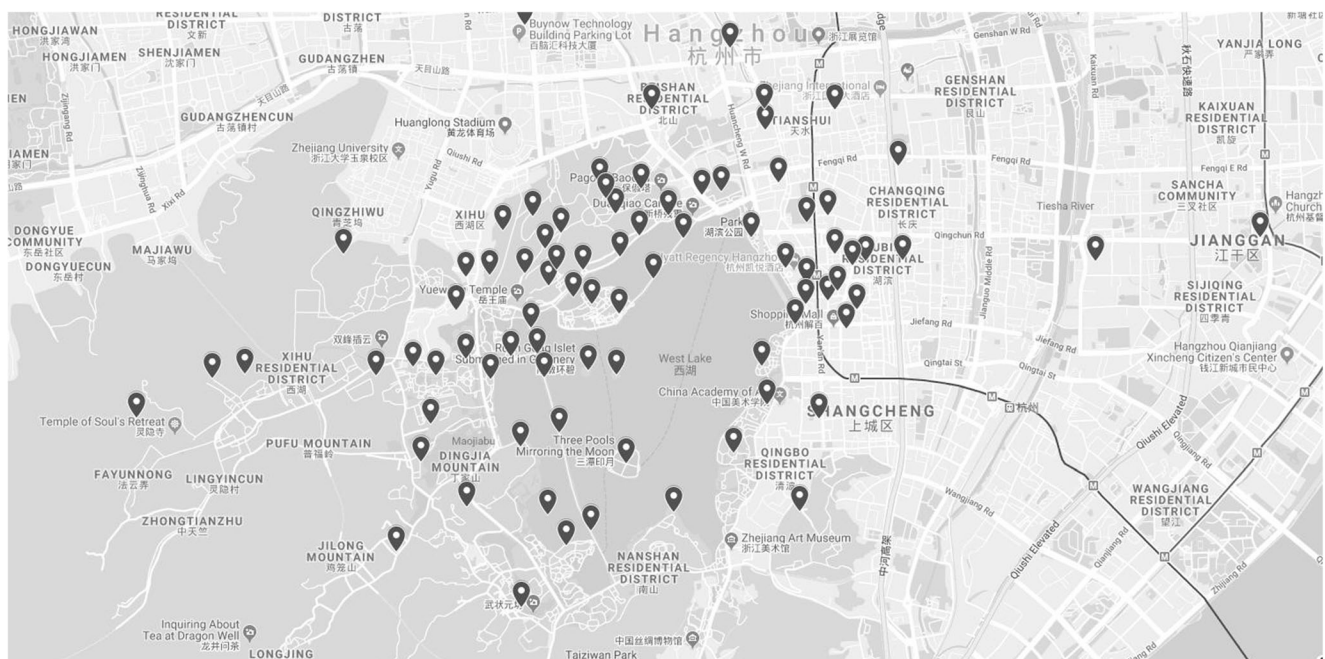
Fig. 4 The clustering results in Hangzhou

one parameter from 5 to 50 with a step size of 5 while fixing the other parameter to 5. The results are shown in Fig. 5 and Fig. 6, from which we can find that when γ is 15 and d is 25, MAP@5 and MAP@10 reach the highest. Therefore, we set $\gamma = 15$ and $d = 25$ in the following experiments.

Table 2 The network structure of VAE

Layer (type)	Output Shape	Connected to
input_1 (InputLayer)	(100, 32, 32, 3)	
conv2d_1 (Conv2D)	(100, 32, 32, 3)	input_1
conv2d_2 (Conv2D)	(100, 16, 16, 64)	conv2d_1
conv2d_3 (Conv2D)	(100, 16, 16, 64)	conv2d_2
conv2d_4 (Conv2D)	(100, 16, 16, 64)	conv2d_3
flatten_1 (Flatten)	(100, 16,384)	conv2d_4
dense_1 (Dense)	(100, 512)	flatten_1
dense_2 (Dense)	(100, 50)	dense_1
dense_3 (Dense)	(100, 50)	dense_1
lambda_1 (Lambda)	(100, 50)	dense_2 dense_3
dense_4 (Dense)	(100, 512)	lambda_1
dense_5 (Dense)	(100, 16,384)	dense_4
reshape_1 (Reshape)	(100, 16, 16, 64)	dense_5
conv2d_transpose_1 (Conv2DTranspose)	(100, 16, 16, 64)	reshape_1
conv2d_transpose_2 (Conv2DTranspose)	(100, 16, 16, 64)	conv2d_transpose_1
conv2d_transpose_3 (Conv2DTranspose)	(100, 33, 33, 64)	conv2d_transpose_2
conv2d_5 (Conv2D)	(100, 32, 32, 3)	conv2d_transpose_3
custom_variational_layer_1 (CustomVariationalLayer)	(100, 32, 32, 3), (100, 32, 32, 3)	input_1 conv2d_5

5.4 The effects of time threshold ΔT in visit sequence

Time threshold ΔT determines the length and amount of visit sequences. When ΔT is small, the extracted visit sequences are generally too short to obtaining sequential representations; while when ΔT takes a larger value, more noise would be introduced. We use $\mathcal{D}_{train}$ to build models and evaluate on $\mathcal{D}_{valid}$ to study the effects of time threshold ΔT. The results are shown in Table 3, from which we can find that MAP@5 and MAP@10 reach the highest when ΔT is 12. Therefore, ΔT is set to 12 in the following experiments.

5.5 The effects of topic number k in topic model

Topic number k determines the expressiveness of a topic model. When k is small, the extracted topics are not expressive enough to represent the documents; while when k takes a larger value, more noise would be introduced. We use $\mathcal{D}_{train}$ to build models and evaluate on $\mathcal{D}_{valid}$ to study the effects of topic number k. The results are shown in Fig. 7, from which we can find that MAP@5 and MAP@10 reach the highest when k is 20. Therefore, k is set to 20 in the following experiments.

5.6 The effects of different additional information

In order to study the effects of different additional information, we remove the visual, sequential, and textual similarities of users and travel locations, as well as co-visit probabilities, forming seven variants: WIND-MF-uvis, WIND-MF-useq,

Fig. 5 The MAP under different values of γ
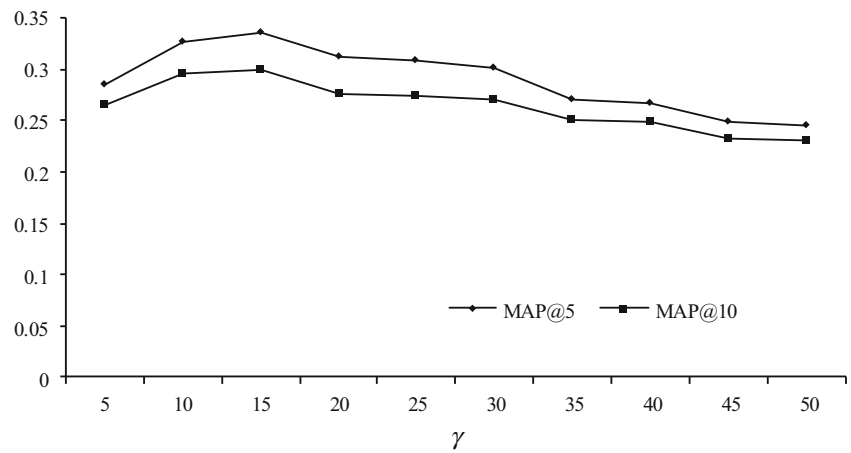
WIND-MF-utex, WIND-MF-lvis, WIND-MF-lseq, WIND-MF-ltex, and WIND-MF-dis.

We use $\mathcal{D}_{train}$ to build models and evaluate on $\mathcal{D}_{test}$. The parameters of the compared methods are well tuned to ensure fairness. The results are shown in Table 4, from which we can find:

1) Different additional information enhances recommendation performance to different degrees. According to influence degree, the sorting result of different additional information is: location visual effect > user visual effect > location sequential effect > user sequential effect > user textual effect > geographical effect > location textual effect.

2) For both users and travel locations, visual information plays the most important role, which indicates that uses like to visit travel locations with similar visual appearances [21]. Sequential information brings the second largest promotion on recommendation performance, as human movement usually exhibits strong sequential patterns [32].

3) Textual information is more important for users than for travel locations, which might be that the textual tags of geo-tagged photos are generated by users to express their feelings other than describe the travel locations.

4) Geographical information brings minor promotion on recommendation performance, which might be that most of the travel locations in a city are not far enough (76% pairs of travel locations in a city are less than 5 km apart) for users to take geographical distance information into account when making tradeoff decisions.

5) The performance of the proposed method is significantly higher than that of the seven variants, showing that considering visual, sequential, textual, and geographical information simultaneously can promote recommendation performance significantly.

5.7 The effects of the number of visited cities of the target user

In this section, we study the recommendation performance of the proposed method under different numbers of visited cities of the target user. The results are shown in Fig. 8, from which we can find that the more cities the target user
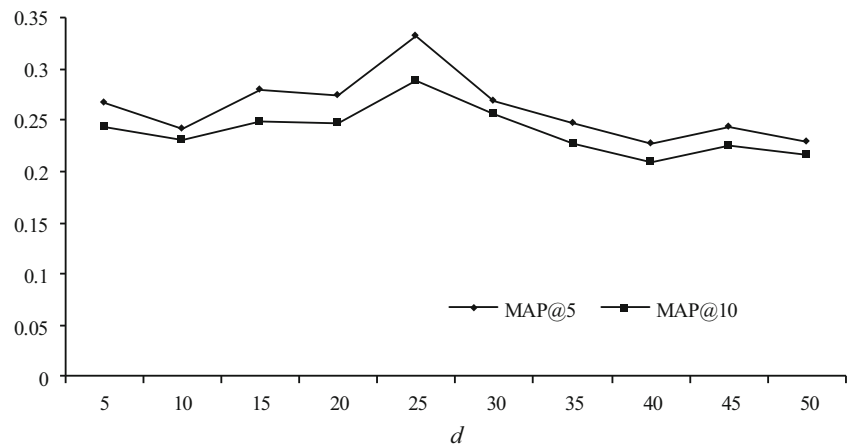
Fig. 6 The MAP under different values of d

Table 3 The MAP under different values of ΔT (mean ± standard deviation)

ΔT (hours)	MAP@5	MAP@10
6	0.260 ± 0.978	0.255 ± 0.943
12	**0.323 ± 0.603**	**0.318 ± 0.494**
24	0.260 ± 0.023	0.293 ± 0.636
48	0.143 ± 0.572	0.161 ± 0.304

Table 4 The effects of different additional information (mean ± standard deviation), * indicates WIND-MF is statistically superior to the compared method (pairwise *t*-test at a significance level of 5%)

Methods	MAP@5	MAP@10
WIND-MF-lvis	0.276 ± 0.087*	0.239 ± 0.065*
WIND-MF-uvis	0.292 ± 0.057*	0.255 ± 0.045*
WIND-MF-lseq	0.294 ± 0.064*	0.255 ± 0.048*
WIND-MF-useq	0.308 ± 0.054*	0.269 ± 0.042*
WIND-MF-utex	0.311 ± 0.050*	0.273 ± 0.034*
WIND-MF-dis	0.317 ± 0.040*	0.271 ± 0.031*
WIND-MF-ltex	0.319 ± 0.082*	0.276 ± 0.057*
WIND-MF	**0.336 ± 0.024**	**0.284 ± 0.043**

has visited, the higher recommendation performance the proposed method can offer.

5.8 The effects of the number of visited users in the Query City

In this section, we study the recommendation performance of the proposed method under different numbers of visited users in the query city. The results are shown in Fig. 9, from which we can find that the more users have visited the query city, the higher recommendation performance the proposed method can offer.

5.9 The comparison of different methods

In this section, the proposed method is compared with some state-of-the-art location recommendation methods. The compared methods are as follows:

1) Regularized Matrix Factorization based method (RMF) [51] is a baseline method based on matrix factorization without considering any additional information.
2) Dynamic Topic Model and Matrix Factorization based method (DTMMF) [27] firstly profiles users and travel locations as WIND-MF, and then concatenates visual, sequential, and textual representations to represent users and travel locations, on which user-user and travel

location-travel location similarities are calculated to constrain the factorization of user-travel location matrix.
3) Multi-Context Trajectory Embedding Model (MC-TEM) [25] embeds user-level, visual-level, topic-level, sequence-level, and location-level contexts into a shared low-dimension space, and recommends travel locations that are close to the target user.
4) Author Topic model-based Collaborative Filtering method (ATCF) [17] firstly profiles users as WIND-MF, and then concatenates visual, sequential, and textual representations to represent users. User-user similarities are calculated based on the concatenated representations, and then user based CF is exploited for recommendation.
5) Preference And Context Embedding (PACE) [24] leverages a deep neural network to model non-linear complex feature interactions between users and travel locations, while exploiting user-visual words, user-topic, location-location (co-visiting), location-location (geographical proximity), location-visual words, and location-topic context graphs to ensure that users or locations sharing more similar contexts will have closer embeddings.
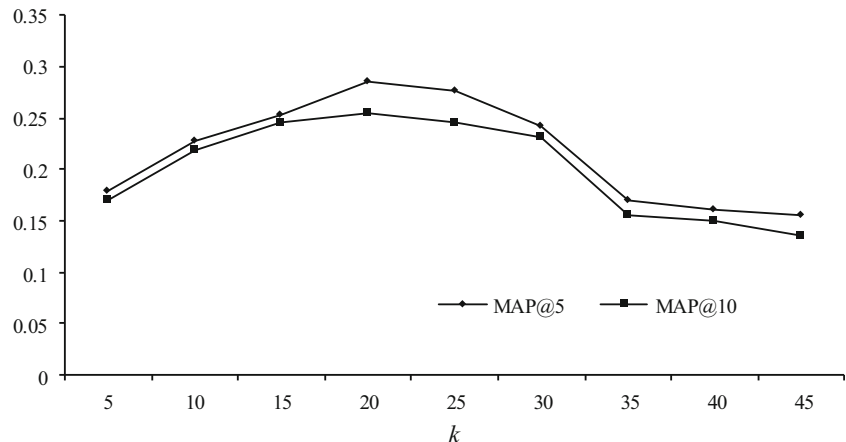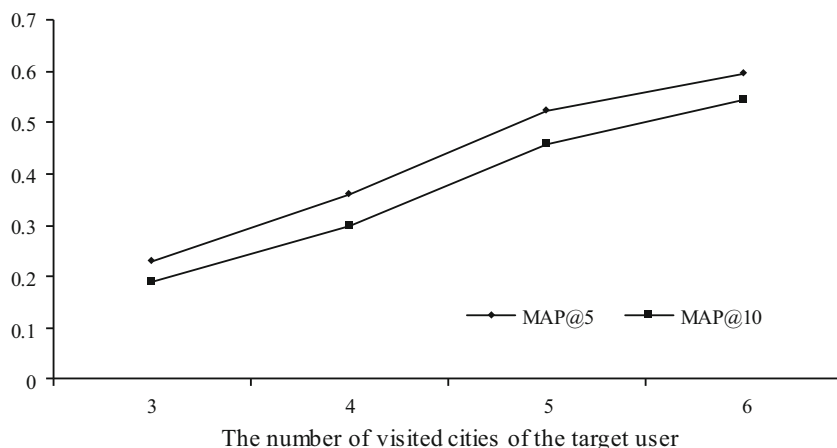6) RecNet [39] firstly factorizes co-visiting, geographical proximity, topic correlation, and visual correlation

Fig. 7 The MAP under different values of k

Fig. 8 The MAP under different numbers of visited cities of the target user

The number of visited cities of the target user

matrices to obtain the embeddings of travel locations, and then embeds users according to their visited locations. A deep neural network is finally leveraged to learn high-order feature interactions.

We use $\mathcal{D}_{train}$ to build models and evaluate on $\mathcal{D}_{test}$. The parameters of the compared methods are well tuned to ensure fairness. The results are shown in Table 5, from which we can find:

1) The recommendation performance of RMF is weaker than that of all the other methods, which might be that RMF infers users' preferences only by factorizing the sparse user-travel location matrix, without considering any additional information.

2) The recommendation performance of MC-TEM is better than that of ATCF, which might be that MC-TEM leverages additional information to embed both users and travel locations, while ATCF only leverages additional information to profile users.

3) The recommendation performance of PACE and RecNet is better than that of MC-TEM, which might

be that deep neural network based methods are able to learn high-order feature interactions. In addition, MC-TEM considers different additional information together by using a united conditional probability function, which may aggravate the sparsity problem.

4) The recommendation performance of RecNet is better than that of PACE, which might be that RecNet embeds users according to their visited locations, which can reduce the amount of parameters that are needed to train.

5) The recommendation performance of DTMMF is better than that of RecNet, which might be that DTMMF uses features extracted from additional information as priors to factorize the user-travel location matrix, which has fewer parameters compared to RecNet that uses a deep neural network to learn high-order feature interactions, thus can achieve better performance given a sparse user-travel location matrix.

6) The recommendation performance of WIND-MF is better than that of DTMMF, which might be that WIND-MF can capture the significance of different additional information by assigning different weights, while DTMMF assigns a same weight to different additional information.

Fig. 9 The MAP under different numbers of visited users in the query city

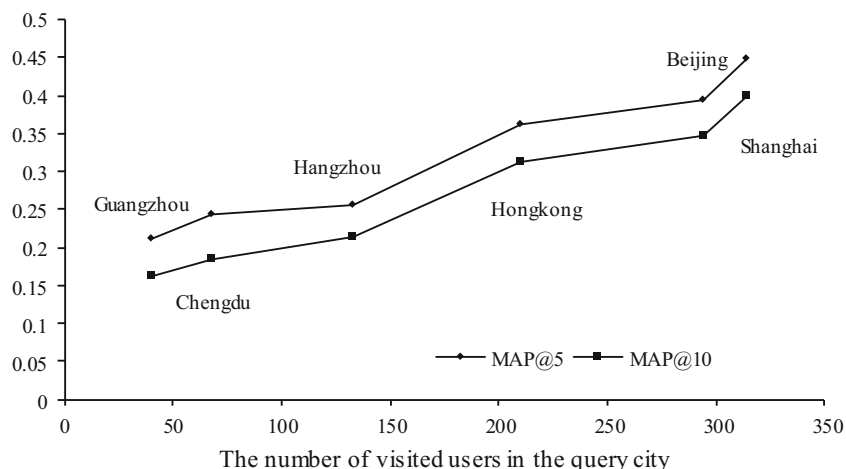The number of visited users in the query city

Table 5 The comparison of different location recommendation methods (mean ± standard deviation), * indicates WIND-MF is statistically superior to the compared method (pairwise *t*-test at a significance level of 5%)

Methods	MAP@5	MAP@10
RMF	0.111 ± 0.022*	0.086 ± 0.045*
ATCF	0.271 ± 0.054*	0.174 ± 0.024*
MC-TEM	0.237 ± 0.091*	0.210 ± 0.032*
PACE	0.256 ± 0.047*	0.217 ± 0.067*
RecNet	0.259 ± 0.096*	0.225 ± 0.081*
DTMMF	0.316 ± 0.023*	0.264 ± 0.047*
WIND-MF	**0.336 ± 0.024**	**0.284 ± 0.043**

6 Conclusions and future work

In this paper, we propose WIND-MF for personalized travel location recommendation based on geo-tagged photos. The method profiles users and travel locations based on the visual, sequential, and textual information of geo-tagged photos, and models co-visit probabilities based on geographical distances. Visual, sequential, and textual similarities as well as co-visit probabilities are assigned with different weights, to obtain weighted user-user and travel location-travel location similarities, which are then used as regularization terms to constrain the factorization of user-travel location matrix. The experiment results show the superiority of the proposed method. We also find that visual and sequential information contributes most to improving recommendation performance.

The proposed method also has space to further expand. We intend to introduce users' preferences to guide visual feature extracting from photos. In addition, more context information (e.g., time, weather, and traffic) can be introduced to make the proposed method to be context-aware.

References

1. Zheng YT, Zha ZJ, Chua TS (2011) Research and applications on georeferenced multimedia: a survey. Multimed Tools Appl 51(1):77–98
2. Zheng Y, Zhang L, Xie X, Ma WY (2009) Mining interesting locations and travel sequences from GPS trajectories. Proceedings of the International Conference on World Wide Web: 791–800
3. Zheng Y, Zhang L, Ma Z, Xie X, Ma WY (2011) Recommending friends and locations based on individual location history. ACM Trans Web 5(1):5
4. Zheng Y, Xie X (2011) Learning travel recommendations from user-generated GPS traces. ACM Trans Intell Syst Technol 2(1):2
5. Ye M, Yin P, Lee WC, Lee DL (2011) Exploiting geographical influence for collaborative point-of-interest recommendation. Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval: 325–334
6. Hsieh HP, Li CT, Lin SD (2012) Exploiting large-scale check-in data to recommend time-sensitive routes. Proceedings of the ACM SIGKDD International Workshop on Urban Computing: 55–62
7. Gao H, Tang J, Hu X, Liu H (2015) Content-aware point of interest recommendation on location-based social networks. Proceedings of the AAAI conference on artificial intelligence: 1721–1727
8. Ying Y, Chen L, Chen G (2017) A temporal-aware POI recommendation system using context-aware tensor decomposition and weighted HITS. Neurocomputing 242:195–205
9. Clements M, Serdyukov P, De Vries AP, Reinders MJ (2010) Using flickr geotags to predict user travel behaviour. In: Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval: 851–852
10. Mamei M, Rosi A, Zambonelli F (2010) Automatic analysis of geotagged photos for intelligent tourist services. Proceedings of the International Conference on Intelligent Environments: 146–151
11. Popescu A, Grefenstette G (2011) Mining social media to create personalized recommendations for tourist visits. Proceedings of the international conference on computing for Geospatial Research & Applications: 37
12. Yang C, Sun M, Zhao WX, Liu Z, Chang EY (2017) A neural network approach to jointly modeling social networks and mobile trajectories. ACM Trans Inf Syst 35(4):36
13. Liu X, Liu Y, Li X (2016) Exploring the context of locations for personalized location recommendations. Proceedings of the international joint conference on artificial intelligence: 1188–1194
14. Kurashima T, Iwata T, Irie G, Fujimura K (2013) Travel route recommendation using geotagged photos. Knowl Inf Syst 37(1):37–60
15. Majid A, Chen L, Mirza HT, Hussain I, Chen G (2015) A system for mining interesting tourist locations and travel sequences from public geo-tagged photos. Data Knowl Eng 95:66–86
16. Majid A, Chen L, Chen G, Mirza HT, Hussain I, Woodward J (2013) A context-aware personalized travel recommendation system based on geotagged social media data mining. Int J Geogr Inf Sci 27(4):662–684
17. Jiang S, Qian X, Shen J, Fu Y, Mei T (2015) Author topic model-based collaborative filtering for personalized POI recommendations. IEEE Trans Multimed 17(6):907–918
18. Yuan Q, Cong G, Ma Z, Sun A, Thalmann NM (2013) Time-aware point-of-interest recommendation. Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval: 363–372
19. Cheng C, Yang H, King I, Lyu MR (2012) Fused matrix factorization with geographical and social influence in location-based social networks. Proceedings of the AAAI conference on artificial intelligence: 17–23
20. Zhao P, Xu X, Liu Y, Sheng VS, Zheng K, Xiong H (2017) Photo2trip: exploiting visual contents in geo-tagged photos for personalized tour recommendation. Proceedings of the ACM International Conference on Multimedia: 916–924
21. Wang S, Wang Y, Tang J, Shu K, Ranganath S, Liu H (2017) What your images reveal: exploiting visual contents for point-of-interest recommendation. Proceedings of the International Conference on World Wide Web: 391–400
22. Cheng AJ, Chen YY, Huang YT, Hsu WH, Liao HYM (2011) Personalized travel recommendation by mining people attributes

from community-contributed photos. Proceedings of the ACM International Conference on Multimedia: 83–92

23. Chen YY, Cheng AJ, Hsu WH (2013) Travel recommendation by mining people attributes and travel group types from community-contributed photos. IEEE Trans Multimed 15(6):1283–1295

24. Yang C, Bai L, Zhang C, Yuan Q, Han J (2017) Bridging collaborative filtering and semi-supervised learning: a neural approach for POI recommendation. Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining 1245–1254

25. Zhou N, Zhao WX, Zhang X, Wen JR, Wang S (2016) A general multi-context embedding model for mining human trajectory data. IEEE Trans Knowl Data Eng 28(8):1945–1958

26. Cai L, Xu J, Liu J, Pei T (2018) Integrating spatial and temporal contexts into a factorization model for POI recommendation. Int J Geogr Inf Sci 32(3):524–546

27. Xu Z, Chen L, Dai Y, Chen G (2017) A dynamic topic model and matrix factorization-based travel recommendation method exploiting ubiquitous data. IEEE Trans Multimed 19(8):1933–1945

28. Zheng VW, Zheng Y, Xie X, Yang Q (2010) Collaborative location and activity recommendations with GPS history data. Proceedings of the International Conference on World Wide Web: 1029–1038

29. Bao J, Zheng Y, Wilkie D, Mokbel M (2015) Recommendations in location-based social networks: a survey. GeoInformatica 19(3):525–565

30. Gao H, Tang J, Liu H (2012) Exploring social-historical ties on location-based social networks. Proceedings of the international AAAI conference on weblogs and social media: 114–121

31. Long X, Joshi J (2013) A HITS-based POI recommendation algorithm for location-based social networks. Proceedings of the International Conference on Advances in Social Networks Analysis and Mining: 642–647

32. Cheng C, Yang H, Lyu MR, King I (2013) Where you like to go next: successive point-of-interest recommendation. Proceedings of the international joint conference on artificial intelligence: 2605–2611

33. Le Q, Mikolov T (2014) Distributed representations of sentences and documents. Proceedings of the international conference on machine learning: 1188–1196

34. Ye M, Shou D, Lee WC, Yin P, Janowicz K (2011) On the semantic annotation of places in location-based social networks. Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining: 520–528

35. Memon I, Chen L, Majid A, Lv M, Hussain I, Chen G (2015) Travel recommendation using geo-tagged photos in social media for tourist. Wirel Pers Commun 80(4):1347–1362

36. Blei DM, Ng AY, Jordan MI (2003) Latent dirichlet allocation. J Mach Learn Res 3(Jan):993–1022

37. Zhang F, Yuan NJ, Lian D, Xie X, Ma WY (2016) Collaborative knowledge base embedding for recommender systems. Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining: 353–362

38. Kingma DP, Welling M (2013) Auto-encoding variational bayes. arXiv preprint arXiv: 13126114

39. Ding R, Chen Z (2018) Recnet: a deep neural network for personalized POI recommendation in location-based social networks. Int J Geogr Inf Sci 32(8):1631–1648

40. Kurashima T, Iwata T, Irie G, Fujimura K (2010) Travel route recommendation using geotags in photo sharing sites. Proceedings of the ACM International Conference on Information and Knowledge Management: 579–588

41. Zheng YT, Zha ZJ, Chua TS (2012) Mining travel patterns from geotagged photos. ACM Trans Intell Syst Technol 3(3):56

42. Kisilevich S, Mansmann F, Keim D (2010) P-DBSCAN: a density based clustering algorithm for exploration and analysis of attractive areas using collections of geo-tagged photos. Proceedings of the international conference and exhibition on computing for Geospatial Research & Application: 38

43. Xu Z, Chen L, Chen G (2015) Topic based context-aware travel recommendation method exploiting geotagged photos. Neurocomputing 155:99–107

44. Mikolov T, Sutskever I, Chen K, Corrado GS, Dean J (2013) Distributed representations of words and phrases and their compositionality. Proceedings of the neural information processing systems conference: 3111–3119

45. Papadimitriou CH, Raghavan P, Tamaki H, Vempala S (2000) Latent semantic indexing: a probabilistic analysis. J Comput Syst Sci 61(2):217–235

46. Grbovic M, Radosavljevic V, Djuric N, Bhamidipati N, Savla J, Bhagwan V, Sharp D (2015) E-commerce in your inbox: product recommendations at scale. Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining: 1809–1818

47. Isinkaye F, Folajimi Y, Ojokoh B (2015) Recommendation systems: principles, methods and evaluation. Egyptian Inform J 16(3):261–273

48. Bokde D, Girase S, Mukhopadhyay D (2015) Matrix factorization model in collaborative filtering algorithms: a survey. Proc Comput Sci 49:136–146

49. Yu Y, Wang C, Wang H, Gao Y (2017) Attributes coupling based matrix factorization for item recommendation. Appl Intell 46(3):521–533

50. Xing S, Liu F, Zhao X, Li T (2018) Points-of-interest recommendation based on convolution matrix factorization. Appl Intell 48(8):2458–2469

51. Koren Y, Bell R, Volinsky C (2009) Matrix factorization techniques for recommender systems. Computer (8):30–37

52. Hu Y, Koren Y, Volinsky C (2008) Collaborative filtering for implicit feedback datasets. Proceedings of the IEEE International Conference on Data Mining: 263–272

Dandan Lyu received the B.S. degree in software engineering from Nanjing University of Science and Technology, China, in 2016. She is currently a Ph.D. candidate in the College of Computer Science and Technology at Zhejiang University, China. Her research interests include recommender systems and data mining.

Ling Chen received the B.S. and Ph.D. degrees in computer science from Zhejiang University in 1999 and 2004, respectively. He is currently an associate professor with the College of Computer Science and Technology, Zhejiang University. His research interests include ubiquitous computing, AI, pattern recognition, and human computer interaction.

Shanshan Yu received the B.S. degree in computer science and technology from Xidian University, China, in 2017. She is currently a master candidate in the College of Computer Science and Technology at Zhejiang University, China. Her research interests include recommender systems and data mining.

Zhenxing Xu received the B.S. degree in computer science and technology from Chaohu University, China, in 2008, the M.S. degree in technology of computer application from Zhejiang Sci-Tech University, China, in 2012, and the Ph.D. degree in computer science and technology from Zhejiang University, China, in 2017. His research interests include recommender systems and data mining.