Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence (IJCAI 2015)

Personalized Ranking Metric Embedding for Next New POI Recommendation

Shanshan Feng,[1] **Xutao Li,**[2] **Yifeng Zeng,**[3] **Gao Cong,**[2] **Yeow Meng Chee,**[4] **Quan Yuan**[2]

[1]Interdisciplinary Graduate School,
Nanyang Technological University, Singapore, sfeng003@e.ntu.edu.sg
[2]School of Computer Engineering, Nanyang Technological University, Singapore,
{lixutao@, gaocong@, qyuan1@e.}ntu.edu.sg
[3]School of Computing, Teesside University, UK, Y.Zeng@tees.ac.uk
[4]School of Physical and Mathematical Sciences,
Nanyang Technological University, Singapore, ymchee@ntu.edu.sg

Abstract

The rapidly growing of Location-based Social Networks (LBSNs) provides a vast amount of check-in data, which enables many services, e.g., point-of-interest (POI) recommendation. In this paper, we study the *next new* POI recommendation problem in which *new* POIs with respect to users' current location are to be recommended. The challenge lies in the difficulty in precisely learning users' sequential information and personalizing the recommendation model. To this end, we resort to the Metric Embedding method for the recommendation, which avoids drawbacks of the Matrix Factorization technique. We propose a personalized ranking metric embedding method (PRME) to model personalized check-in sequences. We further develop a PRME-G model, which integrates sequential information, individual preference, and geographical influence, to improve the recommendation performance. Experiments on two real-world LBSN datasets demonstrate that our new algorithm outperforms the state-of-the-art next POI recommendation methods.

1 Introduction

With the increasing popularity of location-based social networks (LBSNs), users would like to share their locations by checking-in points-of-interest (POIs). The large amount of check-in data offers opportunity to better understand users' mobility behavior, based on which recommending POIs will become valuable. POI recommendation is of great value to help users explore their surroundings. Importance of POI recommendation has attracted a significant amount of research interest on developing recommendation techniques [Cho *et al.*, 2011; Ye *et al.*, 2011; Cheng *et al.*, 2012; Yuan *et al.*, 2013; Lian *et al.*, 2014; Li *et al.*, 2015].

Compared to the POI recommendation, the *next* POI recommendation [Cheng *et al.*, 2013] has received relatively little research attention. Besides users' preference, the next POI recommendation additionally considers the sequential information of users' check-ins. The sequential behavior is important for POI recommendation because human movement exhibits sequential patterns [Ye *et al.*, 2013]. We verify users' sequential behavior in the analysis of two real-world datasets. Meanwhile, we observe that users often visit *new* POIs that they have not been visited before. In this paper, we focus on the *Next New* POI recommendation problem (simplified as N^2-POI recommendation), which is to recommend *new* POIs to be visited *next* given a user's current location.

The challenge of N^2-POI recommendation is to learn transitions of users' check-ins that are commonly represented by a first-order Markov chain model. Due to the sparse transition data, it is difficult to estimate the transition probability in Markov chain, especially for the unobserved transition. Factorized Personalized Markov Chain (FPMC) [Rendle *et al.*, 2010] method has been used to calculate the item transitions. FPMC exploits matrix factorization technique to factorize the Markov transition matrix. To model the transition, FPMC represents each item with two independent vectors. However, these two vectors are related to the same item and their latent relationship is not exploited (more details in Section 2). Consequently, this technique is not sufficiently effective to learn the item transitions.

By projecting every POI into one object in a low-dimensional Euclidean latent space, we use the Metric Embedding algorithm to effectively compute the location transition in a Markov chain model. Intuitively, the distance of two objects measures the strength of their sequential relation. We further propose a pair-wise ranking Metric Embedding algorithm that ranks potential POIs in a latent space. Subsequently, we develop a Personalized Ranking Metric Embedding model (PRME), which jointly models the sequential information and individual preference. Since users incline to visit the POIs close to their current positions, geographical influence is important for the recommendation task. We extend the PRME model to accommodate geographical influence in the N^2-POI recommendation. We summarize the main contributions of this paper as follows.

- We develop a pair-wise Metric Embedding algorithm to model the sequential transition of POIs. To the best of our knowledge, this is the first work that uses the Metric Embedding method for the POI recommendation.

- To model the personalized sequential information, we propose a novel PRME algorithm , which jointly considers sequential transition and user preference. We further develop a PRME-G algorithm to incorporate geographical influence for the N^2-POI recommendation problem.
- We conduct comprehensive experiments by comparing our algorithms with the state-of-the-art techniques over two real-world datasets.

2 Related Work

Location recommendation has attracted intensive research attention recently. Most of previous methods are based on Collaborative Filtering (CF) technique. One of the most popular CF algorithms is the user-based CF [Ye *et al.*, 2011; Yuan *et al.*, 2013; Chen *et al.*, 2015], which takes advantage of check-ins of similar users for the recommendation purpose. Another CF method is Matrix Factorization [Cheng *et al.*, 2012; Lian *et al.*, 2014], which learns the general taste of a user by factorizing the observed user-item preference matrix. The CF based algorithms mainly exploit the user preference to make recommendations. Currently, geographical influence has been fused with the CF algorithms to enhance POI recommendation. For example, Gaussian Mixture distribution[Cho *et al.*, 2011; Cheng *et al.*, 2012] and power law distribution [Ye *et al.*, 2011; Yuan *et al.*, 2013] have been proposed to model the geographical influence.

Sequential influence has been considered for the POI recommendation. Most of the studies utilize the Markov chain property to predict the next check-ins. Zhang *et al.* [Zhang *et al.*, 2014] predict the sequential probability through an additive Markov chain. However, this method fails to assign the transition probability for the unobserved data because it directly constructs the Markov chain model based on check-in data. Instead of using transition pattern of POIs, [Liu *et al.*, 2013; Ye *et al.*, 2013] exploit the transition pattern of POI categories to predict future check-ins. However, the accuracy of these methods highly depends on the category information.

FPMC [Rendle *et al.*, 2010] is the state-of-the-art personalized Markov chain algorithm. To model the Markov chain transition, FPMC associates each item l with two independent vectors: $\vec{F_l}$ to embody the transition to other items, and $\vec{T_l}$ to represent getting transition from other items. The transition from item l_i to another item l_j is embodied as the inner product of the latent vectors $\vec{F_{l_i}} \cdot \vec{T_{l_j}}$. However, FPMC fails to model relations among multiple items. For example, given $l_i \rightarrow l_j$ and $l_j \rightarrow l_k$, transition $l_i \rightarrow l_k$ is expected to have a high probability because both l_i and l_k have close relations with l_j. However, FPMC fails to reflect such relationship due to the independent assumption on $\vec{F_{l_j}}$ and $\vec{T_{l_j}}$. This drawback limits its performance.

The work by Cheng *et al.* [Cheng *et al.*, 2013] is most related to ours. The research aims to recommend POIs for the next hours by merging consecutive check-ins in the previous hours. It employs FPMC to model the personalized POI transition. Based on the current POI, their method only considers the POIs in the defined region as candidates. Lian *et al.* [Lian *et al.*, 2013] also adopts FPMC to represent the short-term and long-term preference to predict the next check-in. Differently from these studies, we propose a metric embedding model to learn the personalized sequential information.

Embedding items in a low-dimension Euclidean space is mainly used for the purpose of visualization and exploratory data analysis [Roweis and Saul, 2000; Hinton and Roweis, 2002]. Recently Metric Embedding is adopted in the music playlist prediction. Chen *et al.* [Chen *et al.*, 2012; 2013] propose a Logistic Markov embedding (LME) for generating the playlists. LME maps each song to one point (or multiple points) in a latent Euclidean space. The transition probability from one song to another is related to the Euclidean distance of the two songs in the latent space. The research [Wu *et al.*, 2013; Moore *et al.*, 2013] embeds users and songs into a common latent space to represent the personalized Markov Chain. Our work differs from the aforementioned studies in that we exploit pair-wise ranking scheme to learn the parameters, and adapt Metric Embedding for the POI recommendation task by incorporating multiple factors.

3 Next New POI Recommendation

We use two publicly available datasets. The first dataset is the *FourSquare* check-ins within Singapore [Yuan *et al.*, 2013] while the second one is the *Gowalla* check-ins dataset within California and Nevada [Cho *et al.*, 2011]. We use one-year data in both datasets. Each check-in is a tuple in the form of $< user, POI, time >$. Each POI is associated with the latitude and longitude. We remove users who have check-in fewer than 10 POIs, and POIs which have been visited by fewer than 10 users. The basic statistics of datasets used in this paper are summarized in Table 1.

Dataset	#User	#POI	#Check-in	Time range
FourSquare	1917	2675	155365	08/2010-07/2011
Gowalla	4996	6871	245157	11/2009-10/2010

Table 1: Statistics of two datasets

3.1 Observations on real-world datasets

Observation 1: exploration of new locations

Figure 1 shows the average ratio of new POIs over all users on two datasets for every 50 days. For example, the ratio at the 100^{th} day is the proportion of POIs visited at the 100^{th} day that have not been visited in the previous days. The ratio of new POIs is pretty high (most of the ratios above 0.4) on both datasets, which implies that people always like to explore new POIs. This observation is in accordance with the recent findings [Lian *et al.*, 2013].

Observation 2: temporal influence

Figure 2(a) shows the cumulative distribution function (CDF) of the time difference of two sequential check-ins. Figure 2(a) demonstrates that more than 50% successive check-ins occur in less than 24 hours. Meanwhile, many consecutive check-ins occur in a longer time. For more than 25% consecutive check-ins, their time differences are larger than 48 hours.

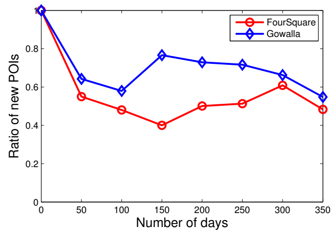
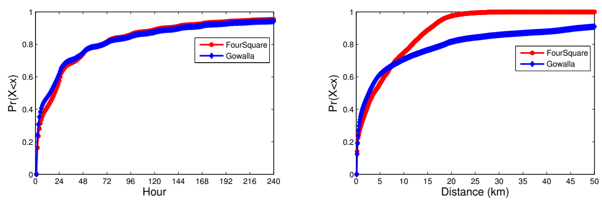


Figure 1: The average ratio of new POIs with number of days



(a) Time difference (b) Geographical distance

Figure 2: CDF of time difference and geographical distance of two consecutive check-ins.

Observation 3: spatial influence

We compute the geographical distance of two consecutive check-ins and plot the CDF distribution in Figure 2(b), which shows that 70% consecutive check-ins have less than $10km$ in both datasets. The CDF curve increases fast when distance is small, which implies that most check-ins occur in nearby areas. This result indicates that users' next movements are influenced by their current locations. The finding is in accordance with the reported result [Yuan *et al.*, 2013; Cheng *et al.*, 2013].

3.2 Next new POI problem definition

When two check-ins occur in a short time period, markov chain property exists [Cheng *et al.*, 2013]. This motivates us to consider the POI transition within a short period, which means the next POI is influenced by current POI. Following [Zhang *et al.*, 2014], if the time difference of two consecutive check-ins is smaller than τ, sequential influence shall be considered. Here τ is the time threshold. In Section 6, we investigate the impact of τ. Meanwhile, according to observation 1, users incline to visit new POIs for their exploration interests, indicating that the problem of suggesting new POIs for users is meaningful.

Based on the sequential property within a short time period and users' willingness of new POIs, we formally define the N^2-POI recommendation problem below.

Definition 1 ***(N^2-POI Recommendation Problem)*** *For a set of users $\mathcal{U}$ and a set of POIs $\mathcal{L}$, $\mathcal{C}$ is the historical check-in data, and $\mathcal{L}^u$ is the set of POIs that user u has visited before. Given current POI l^c of user u, the N^2-POI problem is to recommend a set of POIs $\mathcal{S}^{u,l^c} \subset \mathcal{L}$ for the user u to visit next and the POIs are new to the user*

$$\mathcal{S}^{u,l^c} = \{l \in \mathcal{L} \setminus \mathcal{L}^u\}$$

Note that in the N^2-POI recommendation problem we only recommend *new* POIs to users. In contrast, a next POI prediction task outputs both the visited and un-visited POIs as the results. As reported in [Lian *et al.*, 2013], if we only use the frequently visited POIs of a user to predict his next POIs, a high prediction precision can be achieved. In addition, it's challenging to estimate the implicit transition probability of potential new POIs based on the sparse historical data. Thus, the N^2-POI recommendation problem is harder than the next POI prediction problem.

4 Personalized Ranking Metric Embedding

We first introduce Metric Embedding algorithm with Pairwise Ranking to model location transitions in Section 4.1. The Personalized Ranking Metric Embedding (PRME) is then presented in Section 4.2. Section 4.3 states PRME-G model, which jointly incorporates sequential transition, user preference, and geographical influence.

4.1 Pairwise Ranking Metric Embedding

Embedding POIs in latent space

To model the sequential information, we need learn the transition probability in a Markov chain model. However, due to the data sparsity, it is infeasible to estimate the transitions by using standard counting methods. Metric embedding model can be used to handle the data sparsity and be generalized to the unobserved data. We represent each POI as one point in a latent space. We assume that Euclidean distance between POIs in the latent space reflects the transition probability. The larger the distance, the lower the strength of transitions. With all POIs embedded in a latent space, our model estimates the sensible transition probabilities of POIs. It is also possible to assign meaningful probabilities to those unobserved transitions.

In the Metric Embedding model, each POI l has a position $X(l)$ in a K-dimensional space. Given the observed sequential POI transitions, the goal is to learn the positions of all POIs. We relate the transition probability of a pair, l_i and l_j, to the Euclidean distance as defined in Eq. 1.

$$\hat{P}(l_j|l_i) = \frac{e^{-||X(l_j)-X(l_i)||^2}}{Z(l_i)} \tag{1}$$

where $||X(l_i) - X(l_j)||^2 = \sum_{k=1}^{K}(X_k(l_i) - X_k(l_j))^2$, K is the number of dimensions of the latent space and $Z(l_i) = \sum_{n=1}^{|L|} e^{-||X(l_n)-X(l_i)||^2}$ is the normalization term.

Compared to matrix factorization in FPMC, Metric Embedding (ME) can better model the sequential transition. It represents each item as a single point in a latent space rather than two independent vectors. This representation is more natural to embody the latent relations of items. For example, given observed transition $l_i \rightarrow l_j$ and $l_j \rightarrow l_k$, $l_i \rightarrow l_k$ is expected to be a potential transition. ME is able to capture this kind of relation. $||X(l_i) - X(l_k)||^2$ would be small, because both $X(l_i)$ and $X(l_k)$ will be pulled closely to a same position $X(l_j)$.

Ranking based metric embedding

Eq. 1 only exploits the observed check-ins to learn the latent position of each POI. Since the observed data is very sparse, we learn the latent position by fitting the rankings for the POI transition. Consequently, we can additionally make use of the unobserved data to learn the parameters. We utilize POI pairs as training data and optimally estimate the pair-wise ranking. We assume that the observed next POI is more related to current POI than the unobserved POI. For example, if transition $l^c \to l_i$ is observed and $l^c \to l_j$ is not observed, POI l_i should be ranked higher than POI l_j. We model it as a ranking $>$ over POIs:

$$l_i >_{l^c} l_j \Leftrightarrow \hat{P}(l_i|l^c) > \hat{P}(l_j|l^c) \tag{2}$$

The goal of POI recommendation is to provide a ranking of all the items, and accordingly recommend the top items. Furthermore, since we are interested in the ranking of POIs, we can simplify the computation by only keeping the Euclidean distance $||(X(l_i) - X(l_j)||^2$ in the latent space (abbreviated as $\mathcal{D}_{l_i,l_j}$). Instead of utilizing the exponential function, we use the Euclidean distance to rank the POIs.

$$\begin{aligned}\hat{P}(l_i|l^c) > \hat{P}(l_j|l^c) &\Rightarrow e^{-||X(l_i)-X(l^c)||^2} > e^{-||X(l_j)-X(l^c)||^2} \\ &\Rightarrow ||X(l_i) - X(l^c)||^2 < ||X(l_j) - X(l^c)||^2 \\ &\Rightarrow \mathcal{D}_{l^c,l_j} - \mathcal{D}_{l^c,l_i} > 0\end{aligned} \tag{3}$$

4.2 Personalized Ranking Metric Embedding

Individual preference has been proved to be an important factor for the POI recommendation because each user would prefer some favorite POIs. Given current location l^c of user u, the recommended N^2-POI shall not only be related to l^c, but also capture user's preference.

We use ME to model the user-item preference. We project each POI and each user into a latent space. The distance between a user and a POI reflects the strength of their relations. Intuitively, if user u likes POI l, the distance $||X(u)-X(l)||^2$ in latent space should be small. Otherwise, $||X(u) - X(l)||^2$ would be large. By doing this, we model the user-item preference in a latent low-dimension space.

Since RME exploits transition data of all users, it does not reflect the user-specific transition. We further develop a Personalized Ranking Metric Embedding (PRME) method, which considers sequential information and individual preference together. We utilize two latent spaces: one is the *sequential transition* space and the other is the *user preference* space. Each POI l has one latent position $X^S(l)$ in the sequential transition space, and the Euclidean distance of two POIs l_i and l_j is defined as $\mathcal{D}^S_{l_i,l_j} = ||X^S(l_i) - X^S(l_j)||^2$. In the user preference space, each user u has a latent position $X^P(u)$ and each POI l has a latent position $X^P(l)$. $\mathcal{D}^P_{u,l} = ||X^P(u) - X^P(l)||^2$ denotes the Euclidean distance of user u and POI l in the user preference space.

Given current location l^c of user u, we model personalized sequential transition for a candidate POI l by combining the two kinds of metric. As two components contribute differently into the metric score of POI l, we use a linear interpolation to weight the two metrics.

$$\mathcal{D}_{u,l^c,l} = \alpha\mathcal{D}^P_{u,l} + (1-\alpha)\mathcal{D}^S_{l^c,l} \tag{4}$$

where $\alpha \in [0, 1]$ controls the weight of different kinds of distance.

Based on observation 2 (in Section 3.1), some successive check-ins have large time difference, which may indicate their irrelevance. We assume that if the time interval between two adjacent check-ins is larger than threshold τ, only the user preference is considered. We then recompute the distance metric $\mathcal{D}_{u,l,l^c}$ below.

$$\mathcal{D}_{u,l^c,l} = \begin{cases}\mathcal{D}^P_{u,l} & \text{if } \Delta(l,l^c) > \tau \\ \alpha\mathcal{D}^P_{u,l} + (1-\alpha)\mathcal{D}^S_{l^c,l} & \text{otherwise}\end{cases} \tag{5}$$

where $\Delta(l, l^c)$ is the time difference of two successive check-ins l and l^c.

4.3 Incorporating Geographical Influence

As stated in Section 3.1, given current location, users would like to visit the near POIs rather than the far way POIs. Geographical distance affects users' visiting behavior. Thus we propose the PRME-G model to incorporate the geographical influence into the PRME model.

We accommodate the geographical influence by using the weight of the geographical distance. For any pair of POIs, we can calculate the geographical distance via their longitude and latitude. Given the geographical distance $d_{l^c,l}$, we use a weight function $w_{l^c,l}$. In this paper, $w_{l^c,l} = (1 + d_{l^c,l})^{0.25}$. The fused distance metric becomes $\mathcal{D}_{u,l^c,l} \cdot w_{l^c,l}$. When the distance of a POI is large, the fused metric would be large, and thus this POI is less likely to be recommended. The integrated metric with geographical influence is defined as below.

$$\mathcal{D}^G_{u,l^c,l} = \begin{cases}\mathcal{D}^P_{u,l} & \text{if } \Delta(l,l^c) > \tau \\ w_{l^c,l} \cdot \left(\alpha\mathcal{D}^P_{u,l} + (1-\alpha)\mathcal{D}^S_{l^c,l}\right) & \text{otherwise}\end{cases} \tag{6}$$

Note that the PRME-G model is not special for N^2-POI recommendation and is general for solving next POI prediction problem. We do not show the improvement for predicting next POI due to the limited space.

5 Parameter Learning

5.1 Optimization Criterion

Assuming that users' check-ins are independent with each other, we can derive the optimization criterion of PRME model. Analogous to Bayesian Personalized Ranking (BPR) approach [Rendle *et al.*, 2009], we estimate the PRME model by using maximum a posterior (MAP):

$$\Theta = \underset{\Theta}{\operatorname{argmax}} \prod_{u\in\mathcal{U}}\prod_{l^c\in\mathcal{L}}\prod_{l_i\in\mathcal{L}}\prod_{l_j\in\mathcal{L}} P(>_{u,l^c}|\Theta)P(\Theta) \tag{7}$$

where $\Theta = \{X^S(\mathcal{L}), X^P(\mathcal{L}), X^P(\mathcal{U})\}$ is the set of parameters.

Using logistic function $\sigma(z) = \frac{1}{1+e^{-z}}$, ranking probability can be further written as

$$\begin{aligned}P(>_{u,l^c}|\Theta) &= P\left((\mathcal{D}_{u,l^c,l_j} - \mathcal{D}_{u,l^c,l_i}) > 0|\Theta\right) \\ &= \sigma(\mathcal{D}_{u,l^c,l_j} - \mathcal{D}_{u,l^c,l_i})\end{aligned} \tag{8}$$

With Gaussian priors on the parameters, we have the final objective function in Eq. 9, where λ is a parameter controlling the regularization term.

$$\begin{aligned}\Theta &= \underset{\Theta}{\operatorname{argmax}} \log \prod_{u\in\mathcal{U}}\prod_{l^c\in\mathcal{L}}\prod_{l_i\in\mathcal{L}}\prod_{l_j\in\mathcal{L}} \sigma(\mathcal{D}_{u,l^c,l_j} - \mathcal{D}_{u,l^c,l_i})P(\Theta) \\ &= \underset{\Theta}{\operatorname{argmax}} \sum_{u\in\mathcal{U}}\sum_{l^c\in\mathcal{L}}\sum_{l_i\in\mathcal{L}}\sum_{l_j\in\mathcal{L}} \log\left(\sigma(\mathcal{D}_{u,l^c,l_j} - \mathcal{D}_{u,l^c,l_i})\right) \\ &\quad - \lambda||\Theta||^2\end{aligned} \tag{9}$$

5.2 Learning Algorithm

Directly solving Eq. 9 is time consuming. Following the approach of BPR, we independently draw the training tuple and utilize Stochastic Gradient Descent to update the parameters. Based on the historical check-in data, we can obtain a set of observations $\{< u, l^c, l_i >\}$, where l^c is the current location of user u and l_i is the next check-in. For each observation, we randomly generate a POI l_j, which is not observed given u and l^c. Given a training tuple $< u, l^c, l_i, l^j >$, the update procedure is described as below:

$$\begin{aligned}\Theta &\leftarrow \Theta + \gamma \frac{\partial}{\partial \Theta} \left(\log \sigma(z) - \lambda ||\Theta||^2 \right) \\ &\leftarrow \Theta + \gamma \left((1 - \sigma(z)) \frac{\partial z}{\partial \Theta} - 2\lambda \Theta \right)\end{aligned} \tag{10}$$

where $z = \mathcal{D}_{u,l^c,l_j} - \mathcal{D}_{u,l^c,l_i}$, and γ is the learning rate.

The learning algorithm of PRME is summarized in **Algorithm** 1. First, we initialize all the parameters with a Normal distribution (Line 1). For each observation $< u, l^c, l_i >$, we randomly generate a POI l_j that user u has not visited after l^c. We then calculate the time difference $\Delta(l^c, l_i)$ to determine whether it is related to l^c. If $\Delta(l^c, l_i) < \tau$, latent positions in both user preference latent space (Line 6) and sequential transition space (Line 7) are updated. If $\Delta(l^c, l_i) \geq \tau$, only user preference is considered (Line 9). Note that all the update procedures (Lines 6,7,9) are based on Eq. 10. For PRME-G model, we utilize $\mathcal{D}^G_{u,l^c,l}$ in Eq. 9 as the optimization criterion and use the similar procedure to learn parameters.

The time complexity of the algorithm is $O(IK|\mathcal{C}|)$, where I is the number of iterations, K is the number of dimensions and $|\mathcal{C}|$ is the number of observed check-ins in the dataset.

Algorithm 1: PRME

```
input : check-in data C, learning rate γ, regularization λ,
        component weight α, time threshold τ
output: model parameters Θ = {X^S(L), X^P(L), X^P(U)}
1  Initialize Θ with Normal distribution N(0, 0.01);
2  repeat
3      for Each observation < u, l^c, l_i > do
4          Randomly generate an unobserved POI l_j;
5          if Δ(l^c, l_i) < τ then
6              Update X^P(u), X^P(l_i), X^P(l_j);
7              Update X^S(l^c), X^S(l_i), X^S(l_j);
8          if Δ(l^c, l_i) ≥ τ then
9              Update X^P(u), X^P(l_i), X^P(l_j);
   until convergence;
10 return Θ = {X^S(L), X^P(L), X^P(U)}
```

6 Experiments

6.1 Experimental Setup

Experimental setting

In the experiments, we use the two datasets introduced in Section 3. For the one-year check-ins data, we use the check-ins in the first 10 months as training set, the 11^{th} month as tuning set, and the last month as test set. We exploit two well-known measure metrics [Yuan *et al.*, 2013], namely Precision@N and Recall@N (denoted by Pre@N and Rec@N respectively). Given a user and his current location, we use the next check-in in successive τ hours as the ground truth. The time window threshold τ is set at 6 hours following [Cheng *et al.*, 2013]. Based on the tuning set, the number of dimensions is set at $K = 60$, learning rate $\gamma = 0.005$, regularization term $\lambda = 0.03$ and component weight $\alpha = 0.2$.

Evaluated methods

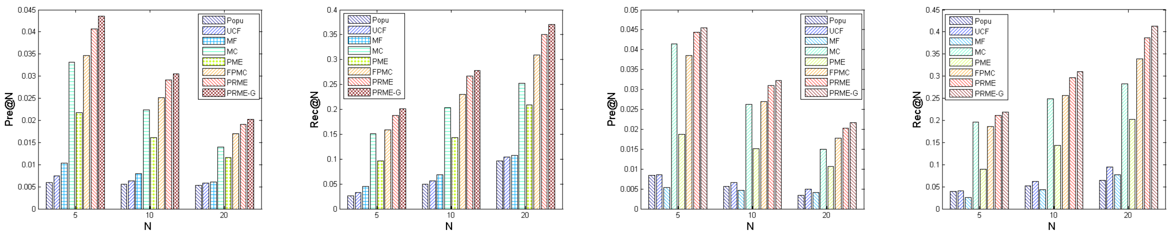
In the experiments, we compare PRME and PRME-G with the following baselines:

- Popu: the most popular POIs are recommended (user count ranking scheme in [Ye *et al.*, 2013]).
- UCF: User based CF [Yuan *et al.*, 2013; Ye *et al.*, 2011]
- MF: Matrix Factorization with BPR [Rendle *et al.*, 2009], which factorizes the user-item preference matrix.
- MC: first order Markov chain model [Zhang *et al.*, 2014], which computes the transition probability by counting method.
- PME: the personalized metric embedding [Wu *et al.*, 2013; Moore *et al.*, 2013], which projects users and POIs in a common latent space.
- FPMC: the state-of-the-art personalized next POI recommendation algorithm [Cheng *et al.*, 2013; Rendle *et al.*, 2010].

6.2 Performance of Methods

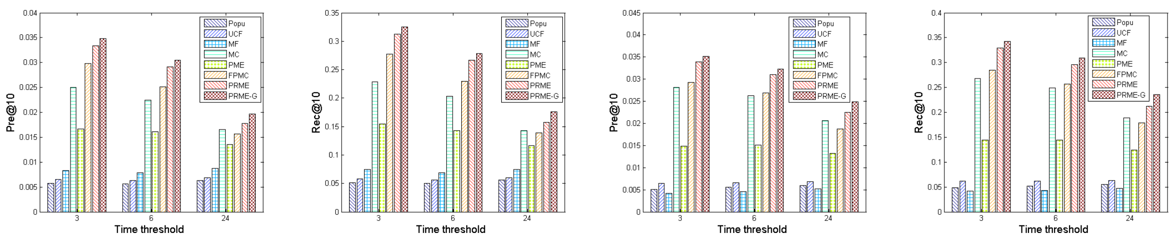
We compare the precision and recall of various methods in Figure 3. The lower precision and recall of Popu method indicates that this naive approach is not feasible for the next POI recommendation. Both UCF and MF perform poorly because they do not make use of sequential information. This result shows the conventional POI recommendation algorithms, which mainly exploit the user preference, are not effective for the N^2-POI recommendation. The relatively high performance of MC method demonstrates that the sequential information plays an important role in the N^2-POI recommendation. The PME performance is not acceptable because the learning of sequential transition and user preference would be interfered with each other in a common latent space. Note that FPMC is state-of-the-art personalized Markov chain algorithm. PRME consistently outperforms FPMC, which illustrates that representing each POI as one point in latent space is more effective than two independent vectors. PRME-G achieves the best performance, which shows that the geographical influence is beneficial for the N^2-POI recommendation. All reported improvements over baseline methods are statistically significant with p-value < 0.01.

We further compare PRME and FPMC with the localized region constraint [Cheng *et al.*, 2013]. Figure 4 shows the precision with different region constraints. The local region constraint means we only consider the candidate POIs with no more than d km (d=10, 20 and 40) from current location. Note that PRME and FPMC use the same setting. Under different local region constraints, PRME outperforms FPMC, which further verifies the advantage of PRME.



(a) Precision on FourSquare (b) Recall on FourSquare (c) Precision on Gowalla (d) Recall on Gowalla

Figure 3: The result of methods on Foursquare and Gowalla.



(a) Pre@10 on FourSquare (b) Rec@10 on FourSquare (c) Pre@10 on Gowalla (d) Rec@10 on Gowalla

Figure 5: The effect of time threshold τ on Foursquare and Gowalla.

6.3 Effect of Parameters

Effect of time threshold τ

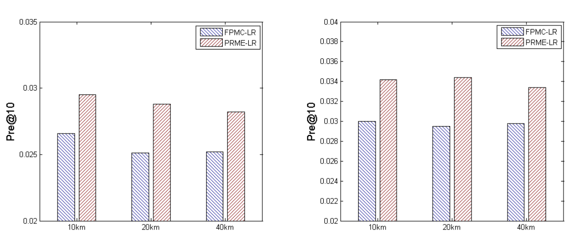
Figure 5 depicts the impact of the time threshold τ ($\tau = 3$, 6 and 24 hours). PRME and PRME-G outperform the baselines with various τ. Besides, when τ is large, the performance of sequence based methods (MC, PME, FPMC,PRME and PRME-G) decreases. This is because the location sequential transition becomes weak when time difference is large.

Effect of component weight α

Figure 6 shows the effect of weight α. The performance at $\alpha = 0$ (only sequential transition) is much better than $\alpha = 1$ (only user preference). This result implies that sequential influence is more important than user preference in the N^2-POI recommendation problem. The best performance is obtained when $\alpha = 0.2$. Hence, we set $\alpha = 0.2$ in our experiments.

Effect of number of dimensions K

We further investigate the impact of K with FPMC and PRME. Figure 7 shows the precision and recall for various



(a) Foursquare (b) Gowalla

Figure 4: Different localized region constraint on two datasets.

K. When $K \geq 20$, PRME outperforms FPMC in recommendation quality, which implies the superiority of PRME. The performance of FPMC and PRME increases with K because high dimensions can better embody the latent metric relationships. Empirically, we set $K = 60$ in our experiments, which achieves a satisfying trade off between recommendation quality and running time.

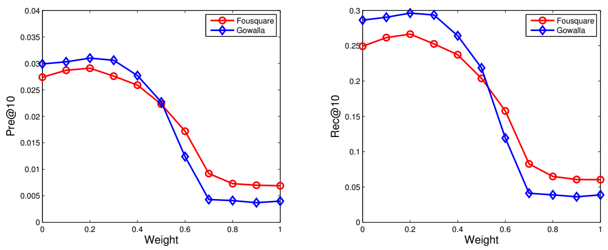
7 Conclusion and Future Work

In this paper, we study the *next new* POI recommendation problem. We propose a novel pair-wise Metric Embedding to model the sequential POI transition. We further develop PRME-G that jointly models three factors: sequential transition, individual preference, and geographical influence. Performance of our algorithms is demonstrated by extensive experiments on two datasets.

Several interesting future directions exist for further exploration. First, metric embedding can be used to provide visualization of the POI-POI and user-POI relationships. Second, the PRME model is not specific for POI recommendation task and can be utilized for other applications, such as product recommendation and friend recommendation.

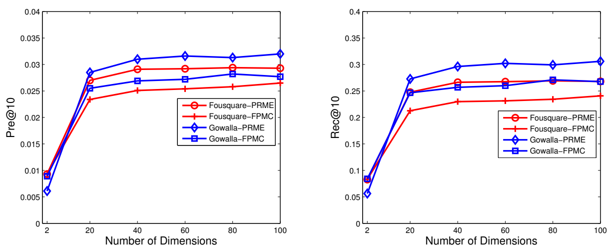
Acknowledgements

This work was supported in part by a grant awarded by Microsoft. ShanShan Feng would like to acknowledge the Ph.D. grant from Advanced Environmental Biotechnology Centre, Nanyang Environment and Water Research Institute, Interdisciplinary Graduate School, Nanyang Technological University, Singapore.



(a) Pre@10 (b) Rec@10

Figure 6: Effect of component weight α.



(a) Pre@10 (b) Rec@10

Figure 7: Effect of the number of dimension for PRME.

References

[Chen *et al.*, 2012] Shuo Chen, Josh L Moore, Douglas Turnbull, and Thorsten Joachims. Playlist prediction via metric embedding. In *KDD*, pages 714–722, 2012.

[Chen *et al.*, 2013] Shuo Chen, Jiexun Xu, and Thorsten Joachims. Multi-space probabilistic sequence modeling. In *KDD*, pages 865–873, 2013.

[Chen *et al.*, 2015] Xuefeng Chen, Yifeng Zeng, Gao Cong, Shengchao Qin, Yanping Xiang, and Yuanshun Dai. On information coverage for location category based point-of-interest recommendation. In *AAAI*, pages 37–43, 2015.

[Cheng *et al.*, 2012] Chen Cheng, Haiqin Yang, Irwin King, and Michael R Lyu. Fused matrix factorization with geographical and social influence in location-based social networks. In *AAAI*, 2012.

[Cheng *et al.*, 2013] Chen Cheng, Haiqin Yang, Michael R Lyu, and Irwin King. Where you like to go next: Successive point-of-interest recommendation. In *IJCAI*, pages 2605–2611, 2013.

[Cho *et al.*, 2011] Eunjoon Cho, Seth A Myers, and Jure Leskovec. Friendship and mobility: user movement in location-based social networks. In *KDD*, pages 1082–1090, 2011.

[Hinton and Roweis, 2002] Geoffrey E Hinton and Sam T Roweis. Stochastic neighbor embedding. In *NIPS*, pages 833–840, 2002.

[Li *et al.*, 2015] Xutao Li, Gao Cong, Xiaoli Li, Tuan Anh Nguyen Pham, and Shonali Krishnaswamy. Rank-geofm: A ranking based geographical factorization method for point of interest recommendation. In *SIGIR*, 2015.

[Lian *et al.*, 2013] Defu Lian, Vincent W Zheng, and Xing Xie. Collaborative filtering meets next check-in location prediction. In *WWW*, pages 231–232, 2013.

[Lian *et al.*, 2014] 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 *KDD*, pages 831–840, 2014.

[Liu *et al.*, 2013] Xin Liu, Yong Liu, Karl Aberer, and Chunyan Miao. Personalized point-of-interest recommendation by mining users' preference transition. In *CIKM*, pages 733–738, 2013.

[Moore *et al.*, 2013] Joshua L Moore, Shuo Chen, Douglas Turnbull, and Thorsten Joachims. Taste over time: The temporal dynamics of user preferences. In *ISMIR*, pages 401–406, 2013.

[Rendle *et al.*, 2009] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. Bpr: Bayesian personalized ranking from implicit feedback. In *UAI*, pages 452–461, 2009.

[Rendle *et al.*, 2010] Steffen Rendle, Christoph Freudenthaler, and Lars Schmidt-Thieme. Factorizing personalized markov chains for next-basket recommendation. In *WWW*, pages 811–820, 2010.

[Roweis and Saul, 2000] Sam T Roweis and Lawrence K Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500):2323–2326, 2000.

[Wu *et al.*, 2013] Xiang Wu, Qi Liu, Enhong Chen, Liang He, Jingsong Lv, Can Cao, and Guoping Hu. Personalized next-song recommendation in online karaokes. In *RecSys*, pages 137–140, 2013.

[Ye *et al.*, 2011] Mao Ye, Peifeng Yin, Wang-Chien Lee, and Dik-Lun Lee. Exploiting geographical influence for collaborative point-of-interest recommendation. In *SIGIR*, pages 325–334, 2011.

[Ye *et al.*, 2013] Jihang Ye, Zhe Zhu, and Hong Cheng. Whats your next move: User activity prediction in location-based social networks. In *SDM*, 2013.

[Yuan *et al.*, 2013] Quan Yuan, Gao Cong, Zongyang Ma, Aixin Sun, and Nadia Magnenat Thalmann. Time-aware point-of-interest recommendation. In *SIGIR*, pages 363–372, 2013.

[Zhang *et al.*, 2014] Jia-Dong Zhang, Chi-Yin Chow, and Yanhua Li. Lore: Exploiting sequential influence for location recommendations. In *SIGSPATIAL*, 2014.