# Inferring Continuous Latent Preference on Transition Intervals for Next Point-of-Interest Recommendation

Jing He[1], Xin Li[1] *, Lejian Liao[1], and Mingzhong Wang[2]

[1] BJ ER Center of HVLIP&CC, School of Computer Science, Beijing Institute of Technology, China
{skyhejing, xinli, liaolj} @ bit.edu.cn
[2] School of Business, University of Sunshine Coast, Australia
mwang@usc.edu.au

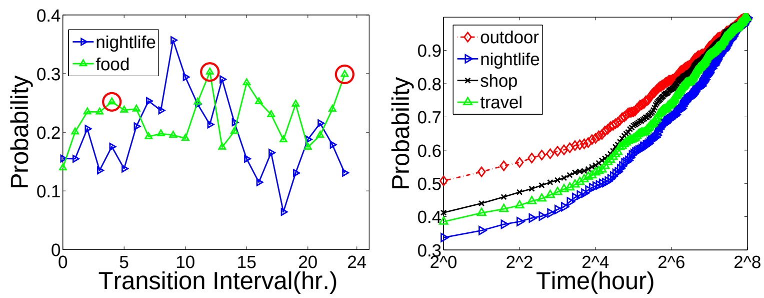
**Abstract.** Temporal information plays an important role in Point-of-Interest (POI) recommendations. Most existing studies model the temporal influence by utilizing the observed check-in time stamps explicitly. With the conjecture that transition intervals between successive check-ins may carry more information for diversified behavior patterns, we propose a probabilistic factor analysis model to incorporate three components, namely, personal preference, distance preference, and transition interval preference. They are modeled by an observed third-rank transition tensor, a distance constraint, and a continuous latent variable, respectively. In our framework, the POI recommendation and the transition interval for user's very next move can be inferred simultaneously by maximizing the posterior probability of the overall transitions. Expectation Maximization (EM) algorithm is used to tune the model parameters. We demonstrate that the proposed methodology achieves substantial gains in terms of prediction on next move and its expected time over state-of-the-art baselines.

**Keywords:** Point-of-Interest · Recommendation · Probabilistic factor analysis model.

## 1 Introduction

Recently, many efforts have been devoted to next Point-of-Interest (POI) recommendation [8, 18], which not only help users promptly identify their very next favorite POIs, but also benefit location-based social network (LBSNs) service providers to acquire more customers. However, to achieve accurate personalized POI recommendation is very challenging as it's well known that the check-in data of each user is in fact highly sparse. Existing work studied users' preferences on POIs by employing various context information, e.g., social relations, distance constraints and temporal information etc., to enhance the recommendation performance. To leverage on the temporal influence, some of existing works simply

* Corresponding Author: Xin Li.




(a) Trans. interv. after work (b) CDF of Transition Intervals

**Fig. 1.** Observations of transition intervals

explore the temporal periodicity of human mobility, as the intuition behind is that human mobility often exhibits periodical patterns. For example, certain types of locations, such as office and Gym, are visited regularly at the same time slot [5, 17, 9].

Another research line of considering the temporal information is to explore the sequential behavior from successive check-ins. Cheng et al. utilized the factorized personalized Markov Chain (FPMC) [4] to capture the time-critical successive check-ins for next POI recommendation. Specifically, [12] investigated how to perform POI recommendations for a specific time period by learning users' evolving sequential preferences. Therefore, the expected time that a user will check in some places can be estimated by enumerating all possible time.

Furthermore, almost all existing work focused on utilizing the time stamps (absolute time) of check-ins, e.g., Monday morning, or 9:00 pm on Saturday. As reported in [9], user behavior shows clear periodic patterns. For example, people usually check in a workplace at 9:00 am. However, we argue that utilizing such patterns implicitly assumes that all check-ins follow a probability distribution, e.g., Gaussian with fixed mean and variance. In fact, people with different occupations follow various office hours. Consequently, the temporal patterns towards other activities should also vary. Thus, most existing work cannot take such diversity into consideration. Therefore, we argue that the time-intervals between check-ins do reflect the temporal behavior patterns of human activity with more flexibility.

Fig.1(a) plots the probability of next check-in locations in City of New York (NYC)[3] for "food" and "nightlife" given the previous check-in as "work" over different time intervals. There are three local maxima when the intervals are 4 hours, 12 hours, and 23 hours for food check-in. The observation confirms a behavior pattern that people usually have breakfast 1 hour before, and have lunch and dinner at about 4 hours and 12 hours after they start to work, respectively. And the maximum of nightlife check-in indicates that people usually visit nightlife spots at about 10 hours after the start of work. In summary, the

[3] The check-in data is collected from Foursquare (See Sec.4.1)

working hour for each user may be different, but the transition intervals between check-in activities follow similar patterns. Fig.1(b) plots the cumulative distribution function (CDF) of the top-4 most popular location categories for "after work", in which the transition interval pattern is evident, e.g., the outdoor spots are often checked in after work within a short interval. We argue that such latent behavior patterns with time intervals should play a key role for next POI recommendation. The challenges come from two aspects, how to model such personalized latent behavior pattern and how to boost next POI recommendation with incorporation of the latent preferences.

In this paper, we propose a probabilistic approach to infer the continuous latent preference on transition intervals for next POI recommendation. Under our proposed framework, the next check-in and the time-intervals between successive check-ins can be obtained simultaneously. Thus, the checked-in time for a future activity can also be inferred from the time stamp of the previous check-in. Specifically, we model the overall transition preference with three components, namely, personal preference, spatial preference, and transition interval preference. They are modeled by a third-rank tensor, a distance constraint, and a continuous latent variable, respectively. Then, we develop a probabilistic factor analysis model to combine these three components. The probabilistic model performs statistical inference and Bayesian methods where the latent preference variables are assumed to follow Gaussian distributions with different means and variances. The transition probabilities, the estimated transition intervals, and the uncertainty of them are then obtained. During model learning phases, the Expectation Maximization (EM) is developed to optimize the proposed probability model. Experimental results demonstrate that the proposed model outperforms other state-of-the-art methods in terms of next POI recommendation and the expected transition time.

## 2 Related Work

Latest studies have integrated temporal influence into POI recommendation for further performance improvement.

[7] investigated the temporal cyclic patterns of user check-ins in terms of temporal non-uniformness and temporal consecutiveness and demonstrated its effectiveness to improve recommendation performance. [4] proposed a tensor-based FPMC-LR model for next POI recommendation by considering the sequential behaviors between check-ins. [6] proposed a personalized ranking metric embedding model (PRME) for next new POI recommendation by considering the order relationship between check-ins. [12] proposed a bi-weighted low-rank graph construction model to make recommendations for a specified future time period by considering users' evolving sequential preferences.

[11] explicitly modeled the check-in time as the modes of a fourth-rank tensor for next POI recommendation, in which the time interval is utilized to capture the intensity of relation between the two successive check-in locations. The approach is designed in a two-fold manner, which predicts the POI category first,

and then determine the expected visitings. Their work assumed that the intensity of relation decays over the transition time, but the temporal periodicity exists. Its applicability is constrained by the availability and accuracy of the categorization.

However, none of the existing methods investigated the transition interval pattern explicitly. Alternatively, they utilized absolute time to capture the temporal preference. We argue that modelling the transition interval preference may help us to capture more diversified human mobility patterns. Moreover, existing work [9] modeled the latent behavior pattern in a discrete manner, and the number of patterns must be predefined. This motivates us to further investigate on the transition latent variables. In this paper, we extend the transition preference with the dimension of transition interval preference, and apply a continuous latent variable factor analysis approach to remove the requirement of predefinition.

## 3 Model Framework

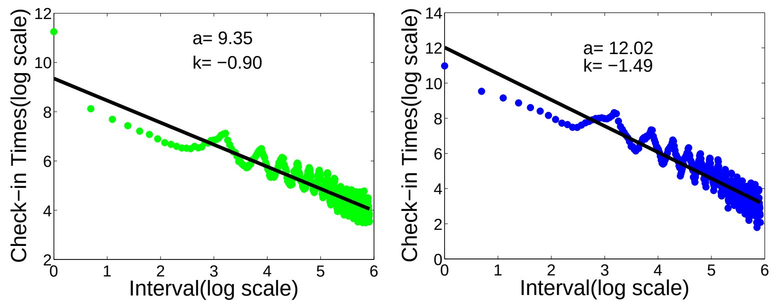
Let $U = \{u_1, u_2, ..., u_M\}$ be a set of LBSN users, and $L = \{l_1, l_2, ..., l_N\}$ be a set of locations (POIs). Let $l_u^t$ be the location visited by user $u$ at time $t$, then the set of locations visited by $u$ before time $t$ is denoted by $L_u^t$, and $L_u^t = \{l_u^1, ..., l_u^{t-1}\}$. Let $\tau_{u,i,j}$ be the transition interval for $u$ between location $i$ and $j$, and the corresponding set of transition intervals for $L_u^t$ is denoted by $T_u = \{\tau_{u,l_u^1,l_u^2}, ..., \tau_{u,l_u^{t-2},l_u^{t-1}}\}$. Our goal is to recommend user $u$ next POI via the ranking of probabilities that he/she will move from the current location $i$ to the next location $j$, with the transition interval $\tau_{u,i,j}$ inferred at the same time. Based on first-order Markov chain property, the transition probability is denoted as $x_{u,i,j} = p(j = l_u^t | i = l_u^{t-1})$. Thus, each user is associated with a specific transition matrix which in total generates a transition tensor $\chi \in \mathbb{N}^{|U| \times |L| \times |L|}$ with each $\chi_{u,i,j}$ representing the observed transition frequency of user $u$ from location $i$ to location $j$.

### 3.1 Preferences Modeling

***Personal Preference***. As the transitions among $\chi$ are partially observed, we follow most previous work [4, 9] to apply the low-rank factorization model, a special case of Canonical Decomposition which models the pairwise interaction between all three modes of the tensor (user $U$, current location $I$, next location $J$), to fill up the missing values, given as:

$$\hat{\chi}_{u,i,j} = \boldsymbol{v}_u^{U,J} \cdot \boldsymbol{v}_j^{J,U} + \boldsymbol{v}_j^{J,I} \cdot \boldsymbol{v}_i^{I,J} + \boldsymbol{v}_u^{U,I} \cdot \boldsymbol{v}_i^{I,U} \quad (1)$$

where $\boldsymbol{v}_u^{U,J}$ and $\boldsymbol{v}_j^{J,U}$ denote the latent factor vectors for users and next locations, respectively. Other notions are defined in the same manner. The term $\boldsymbol{v}_u^{U,I} \cdot \boldsymbol{v}_i^{I,U}$




(a) NYC (b) Gowalla

**Fig. 2.** Transition intervals v.s. the number of check-ins

can be removed since it is independent of location $j$ and does not affect the ranking result [14], thus, leading to a more compact expression for $\hat{\chi}_{u,i,j}$:

$$\hat{\chi}_{u,i,j} = \boldsymbol{v}_u^{U,J} \cdot \boldsymbol{v}_j^{J,U} + \boldsymbol{v}_j^{J,I} \cdot \boldsymbol{v}_i^{I,J} \tag{2}$$

***Transition Interval Preference***. To extract the transition interval preference of user $u$ for the next location, we first define the transition interval tensor $Z$. With the temporal intervals between successive POIs extracted for user $u$, the POI-POI transition interval matrix $Z_u$ is constructed, which in turn generates a transition interval tensor $Z \in \mathbb{R}^{|U|\times|L|\times|L|}$ with each $\hat{z}_{u,i,j}$ representing the estimated transition interval of user $u$ from location $i$ to location $j$. Similar to yielding $\hat{\chi}_{u,i,j}$, $\hat{z}_{u,i,j}$ is generated by modeling the pairwise interaction between the modes of the transition interval tensor, given as:

$$\hat{z}_{u,i,j} = \boldsymbol{e}_u^{U,J} \cdot \boldsymbol{e}_j^{J,U} + \boldsymbol{e}_j^{J,I} \cdot \boldsymbol{e}_i^{I,J} \tag{3}$$

where $\boldsymbol{e}_u^{U,J}$ and $\boldsymbol{e}_j^{J,U}$ denote the latent factor vectors for users and next locations in tensor $Z$, respectively. Other notions are defined in the same manner.

To investigate the relation between the number of transitions and the transition intervals, we plot the check-in counts with respect to its transition intervals between two successive check-ins for Foursquare-NYC dataset and Gowalla dataset in log scale (Fig.2). We can observe that the number of transitions decreases as the temporal interval increases, and the relation follows a power law distribution with exponent $k \approx -1$. Therefore, we define $z_{u,i,j}$ to specifically indicate the latent transition interval preference for user $u$ to visit location $j$ from the current location $i$ by leveraging on the observations in Fig.2 under conditions of uncertainty. We assume the latent variable $z_{u,i,j}$ is Gaussian with prior mean of $\hat{z}_{u,i,j}^{-1}$ and variance of $\sigma_1^2$, given as:

$$z_{u,i,j} \sim N(\hat{z}_{u,i,j}^{-1}, \sigma_1^2) \tag{4}$$

***Spatial Preference***. Users' mobility is geographically constrained by the distance that one can travel within a limited time, and their preference to visit a location decreases as the geographical distance increases [5]. Most POIs which are likely to be explored are close to users' residence, workplace, and frequently visited POIs. Hence, the spatial behaviors of users can be utilized to enhance next POI recommendation. Here, we define the spatial influence $sp(d_{i,j})$ for any user to visit a location $j$ which is $d_{i,j}$ km away from the current location $i$ to leverage on the distance constraint: $sp(d_{i,j}) = \rho \cdot d_{i,j}^{-1}$. The optimal setting of $\rho$ will be learned during the inference phase.

### 3.2 A Factor Analysis Latent Variable Model

In this paper, we adopt a so-called statistical *factor analysis* [2] latent variable model to incorporate the aforementioned three preferences, in which the combination is a linear function for observed data $x$:

$$x = w \cdot z + \mu + \epsilon \tag{5}$$

where z denotes the latent variable, $\epsilon$ is a $z$-independent noise process, $w$ contains the *factor loading*, and $\mu$ permits the model to have non-zero mean. Conventionally, the latent variables are assumed to be independent and to follow standard normal distribution, $z \sim N(0, 1)$, and the noise follows Gaussian distribution, $\epsilon \sim N(0, \sigma^2)$. Then the variable x induced from Eq.(5) follows Gaussian distribution, $x \sim N(\mu, w^2 + \sigma^2)$. The parameters may thus be inferred in a maximum-likelihood manner, and there is no closed-form analytic solution for $w$ and $\sigma^2$.

We define $\mu = \hat{\chi}_{u,i,j} + \rho \cdot d_{i,j}^{-1}$ to incorporate the personal preference and distance preference, then a special case of factor analysis for next POI recommendation can be obtained as:

$$\hat{x}_{u,i,j} = w \cdot z_{u,i,j} + \hat{\chi}_{u,i,j} + \rho \cdot d_{i,j}^{-1} + \epsilon \tag{6}$$

where $w$ is a trade-off parameter used to control the contribution of the transition interval preference and will be learned during model inference phase. The set of all parameters for the proposed model is $\boldsymbol{\Theta} := \{\boldsymbol{\rho}, \boldsymbol{w}, \boldsymbol{\sigma}_1^2, \boldsymbol{\sigma}_2^2, \boldsymbol{V}_u^{U,J}, \boldsymbol{V}_j^{J,U}, \boldsymbol{V}_j^{J,I}, \boldsymbol{V}_i^{I,J}, \boldsymbol{E}_u^{U,J}, \boldsymbol{E}_j^{J,U}, \boldsymbol{E}_j^{J,I}, \boldsymbol{E}_i^{I,J}\}$. In general, we believe that user behavior is personalized, and it is reasonable to assume that the check-ins from the same user share the same value of parameters $\{\boldsymbol{w}, \boldsymbol{\sigma}_1^2, \boldsymbol{\sigma}_2^2\}$. Otherwise, they are not learnable for check-ins from the test set.

In our model, the noise represents random influences in the transitions that is not generated from the user preference, but arises from social network, weather, etc. Since the distribution over the noise is also Gaussian and defined as $\epsilon \sim N(0, \sigma_2^2)$, Eq. (6) implies a probability distribution over $x_{u,i,j}$ for a given $z_{u,i,j}$:

$$x_{u,i,j} | z_{u,i,j} \sim N(w \cdot z_{u,i,j} + \hat{\chi}_{u,i,j} + \rho \cdot d_{i,j}^{-1}, \sigma_2^2) \tag{7}$$

Given the above formulations, the marginal distribution for the observed $x_{u,i,j}$ is then obtained by integrating latent variables and is likewise Gaussian:

$$x_{u,i,j} \sim N(w \cdot \hat{z}^{-1}_{u,i,j} + \hat{\chi}_{u,i,j} + \rho \cdot d^{-1}_{i,j}, w^2 \cdot \sigma_1^2 + \sigma_2^2) \tag{8}$$

The conditional distribution of the latent variable $z_{u,i,j}$ given the observed $x_{u,i,j}$ can be derived by Bayes rule and is also Gaussian:

$$z_{u,i,j} | x_{u,i,j} \sim N(C, M) \tag{9}$$

where the posterior mean and posterior variance are specified by $C = \frac{\sigma_1^2 w(x_{u,i,j} - \hat{\chi}_{u,i,j} - \rho d^{-1}_{i,j}) + \hat{z}^{-1}_{u,i,j}\sigma_2^2}{w^2\sigma_1^2 + \sigma_2^2}$ and $M = \frac{\sigma_1^2 \cdot \sigma_2^2}{w^2 \cdot \sigma_1^2 + \sigma_2^2}$ respectively.

### 3.3 Parameter Inference

Learning in probabilistic models can be simplified as maximizing the data log-likelihood with respect to all the model parameters. In our model, we consider the latent variables $\{z_{u,i,j}\}$ to be 'missing data' and complete data to comprise the observed transitions $x_{u,i,j}$ together with them. Assuming that users are independent as well as their check-in histories are independent, the corresponding complete-data log-likelihood is given as:

$$\begin{aligned} \mathcal{L}_C &= \sum_{u \in U} \sum_{i \in L_u} \sum_{j \in L_u^t} \ln\{p(x_{u,i,j}, z_{u,i,j})\} \\ &= \sum_{u \in U} \sum_{i \in L_u} \sum_{j \in L_u^t} \ln\{p(x_{u,i,j} | z_{u,i,j}) p(z_{u,i,j})\} \end{aligned} \tag{10}$$

where the components are derived from Eq. (7) and Eq. (4):

$$\begin{aligned} p(x_{u,i,j} | z_{u,i,j}) = (2\pi\sigma_2^2)^{-\frac{1}{2}} \cdot \\ \exp\{-\frac{(x_{u,i,j} - w \cdot z_{u,i,j} - \hat{\chi}_{u,i,j} - \rho \cdot d^{-1}_{i,j})^2}{2\sigma_2^2}\} \end{aligned} \tag{11}$$

$$p(z_{u,i,j}) = (2\pi\sigma_1^2)^{-\frac{1}{2}} \exp\{-\frac{(z_{u,i,j} - \hat{z}^{-1}_{u,i,j})^2}{2\sigma_1^2}\}. \tag{12}$$

Estimates for the model parameters $\boldsymbol{\Theta}$ may be obtained by iterative maximization of $\mathcal{L}_C$, and a typical approach is to use Expectation-Maximization (EM) algorithm [15]. It is well known that EM algorithm iterates the two steps *expectation* (E-step) and *maximization* (M-step) until convergence, and it is guaranteed to increase the data likelihood to a local maximum. In the E-step, the expectation of $\mathcal{L}_C$, with respect to the posterior distribution of $z_{u,i,j}$ given the observed

$x_{u,i,j}$, is computed by using the current estimate for the parameters $\boldsymbol{\Theta}$. In the M-step, new parameter values $\boldsymbol{\Theta}'$ are determined by maximizing the expected complete-data log-likelihood. In our inference process, the corresponding steps are defined as follows:

***E-step:*** we take the expectation of $\mathcal{L}_C$ with respect to the distributions $p(z_{u,i,j}|x_{u,i,j})$:

$$\begin{aligned}\langle\mathcal{L}_C\rangle = -\frac{1}{2}\sum_{u\in U}\sum_{i\in L_u}\sum_{j\in L_u^t}\{\ln(2\pi\sigma_2^2) + \\ \frac{((x_{u,i,j}-\hat{\chi}_{u,i,j}-\rho d_{i,j}^{-1})^2 + w^2\langle z_{u,i,j}^2\rangle}{\sigma_2^2} - \\ \frac{2w\langle z_{u,i,j}\rangle(x_{u,i,j}-\hat{\chi}_{u,i,j}-\rho d_{i,j}^{-1})}{\sigma_2^2} + \\ \ln(2\pi\sigma_1^2) + \frac{\langle z_{u,i,j}^2\rangle - 2\langle z_{u,i,j}\rangle\hat{z}_{u,i,j}^{-1} + \hat{z}_{u,i,j}^{-2}}{\sigma_1^2}\}\end{aligned} \tag{13}$$

where we have omitted terms independent of the model parameters. The involved expectations are given as:

$$\langle z_{u,i,j}\rangle = \frac{\sigma_1^2 w(x_{u,i,j}-\hat{\chi}_{u,i,j}-\rho d_{i,j}^{-1}) + \hat{z}_{u,i,j}^{-1}\sigma_2^2}{w^2\sigma_1^2+\sigma_2^2} \tag{14}$$

$$\langle z_{u,i,j}^2\rangle = \frac{\sigma_1^2\cdot\sigma_2^2}{\sigma_2^2 + w^2\cdot\sigma_1^2} + \langle z_{u,i,j}\rangle^2 \tag{15}$$

where $\langle z_{u,i,j}\rangle$ denotes the posterior mean of Eq.(9), and $\langle z_{u,i,j}^2\rangle$ is obtained in conjunction with the posterior variance of Eq.(9).

***M-step:*** $\langle\mathcal{L}_C\rangle$ is maximized with respect to the model parameters $\boldsymbol{\Theta}$. This can be done by differentiating Eq.(13) and setting the partial derivatives to be zero, which gives a closed-form solution for parameters of $\{\boldsymbol{\rho}, \boldsymbol{w}, \boldsymbol{\sigma}_1^2, \boldsymbol{\sigma}_2^2\}$. For other parameters in factorization model, their values must be obtained via an iterative procedure as there are no closed-form solutions for them. Specifically, to obtain the revised parameters of $\{\boldsymbol{V}_u^{U,J}, \boldsymbol{V}_j^{J,U}, \boldsymbol{V}_j^{J,I}, \boldsymbol{V}_i^{I,J}, \boldsymbol{E}_u^{U,J}, \boldsymbol{E}_j^{J,U}, \boldsymbol{E}_j^{J,I}, \boldsymbol{E}_i^{I,J}$ $\}$, we follow the widely used stochastic gradient decent (SGD) algorithm to optimize the partial derivations to be zero with respect to each parameter. That is to take the second derivative of $\mathcal{L}_C$. Then, the updating procedure is performed as:

$$\Theta' = \Theta + \alpha(\frac{\partial}{\partial\Theta}(\frac{\partial}{\partial\Theta}\langle\mathcal{L}_C\rangle)) \tag{16}$$

where $\alpha > 0$ is the learning rate.

**Algorithm 1** Our Proposed Methodology

**Input:** check-in data D
**repeat**
  **E-Step:**
  $\langle z_{u,i,j} \rangle \leftarrow \frac{\sigma_1^2 w(\chi_{u,i,j} - \hat{\chi}_{u,i,j} - \rho d_{i,j}^{-1}) + \hat{z}_{u,i,j}^{-1} \sigma_2^2}{w^2 \sigma_1^2 + \sigma_2^2}$
  $\langle z_{u,i,j}^2 \rangle \leftarrow \frac{\sigma_1^2 \cdot \sigma_2^2}{\sigma_2^2 + w^2 \cdot \sigma_1^2} + \langle z_{u,i,j} \rangle^2$
  **M-Step:**
  $\sigma_1^2 \leftarrow \frac{1}{N_{u,d}} \sum_{u,d} (\langle z_{u,i,j}^2 \rangle - 2 \langle z_{u,i,j} \rangle \cdot \hat{z}_{u,i,j}^{-1} + \hat{z}_{u,i,j}^{-2})$
  $\sigma_2^2 \leftarrow \frac{1}{N_{u,d}} \sum_{u,d} (w^2 \cdot \langle z_{u,i,j}^2 \rangle + (\hat{\chi}_{u,i,j} + \rho d_{i,j}^{-1})^2 + \chi_{u,i,j}^2 + 2w \cdot \langle z_{u,i,j} \rangle \cdot (\hat{\chi}_{u,i,j} + \rho d_{i,j}^{-1}) - 2w \cdot \chi_{u,i,j} \cdot \langle z_{u,i,j} \rangle) - 2\chi_{u,i,j} \cdot (\hat{\chi}_{u,i,j} + \rho d_{i,j}^{-1})$
  $\rho \leftarrow \frac{\sum_d d_{i,j}^{-1} \cdot (\chi_{u,i,j} - w \langle z_{u,i,j} \rangle) - \sum_d d_{i,j}^{-1} \cdot \hat{\chi}_{u,i,j}}{\sum_d d_{i,j}^{-2}}$
  $w \leftarrow \frac{\sum_{u,d} \langle z_{u,i,j} \rangle \cdot (\chi_{u,i,j} - \hat{\chi}_{u,i,j} - \rho d_{i,j}^{-1})}{\sum_{u,d} \langle z_{u,i,j}^2 \rangle}$
  $\gamma_1 \leftarrow \hat{\chi}_{u,i,j} + \rho \cdot d_{i,j}^{-1} + w \langle z_{u,i,j} \rangle - \chi_{u,i,j}$
  $\boldsymbol{v}_u^{U,J} \leftarrow \boldsymbol{v}_u^{U,J} + \alpha(2\boldsymbol{v}_j^{J,U} \cdot \gamma_1)$
  $\boldsymbol{v}_j^{J,U} \leftarrow \boldsymbol{v}_j^{J,U} + \alpha(2\boldsymbol{v}_u^{U,J} \cdot \gamma_1)$
  $\boldsymbol{v}_j^{J,I} \leftarrow \boldsymbol{v}_j^{J,I} + \alpha(2\boldsymbol{v}_i^{I,J} \cdot \gamma_1)$
  $\boldsymbol{v}_i^{I,J} \leftarrow \boldsymbol{v}_i^{I,J} + \alpha(2\boldsymbol{v}_j^{J,I} \cdot \gamma_1)$
  $\boldsymbol{e}_u^{U,J} \leftarrow \boldsymbol{e}_u^{U,J} + \alpha(2\boldsymbol{e}_j^{J,U} \cdot (\hat{z}_{u,i,j} - \langle z_{u,i,j} \rangle^{-1}))$
  $\boldsymbol{e}_j^{J,U} \leftarrow \boldsymbol{e}_j^{J,U} + \alpha(2\boldsymbol{e}_u^{U,J} \cdot (\hat{z}_{u,i,j} - \langle z_{u,i,j} \rangle^{-1}))$
  $\boldsymbol{e}_j^{J,I} \leftarrow \boldsymbol{e}_j^{J,I} + \alpha(2\boldsymbol{e}_i^{I,J} \cdot (\hat{z}_{u,i,j} - \langle z_{u,i,j} \rangle^{-1}))$
  $\boldsymbol{e}_i^{I,J} \leftarrow \boldsymbol{e}_i^{I,J} + \alpha(2\boldsymbol{e}_j^{J,I} \cdot (\hat{z}_{u,i,j} - \langle z_{u,i,j} \rangle^{-1}))$
**until** convergence
**Return:** $\boldsymbol{\Theta}$

To maximize the likelihood, the sufficient statistics of the posterior distribution are calculated from the E-step Eq.(14) and Eq.(15), after which revised estimates of parameters are obtained from M-step. These steps are iterated in sequence until the algorithm is judged to be converged. The detailed algorithm and the parameter updating rules are shown in **Algorithm** 1.

## 4 Experiments

### 4.1 Datasets

We evaluate models on two real-world datasets which are acquired from Foursquare (NYC) and Gowalla and provided by [1] and [3] respectively. The statistics of the three datasets are listed in Table 1. Each dataset is split into two non-overlapping subsets to evaluate the model performance (for each user, the earliest 80% of check-ins as training set, and the remaining 20% check-ins as test set).

**Table 1.** Dataset Statistics

| | #User | #POI | #Check-in |
|---|---|---|---|
| Fours.-NYC | 3401 | 106974 | 178143 |
| Gowalla | 1488 | 92679 | 226116 |

### 4.2 Evaluation Metrics

Different from the existing approaches, e.g., FPMC-LR, which took a set of POIs visited within a time interval as the previous/next POI to fit the training process to overcome the data sparsity, we take only two POIs to construct the <previous, next> POI pair during our training process, and to predict the "exact" next POI, instead of a set of next POIs as in FPMC-LR. Thus we only adopt recall to evaluate the performance for the "exact" next POI recommendation, as whatsoever the length of recommendation list is increasing, there exists only one correct solution for the "exact" next POI recommendation, and the precision cannot be higher than $1/|recommendation\ list|$. Therefore, we evaluate the performance of the next POI recommendation and the next new POI recommendation by defining recall as[4]:

$$Recall@N_{POI} = \frac{1}{|U|}\sum_{u\in U}\frac{|S_{N,u}^{POI}\cap S_{visited}^{POI}|}{|S_{visited}^{POI}|} \tag{17}$$

$$Recall@N_{POI}^{new} = \frac{1}{|U|}\sum_{u\in U}\frac{|S_{N,u}^{POI}\cap S_{visited}^{newPOI}|}{|S_{visited}^{newPOI}|} \tag{18}$$

where $S_{N,u}^{POI}$ is the list of top-N recommended POIs in descending order, $S_{visited}^{POI}$ denotes the visited POIs for user $u$, and $S_{visited}^{newPOI}$ denotes the locations that haven't been visited by a user yet in the training set but will be visited in the test set. $|U|$ denotes the number of the users, and N is the size of the next POI candidate list.

To make a fair comparison with the existing works, we further evaluate the performance of next POI recommendation by considering consecutive next check-ins within $\gamma$ hours as the next location set ($\gamma$ is set to 6 following [4] and [6]). Precision and recall are accordingly defined as:

$$Precision@N = \frac{|S_{N,u}^{POI}\cap S_{visited}^{\gamma}|}{N} \tag{19}$$

$$Recall@N = \frac{|S_{N,u}^{POI}\cap S_{visited}^{\gamma}|}{|S_{visited}^{\gamma}|} \tag{20}$$

[4] The precision is $Recall/|recomendation\ list|$ in our problem, where $|recomendation\ list|$ denotes the length of the recomm. list.

where $S_{visited}^{\gamma}$ denotes the visited POIs for user $u$ in the next $\gamma$ hours.The precision and recall are computed by averaging all precision and recall for all samples in test set.

It is a relatively new research topic to predict the transition interval and evaluate the performance for such a model. We use the following two metrics to evaluate the performance of transition interval prediction.

– Mean Absolute Percentage Error (MAPE) focuses on the difference between the estimated transition interval $\hat{z}_{u,i,j}$ and the actual time interval $T_{u,i,j}$ across all testing data:

$$MAPE = \frac{1}{|N_d|} \sum_{N_d} \frac{|T_{u,i,j} - \hat{z}_{u,i,j}|}{T_{u,i,j}} \tag{21}$$

where $|N_d|$ is the size of the test set.The model with smaller error is the better one.

– Precision for the POI recommendation is introduced to help the evaluation for the predicted transition interval of each movement. It is introduced because MAPE is susceptible to large errors and often take more weights from them, which makes MAPE less appropriate to evaluate the task of personalized POI recommendation. It is defined as:

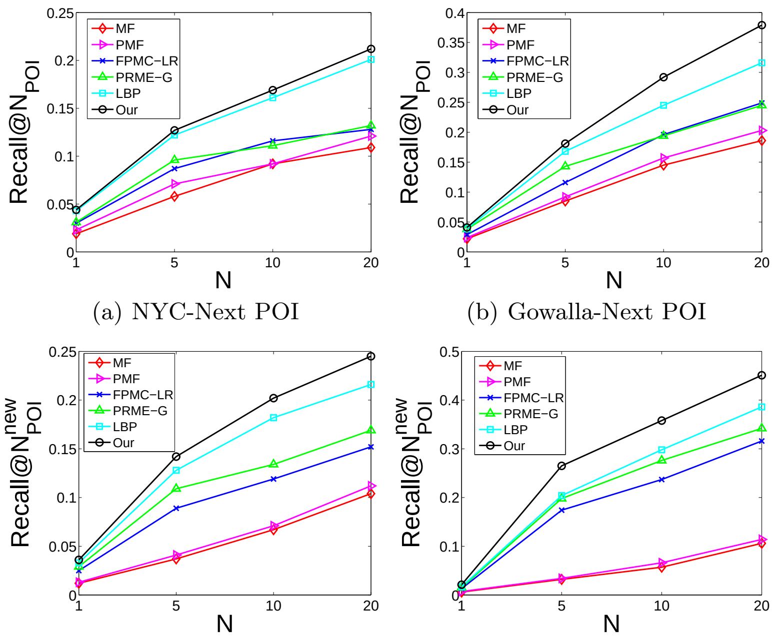
$$Precision@T = \frac{1}{|U|} \sum_{u \in U} \frac{sum(\boldsymbol{S}_{T,u})}{|S_{visited}^{POI}|} \tag{22}$$

where $S_{T,u}$ equals to “1” if the difference between $\hat{z}_{u,i,j}$ and $T_{u,i,j}$ is less than a specified threshold $T$, i.e $|\hat{z}_{u,i,j} - T_{u,i,j}| < T$, or “0” otherwise.

### 4.3 Performance Comparison on Next POI Recommendation

Alongside with our model, the following state-of-the-art methods are evaluated and compared on the performance of next POI recommendation:

– **MF:** it factorizes the user-POI preference matrix in conventional recommender system [10].
– **PMF:** Probabilistic Matrix Factorization (PMF) [13] is a generalized matrix factorization model for traditional recommendation task.
– **FPMC-LR:** it extends factorized personalized Markov chain with the localized region constraint, which uses BPR as the optimization criterion[4].
– **PRME-G:** it considers the distance between current location and next location for metric embedding, which is the state-of-the-art personalized sequential POI recommendation [6].
– **LBP:** it jointly models next POI recommendation under the influence of users’ latent behavior pattern, which is determined by the periodic property




(a) NYC-Next POI (b) Gowalla-Next POI

(c) NYC-Next New POI (d) Gowalla-Next New POI

**Fig. 3.** Performance Comparison on Next POI Recommendation

of human mobility along with time and the transition periodicity of location categories [9][5].

The parameters are tuned in the training set to find the optimal values, and they are subsequently used in the testing set. Fig.3 depicts the detailed results. The results show that:

- FPMC-LR, PRME-G, and our model all outperform MF and PMF significantly. It indicates that spatial influence plays an important role in next POI recommendation. Moreover, our model consistently outperforms FPMC-LR, PRME-G and LBP, which leads to the conclusion that the consideration of transition interval preference can better capture users' mobility in LBSNs.
- Our model has improved performance over baselines for next new POI recommendation. We argue that the gain also comes from the adoption of the transition interval preference. Since $\hat{\chi}_{u,i,j}$ models the observed transitions,

[5] LBP, FPMC-LR, PRME-G are not included in Table 4 for expected time prediction as we cannot find a way to predict transition interval valued in LBP. In fact, they all utilized BPR, a pairwise learning to rank algorithm, which takes the positive instance as pairs. However, there is no reasonable solution to define the positive instance or negative instance for the transition intervals. In this paper we use Canonical Decomposition, which is a special form of Tensor Factorization (TF), as an alternative approach for comparison.

**Table 2.** Performance Comparison of $\gamma$-hour Next POI Recommendation on NYC

| Metrics | Precision\Recall | | | | | |
|---|---|---|---|---|---|---|
| | MF | PMF | F-LR | P-G | LBP | Our |
| top1 | 0.028\0.009 | 0.035\0.010 | 0.042\0.016 | 0.045\0.017 | 0.065\0.043 | **.081\.053** |
| Imprv. | 189%\489% | 131%\430% | 92.9%\231% | 80.0%\212% | 24.6%\23.3% | |
| top5 | 0.025\0.045 | 0.026\0.047 | 0.037\0.065 | 0.037\0.067 | 0.051\0.128 | **.065\.157** |
| Imprv. | 160%\249% | 150%\234% | 75.7%\142% | 75.7%\134% | 27.5%\22.7% | |
| top10 | 0.022\0.070 | 0.023\0.080 | 0.032\0.102 | 0.031\0.100 | 0.040\0.170 | **.050\.211** |
| Imprv. | 127%\201% | 117%\164% | 56.3%\107% | 61.3%\111% | 25.0%\24.1% | |
| top20 | 0.019\0.110 | 0.020\0.120 | 0.024\0.152 | 0.023\0.148 | 0.026\0.210 | **.035\.264** |
| Imprv. | 84%\140% | 75.0%\120% | 45.8%\73.7% | 52.2%\78.4% | 34.6%\25.7% | |

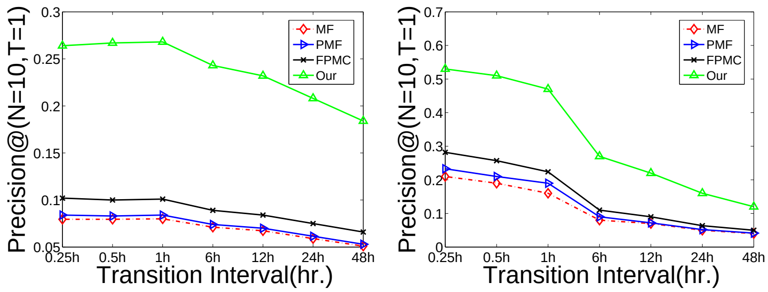
**Table 3.** Performance Comparison of $\gamma$-hour Next POI Recommendation on Gowalla

| Metrics | Precision\Recall | | | | | |
|---|---|---|---|---|---|---|
| | MF | PMF | F-LR | P-G | LBP | Our |
| top1 | 0.008\0.003 | 0.009\0.004 | 0.013\0.006 | 0.014\0.007 | 0.018\0.009 | **.023\.012** |
| Imprv. | 188%\300% | 156%\200% | 76.9%\100% | 64.3%\71.4% | 27.8%\33.3% | |
| top5 | 0.035\0.101 | 0.038\0.105 | 0.059\0.123 | 0.063\0.131 | 0.095\0.195 | **.121\.267** |
| Imprv. | 246%\164% | 218%\154% | 105%\117% | 92.1%\104% | 27.4%\36.9% | |
| top10 | 0.033\0.132 | 0.036\0.143 | 0.052\0.199 | 0.058\0.223 | 0.081\0.302 | **.109\.411** |
| Imprv. | 230%\211% | 203%\187% | 110%\107% | 87.9%\84.3% | 34.6%\36.1% | |
| top20 | 0.030\0.232 | 0.031\0.251 | 0.042\0.293 | 0.045\0.321 | 0.061\0.394 | **.082\.537** |
| Imprv. | 173%\131% | 165%\114% | 95.2%\83.3% | 82.2%\67.3% | 34.4%\36.3% | |

the history of visiting "next new POI" is blank ($\hat{\chi}_{u,i,j} \approx 0$) in the training data. Therefore, the performance enhancement to $\hat{x}_{u,i,j}$ for next new POI recommendation is due to the part of $w \cdot z_{u,i,j}$ in Eq. (8)[6].

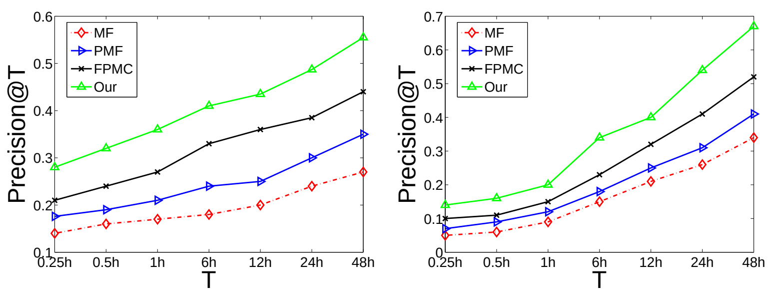
Table 2 - 3 tabulate the comparison results when considering the next POI as a set of locations. It's obvious to see that our proposed model consistently outperforms other baselines in terms of precision and recall. And the improvement is even better than that in Fig.3, which further verifies the effectiveness of incorporating transition interval preference for predicting POIs within a time slot, say for the next $\gamma$ hours. We can also observe that precisions of $\gamma$-hour next POI recommendation fluctuate with increasing the length of recommendation list, which mainly accounts to the nature of the dataset. Our collected check-in data is a relatively sparse dataset. For example, the users may only contribute few check-ins in the next $\gamma$ hour, the precision will inevitably decreases with the size of the candidate list increasing according to Eq.(19). Thus, we argue that recall is a more proper measurement to evaluate the performance for next POI recommendation.

[6] F-LR, P-G and LBP all apply the distance constraint similar to our distance preference model one way or another.




(a) NYC (b) Gowalla

**Fig. 4.** Perf. Comparison for Transition Interval Prediction




(a) NYC (b) Gowalla

**Fig. 5.** Transition interval prediction v.s. T

### 4.4 Performance Comparison on Transition Interval

The performance comparison on transition intervals is performed between our model and the following baselines: MF, PMF, and FPMC. However, to our best knowledge, our model is the first work which is capable of providing the transition interval for next POI, and capable of obtaining the transition interval and next POI recommendation simultaneously. There is no way to get the transition interval values along with the POI recommendation directly in MF, PMF, and FPMC. In order to perform the comparison, we use the transition interval matrix/tensor (from MF, PMF, and FPMC) between POIs to generate the unobserved transition intervals. That is, we factorize the observed transition matrix and interval matrix from them separately, and then align the results to get their transition intervals for next POIs.

Fig.4 shows the performance comparison on transition interval predictions. The results show that the proposed model always achieves the highest precision over baselines, which proves that our model is capable of providing effective POI recommendations to users as well as predicting how soon it will happen. We also compute MAPE between the predicted transition intervals and the ground truth of the test set (See Table 4). Lower values indicate more accurate predictions.

**Table 4.** MAPE for our model and baselines on two test sets

| | MF | PMF | FPMC | Our |
|---|---|---|---|---|
| Fours.-NYC | 14.87 | 12.64 | 6.72 | 1.84 |
| Gowalla | 16.95 | 14.12 | 7.89 | 2.15 |

It is evident that the proposed model outperforms the baselines by a significant margin. Fig.5 shows the performance comparison by relaxing the threshold $T$, and our method outperforms all the baselines again[7].

## 5 Conclusion

This paper proposes a probabilistic approach for next POI recommendation by exploring the transition interval patterns of each user, and the expected transition interval for next move can also be learned simultaneously. Specifically, the proposed model considers the transition interval preference as a key component of the overall transition behavior and utilizes a continuous latent variable to model such preference. The objective function is to maximize the posterior probability of the overall transitions. Expectation Maximization (EM) algorithm is used to estimate the model parameters. The experimental results on the real-world datasets show that the proposed model outperforms the state-of-the-art methods in terms of next POI recommendation, next new POI recommendation, and the expected transition interval prediction. Our proposed model can be easily extended to recommend next basket items and the purchase interval of them by redefining the transition tensor. In addition, we look forward to deploying the proposed model for commercial purposes, such as Weixin App, where the POI recommendation is able to benefit the users and location-based service providers. In the future, we would like to study how to integrate other contextual information into our model, e.g. social relationship and textual content of POIs, which may deeply exploit users check-in behavior to enhance the recommendation performance.

## Acknowledgments

This work has been partially supported by National Key R&D Program of China under Grant No. 2017FB0803300, NSFC under Grant No. 61772074.

## References

1. Bao, J., Zheng, Y., Mokbel, M.F.: Location-based and preference-aware recommendation using sparse geo-social networking data. In: Proceedings of the 20th International Conference on Advances in Geographic Information Systems. pp. 199–208. ACM (2012)

[7] Additional results reports in the Appendix of [16], which leads to similar evaluation conclusions.

2. Basilevsky, A.T.: Statistical factor analysis and related methods: theory and applications, vol. 418. John Wiley & Sons (2009)
3. Cheng, C., Yang, H., King, I., Lyu, M.: Fused matrix factorization with geographical and social influence in location-based social networks. In: Twenty-Sixth AAAI Conference on Artificial Intelligence (2012)
4. Cheng, C., Yang, H., Lyu, M.R., King, I.: Where you like to go next: Successive point-of-interest recommendation. In: Proceedings of the Twenty-Third international joint conference on Artificial Intelligence. pp. 2605–2611. AAAI Press (2013)
5. Cho, E., Myers, S.A., Leskovec, J.: Friendship and mobility: User movement in location-based social networks. In: Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. pp. 1082–1090. KDD '11, ACM, New York, NY, USA (2011)
6. Feng, S., Li, X., Zeng, Y., Cong, G., Chee, Y.M., Yuan, Q.: Personalized ranking metric embedding for next new poi recommendation. In: Proceedings of the 24th International Conference on Artificial Intelligence. pp. 2069–2075. AAAI Press (2015)
7. Gao, H., Tang, J., Hu, X., Liu, H.: Exploring temporal effects for location recommendation on location-based social networks. In: Proceedings of the 7th ACM conference on Recommender systems. pp. 93–100. ACM (2013)
8. Gao, H., Tang, J., Hu, X., Liu, H.: Content-aware point of interest recommendation on location-based social networks. In: Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, January 25-30, 2015, Austin, Texas, USA. pp. 1721–1727 (2015)
9. He, J., Li, X., Liao, L., Song, D., Cheung, W.K.: Inferring a personalized next point-of-interest recommendation model with latent behavior patterns. In: Thirtieth AAAI Conference on Artificial Intelligence (2016)
10. Koren, Y., M.Bell, R., Volinsky, C., Jane, D.: Matrix factorization techniques for recommender systems. vol. 42, pp. 30–37 (Aug 2009)
11. Li, X., Jiang, M., Hong, H., Liao, L.: A time-aware personalized point-of-interest recommendation via high-order tensor factorization. ACM Transactions on Information Systems (TOIS) **35**(4), 31 (2017)
12. Liu, Y., Liu, C., Liu, B., Qu, M., Xiong, H.: Unified point-of-interest recommendation with temporal interval assessment. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. pp. 1015–1024. ACM (2016)
13. Mnih, A., Salakhutdinov, R.: Probabilistic matrix factorization. In: Advances in neural information processing systems. pp. 1257–1264 (2007)
14. Rendle, S., Freudenthaler, C., Schmidt-Thieme, L.: Factorizing personalized markov chains for next-basket recommendation. In: Proceedings of the 19th international conference on World wide web. pp. 811–820. ACM (2010)
15. Rubin, D.B., Thayer, D.T.: Em algorithms for ml factor analysis. Psychometrika **47**(1), 69–76 (1982)
16. Supple.: https://github.com/anonymityabcd/abcd1/blob/master/paper73sup.pdf (2018)
17. Yuan, Q., Cong, G., Ma, Z., Sun, A., Thalmann, N.M.: Time-aware point-of-interest recommendation. In: Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval. pp. 363–372. ACM (2013)
18. Yuan, T., Cheng, J., Zhang, X., Qiu, S., Lu, H., et al.: Recommendation by mining multiple user behaviors with group sparsity. In: AAAI. pp. 222–228 (2014)