Geo-Teaser: Geo-Temporal Sequential Embedding Rank for Point-of-interest Recommendation

Shenglin Zhao[1,2], Tong Zhao[1,2], Irwin King[1,2], and Michael R. Lyu[1,2]
[1]Shenzhen Key Laboratory of Rich Media Big Data Analytics and Application
Shenzhen Research Institute, The Chinese University of Hong Kong
Nanshan District, Shenzhen, China
[2]Department of Computer Science & Engineering
The Chinese University of Hong Kong
Shatin, N.T., Hong Kong
{slzhao, tzhao, king, lyu}@cse.cuhk.edu.hk

ABSTRACT

Point-of-interest (POI) recommendation is an important application for location-based social networks (LBSNs), which learns the user preference and mobility pattern from check-in sequences to recommend POIs. Previous studies show that modeling the sequential pattern of user check-ins is necessary for POI recommendation. Markov chain model, recurrent neural network, and the word2vec framework are used to model check-in sequences in previous work. However, all previous sequential models ignore the fact that check-in sequences on different days naturally exhibit the various temporal characteristics, for instance, "work" on `weekday` and "entertainment" on `weekend`. In this paper, we take this challenge and propose a **Geo**-**Te**mpor**a**l **s**equential **e**mbedding **r**ank (Geo-Teaser) model for POI recommendation. Inspired by the success of the word2vec framework to model the sequential contexts, we propose a *temporal POI embedding* model to learn POI representations under some particular temporal state. The temporal POI embedding model captures the contextual check-in information in sequences and the various temporal characteristics on different days as well. Furthermore, We propose a new way to incorporate the geographical influence into the pairwise preference ranking method through discriminating the unvisited POIs according to geographical information. Then we develop a geographically hierarchical pairwise preference ranking model. Finally, we propose a unified framework to recommend POIs combining these two models. To verify the effectiveness of our proposed method, we conduct experiments on two real-life datasets. Experimental results show that the Geo-Teaser model outperforms state-of-the-art models. Compared with the best baseline competitor, the Geo-Teaser model improves at least 20% on both datasets for all metrics.

Keywords

location-based services, POI recommendation, embedding learning, spatial-temporal data

WWW 2017, April 3–7, 2017, Perth, Australia.
ACM 978-1-4503-4913-0/17/04.
http://dx.doi.org/10.1145/3038912.3038914

1. INTRODUCTION

Location-based social networks (LBSNs) such as Foursquare and Facebook Places have become popular services to attract users sharing their check-in behaviors, making friends, and writing comments on point-of-interests (POIs). With the prosperity of LBSNs, POI recommendation comes out to improve the user experience, which mines users' check-in sequences and recommends places where an individual has not been. POI recommendation not only helps users explore new interesting places in a city, but also facilitates business owners to launch advertisements to target customers. Due to the significance for users and businesses, POI recommendation has attracted much academic attention, and thus a bunch of methods has been proposed to enhance the POI recommendation system [2, 9, 31, 32].

Modeling the sequential pattern of user check-ins is necessary for POI recommendation. Because successive check-ins are usually correlated [3, 17, 30, 35]. Markov chain model, recurrent neural network, and the word2vec framework are used to model the check-in sequences in previous work. Studies in [17, 30, 35] exploit the Markov chain model to capture the successive check-ins' transitive pattern. Besides, researchers in [3, 5, 39] use the latent factor model based on the Markov chain property to model the successive check-ins' correlations. Recently, inspired by the success of deep learning, the neural network has been used to model the check-in sequences. Liu et al. [16] employ the recurrent neural network (RNN) to find the sequential correlations. The work in [18] models the check-in sequences through the word2vec framework to capture the sequential contexts. Moreover, we observe that check-in sequences on different days naturally exhibit the various temporal characteristics. For example, users always check-in at POIs around offices on `weekday` while visit shopping malls on `weekend`. However, all previous sequential models ignore the various temporal characteristics, which motivates this paper.

Inspired by the success of the word2vec framework to model the sequential contexts [18], we propose a temporal POI embedding model to capture the contextual check-in information and the various temporal characteristics as well. In [18], all POIs are built as the "corpus", each POI is treated as a "word", and a user's all sequential check-ins are treated as a "sentence". Then, the word2vec framework [22] can be used to learn the POI embeddings, which contain the contextual relationships of consecutively visited POIs, showing better performance than Markov chain model. Nevertheless, the learned POI embeddings for capturing the sequential contexts cannot subsume the various temporal characteristics on different days. Moreover, the geographical influence is not considered

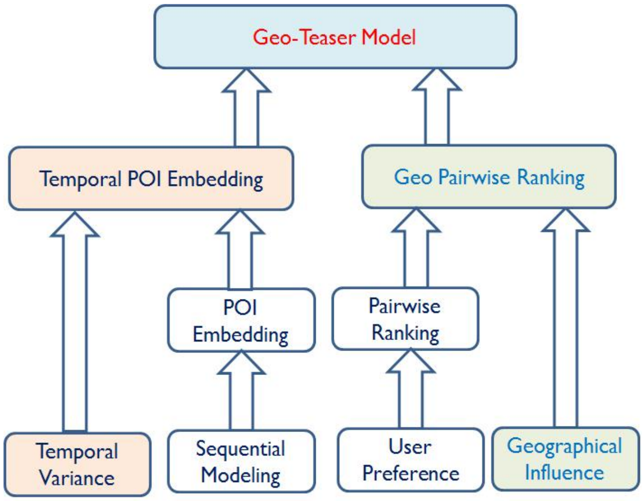


Figure 1: Framework of the Geo-Teaser model

in [18]. Studies on user mobility data show that the geographical influence is the most significant factor for POI recommendation [31, 34, 37]. Therefore, the geographical influence is expected to be incorporated to improve the POI recommendation.

To sum up, we propose a ***Geo-Te***mpor***a***l ***s***equential ***e***mbedding ***r***ank (Geo-Teaser) model for POI recommendation, as shown in Figure 1. On the one hand, we propose a temporal POI embedding model to capture the contextual check-in information and the various temporal characteristics as well. In particular, we treat one user's check-in sequence in one day as a "sentence". Then we consider each sequence under a specific temporal state and define the *temporal POI*, referring to a POI taking a specific temporal state as context. Further, we propose the temporal POI embedding model to learn POI representations and temporal state representations. On the other hand, we incorporate the geographical influence into a pairwise preference ranking model and develop a geographically hierarchical pairwise preference ranking model. Traditionally, we assume users prefer the visited POIs than the unvisited and establish a pairwise ranking model to learn user preference on POIs [13, 39]. Previous studies [2, 31] indicate that users prefer POIs that are geographically adjacent to their visited POIs. This geographical characteristic inspires us to boost the traditional pairwise ranking model through hierarchical pairwise preference relations that discriminate the unvisited POIs according to POIs' geographical information. Finally, we propose the Geo-Teaser model as a unified framework to recommend POIs combining the temporal POI embedding model and the geographically hierarchical pairwise ranking model. We summarize the contributions as follows:

1. We propose the temporal POI embedding model, which captures the check-ins' sequential contexts and the various temporal characteristics on different days. In particular, we introduce the *word2vec* framework to project every POI as one object in an embedding space for learning the sequential relations among POIs. Furthermore, we learn the temporal POI representations from the check-in sequence under some specific temporal state.

2. We propose a new way to incorporate the geographical influence into the pairwise preference ranking method through discriminating the unvisited POIs according to geographical information. In particular, we define a hierarchical pairwise preference relation for each user check-in: the user prefers the visited POI than the unvisited neighboring POI, and the user prefers the unvisited neighboring POI than the unvisited non-neighboring POI. Then we learn the hierarchical pairwise preference to capture the geographical influence and user preference.

3. We propose the Geo-Teaser model as a unified framework combining the temporal POI embedding model and the geographically hierarchical pairwise preference ranking model. Experimental results on two real-life datasets show that the Geo-Teaser model outperforms state-of-the-art models. Compared with the best baseline competitor, the Geo-Teaser model improves at least 20% on both datasets for all metrics.

The rest of this paper is organized as follows. In Section 2, we review the related work. In Section 3, we introduce two real-world datasets and report empirical data analysis that motivates our method. Next, we introduce our proposed Geo-Teaser model and show the learning algorithm in Section 4. Then, we evaluate our proposed model in Section 5. Finally, we conclude this paper and point out possible future work in Section 6.

2. RELATED WORK

In this section, we first demonstrate the recent progress of POI recommendation. Then we report how the prior work exploits the sequential influence and geographical influence to improve the POI recommendation. Since our proposed method adopts an embedding learning method, the word2vec framework, to model check-in sequences, we also review the literature of the word2vec framework and its applications.

POI Recommendation. POI recommendation has attracted intensive academic attention recently. Most of the proposed methods base on Collaborative Filtering (CF) techniques to learn user preference on POIs. On the one hand, the studies in [31, 33] employ the memory-based CF to recommend POIs. The proposed system first finds some users sharing the similar check-in preference with the target user and then recommends POIs where the "similar" users have checked-in but the target user has not. Furthermore, the researchers attempt to analyze the user check-in behavior and incorporate the spatial and temporal influence to improve the recommendation performance. On the other hand, some other studies in [2, 6, 7, 12] leverage the model-based CF, i.e., the Matrix Factorization (MF) technique. They treat the POI as "item" and the check-in frequency as "rating" and establish a user-POI matrix to recommend POIs using traditional MF models. Moreover, the researchers in [14, 20] observe that it is better to treat the check-ins as implicit feedback than explicit way, namely the check-ins are similar to clicks on Webs rather than the rating on Movies. They utilize the weighted regularized MF [10] to model this kind of implicit feedback. In addition, recent work in [13, 39, 38] employs pairwise ranking models to learn the user check-in as an implicit feedback and shows the advantages of ranking methods.

Sequential Modeling. Modeling the sequential pattern is important for POI recommendation. Most of the studies employ the Markov chain property in consecutive check-ins to capture the sequential pattern. We usually categorize the POI recommendation system as generic POI recommendation and successive POI recommendation by subtle differences in the recommendation task whether to be biased to the recent check-in. The successive POI recommendation is proposed to recommend POIs given the recent check-in, which naturally attempts to model the sequential pattern from successive check-ins [3, 5, 34, 16]. Also, researchers leverage the sequential modeling to improve the generic POI recommendation. The studies in [17, 30] learn the categories' transitive pattern in sequential check-ins. Zhang et al. [35] recommend POIs by learning the transitive probability through an additive Markov chain. Recently, inspired by the success of deep learning, the neural network has been used to model the check-in sequences. Liu et al. [16] employ the recurrent neural network (RNN) to find the sequential correlations among POIs. In the meantime, the work [18] models the check-in sequences through the word2vec framework [22] to capture the sequential contexts. The success in the prior work [17, 30, 35, 16, 18] motivates us to capture the sequential pattern in user check-ins to improve the generic POI recommendation. However, all previous sequential models ignore the various temporal characteristics. Hence, we propose a temporal POI embedding method to capture the sequential POIs' correlations under different temporal states.

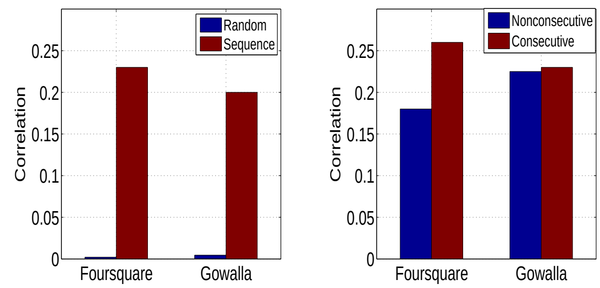
Geographical Influence. Geographical influence plays an important role in POI recommendation. Compared with watching movies on Netflix and online shopping in Amazon, the check-in activity is limited to the physical constraint. Hence, the check-ins usually occur in the POIs nearby the user's home and working place. This observation motivates researchers to capture the geographical influence to improve the POI recommendation.

On the one hand, researchers attempt to establish geographical models to recommend POIs. First, researchers in [31, 33] discover that the distances for each pair of visited POIs in the LBSN follow the power law distribution after analyzing the geographical relations among visited POIs. Then, they propose a power law distribution model to fit the spatial relations among POIs and recommend POIs according to this kind of geographical influence [31, 33]. Moreover, researchers in [4, 2, 37] analyze each user's check-ins rather than all visited POIs and propose the Gaussian distribution based models to capture the geographical influence. Recently, Zhang et al. [34, 36] have observed that each user occupies a group of special parameters in the Gaussian mixture model. Then, they leverage the kernel density estimation to model each user's check-ins for personalization. On the other hand, instead of independently modeling the geographical influence, more researchers attempt to jointly model the geographical influence and other factors such as user preference and temporal influence together. The studies in [14, 20] incorporate the geographical influence into a weighted regularized MF model [10, 24] to learn the geographical influence and user preference together. Similar to [14, 20], we model the check-ins as a kind of implicit feedback. But we learn it through a Bayesian pairwise ranking method [26] due to its success in [39]. Furthermore, we propose a geographically hierarchical pairwise ranking model, which captures the geographical influence via discriminating the unvisited POIs according to their geographical information.

Embedding Learning. The word2vec framework [22] is an effective neural language model to learn embedding representations in word sequences. The key idea is to learn the sentence as the bag of words and represent the relations among words in the embedding subspace, such as "male"-"female"+"queen" = "king". The embedding learning technique in the word2vec framework attempts to capture the words' contextual correlations in sentences, showing better performance than the perspectives of word transitivity in sentences and word similarity. As a result, the embedding learning technique has been widely used in natural language processing recently [21, 23]. Afterwards, paragraph2vector [11] and other variants [15, 19] are proposed to enhance the word2vec framework for specific purposes. Since the efficacy of the framework in capturing the contextual correlations of items, the embedding technique based on the word2vec framework is employed to network embedding [1], as well as in user modeling [28] and item modeling [27]. To take the power of embedding learning for POI recommendation, Liu et al. [18] model the sequential contexts through a Skip-Gram model and achieves better performance than Markov chain model. Xie et al. [29] use similar embedding technique to recommend POIs. However, the previous work [18, 29] ignores two significant factors accounting for the check-in activity, the various temporal characteristics and geographical influence. To incorporate these two factors, we propose the Geo-Teaser model.

Table 1: Data Statistics

	Foursquare	Gowalla
#users[1]	10,034	3,240
#POIs	16,561	33,578
#check-ins	865,647	556,453
Avg. #check-ins of each user[2]	86.3	171.7
Avg. #POIs for each user	24.6	95.4
Avg. #users for each POI	14.9	9.2
Density	0.0015	0.0028



(a) Sequence vs. Random (b) Consecutive vs. Nonconsecutive

Figure 2: POI correlation in sequences

3. DATA DESCRIPTION AND ANALYSIS

In this section, we first introduce two real-world LBSN datasets and then conduct the empirical analysis to explore the properties of check-in sequences in one day.

3.1 Data Description

We use two check-in datasets crawled from real-world LBSNs for data analysis. One is collected from Foursquare provided in [8] and the other is Gowalla data provided in [37]. We preprocess the data by filtering the POIs checked-in less than five users and users whose check-ins are less than ten times. Then we keep the remaining users' check-in records from January 1, 2011 to July 31, 2011. After the preprocessing, the datasets contain the statistical properties as shown in Table 1.

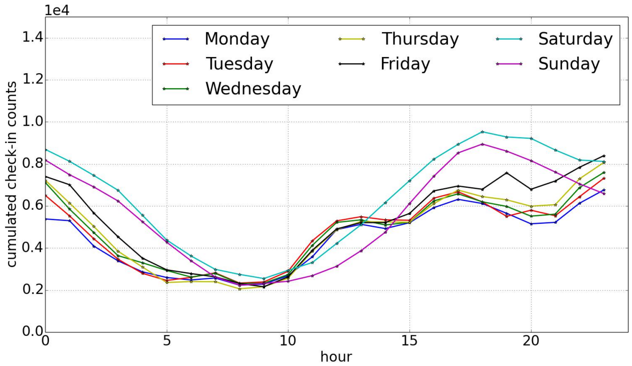
3.2 Empirical Analysis

We conduct data analysis to answer the following two questions: 1) how POIs in sequences of one day correlate each other? 2) how check-in sequences perform on different days?

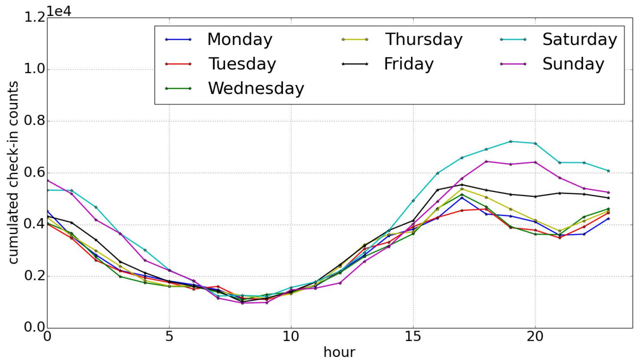
We investigate the correlations of POIs in sequences of one day, as shown in Figure 2. To calculate the correlation between two

[1] "#users" means the number of users.

[2] "Avg. #check-ins" means the average number of check-ins.



(a) Foursquare



(b) Gowalla

Figure 3: Day of week check-in pattern at different hours

POIs, we construct the user-POI matrix according to the check-in records. Then, we measure the correlation of a POI pair regarding the Jaccard similarity of those users who have checked-in at the two POIs. In Figure 2(a), we calculate the average correlation value of POI pairs in sequences for all users and compare it with the average correlation value of 5,000 random POI pairs. We observe that the correlation of POIs in sequences is much higher than random pairs by about 100 times for Foursquare and 50 times for Gowalla, which motivates the sequential modeling. In Figure 2(b), we compare the correlation of consecutive pairs with nonconsecutive pairs in sequences. Take a sequence of (l_1, l_2, l_3) as an example, (l_1, l_2) and (l_2, l_3) are consecutive pairs, and (l_1, l_3) is a nonconsecutive pair. We also calculate the average value of all sequences for all users to make the comparison. We observe that the nonconsecutive pairs contain comparable correlation to the consecutive pairs. Hence, not only consecutive POIs are highly correlated [3, 39], all POIs in a sequence are highly correlated with a contextual property. Accordingly, it is not satisfactory to only model the consecutive check-ins' transitive probability by Markov chain model or the consecutive check-ins' correlation by tensor factorization. This observation motivates us to model the whole sequence through the word2vec framework.

We explore how the various temporal characteristics on different days affect the user's check-in behavior. Previous work [38, 39] shows that user check-ins exhibit different patterns on different days, especially for working days and weekends. Figure 3 demonstrates the number of cumulated check-ins for all users at different hours on different days of a week, from Monday to Sunday. From the statistics of cumulated check-ins in Figure 3, we observe the day of week check-in pattern at different hours: users take more check-ins in the late afternoon and the evening from 16:00 p.m. to 3:00 a.m. on weekends than the weekdays. Hence, Saturday and Sunday take the similar pattern, while the days from Monday to Friday take the similar pattern that is different from the weekends. We may infer that `weekday` and `weekend` exert two types of effects on the user's check-in behavior. Therefore, modeling the sequence pattern should contain this temporal feature.

Table 2: Notation Descriptions

Notation	Description
u	user name
l	POI name
t_s	temporal state for a sequence
k	context window size
h	negative sample size for embedding learning
m	negative sample size for preference learning
d	latent vector dimension
C	the set of check-ins
U	the set of users
L	the set of POIs
L_u	the set of POIs visited by user u
$d(l_i, l_j)$	the distance between two POIs l_i and l_j
S_u	a sequence for user u
S	the set of all sequences
D_{S_u}	the set of preference relations for S_u
$\mathbf{u}$	user latent feature vector
$\mathbf{l}$	POI latent feature vector
$\mathbf{t}_s$	temporal state latent feature
$\mathbf{l}_i^t$	temporal POI embedding vector
$\mathbf{T}$	temporal state feature matrix
$\mathbf{U}$	user latent feature matrix
$\mathbf{L}$	user latent feature matrix

4. METHOD

In this section, we first propose the temporal POI embedding model to capture the various temporal characteristics for sequential modeling. Next, we demonstrate the geographically hierarchical pairwise preference ranking model. Then, we propose the Geo-Teaser model as a unified framework to recommend POIs combining the temporal POI embedding model and the geographically hierarchical pairwise preference ranking model. Finally, we show the learning procedures for the Geo-Teaser model. In order to help understand the paper, we list some important notations in Table 2.

4.1 Temporal POI Embedding

We propose a temporal POI embedding method to learn the sequential pattern, which captures POIs' contextual information from user check-in sequences and as well as the various temporal characteristics. Different from the work [18] that treats a user's all check-ins as a "sentence", we treat a user's check-ins of one day as a "sentence". Because consecutive check-ins on different days may span a long time and be not highly correlated. Further, we assume that check-in sequences on different days exhibit various temporal characteristics. Then, we learn POI embeddings in a sequence with some specific temporal state.

To better describe the model, we present some basic concepts as follows.

DEFINITION 1 (CHECK-IN). *A check-in is a triple $\langle u, l, t\rangle$ that depicts a user u visiting POI l at time t.*

DEFINITION 2 (CHECK-IN SEQUENCE). *A check-in sequence is a set of check-ins of user u in one day, denoted as $S_u = \{\langle l_1, t_1\rangle,$*

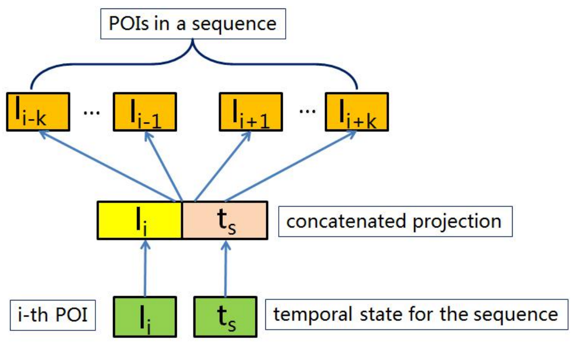


Figure 4: Temporal POI embedding model

..., $\langle l_n, t_n\rangle\}$, where t_1 to t_n belong to the same day. For simplicity, we denote $S_u = \{l_1, \ldots, l_n\}$.

DEFINITION 3 (TARGET POI AND CONTEXT POI). *In a sequence S_u, the chosen l_i is the target POI and other POIs in S_u are context POIs.*

We propose the temporal POI embedding model based on the Skip-Gram model [22]. As shown in Figure 4, we learn the representations of context POIs from l_{i-k} to l_{i+k} given a target POI l_i and the sequence temporal state t_s. Here k is a parameter to control the context window size. In addition, the temporal state t_s is composed of two options, `weekday` and `weekend`. Because we want to discriminate `weekday` and `weekend`, which depict the various temporal characteristics on day level as shown in Figure 3. Formally, given a sequence S_u and its temporal state t_s, our model attempts to learn the temporal POI embeddings through maximizing the following function,

$$\mathcal{L}_{TPE} = \sum_{S_u \in S} \frac{1}{|S_u|} \sum_{l_i \in S_u} \sum_{-k \leq c \leq k, c \neq 0} \big(\log \Pr(l_{i+c}|l_i, t_s)\big), \quad (1)$$

where S is a set containing all sequences S_u for all users. $\mathcal{L}_{TPE}$ aims to maximize the context POI's conditional occurrence likelihood for all sequences.

Furthermore, we formulate the probability $\Pr(l_{i+c}|l_i, t_s)$ using a softmax function. For better description, we introduce two symbols, defined as follows: $\hat{\mathbf{l}}'_c = \mathbf{l}'_c \oplus \mathbf{l}'_c$, $\mathbf{l}^t_i = \mathbf{l}_i \oplus \mathbf{t}_s$, where $\oplus$ is the concatenation operator, and $\mathbf{l}'_c$, $\mathbf{l}_i$, and $\mathbf{t}_s$ are latent vectors of output layer context POI, target POI, and temporal state, respectively. Thus, we get $\hat{\mathbf{l}}'_c \cdot \mathbf{l}^t_i = \mathbf{l}'_c \cdot \mathbf{l}_i + \mathbf{l}'_c \cdot \mathbf{t}_s$. Therefore, the probability $\Pr(l_{i+c}|l_i, t_s)$ can be formulated as,

$$\Pr(l_{i+c}|l_i, t_s) = \frac{\exp(\hat{\mathbf{l}}'_c \cdot \mathbf{l}^t_i)}{\sum_{l_i \in L} \exp(\hat{\mathbf{l}}'_c \cdot \mathbf{l}^t_i)}. \quad (2)$$

As the size of set L in Eq. (2) is large, we exploit the negative sampling technique [22] to learn the model efficiently. Then, the objective function can be formulated in a new form easier to optimize,

$$\begin{aligned}\mathcal{L}_{TPE} = \sum_{S_u \in S} \frac{1}{|S_u|} \sum_{l_i \in S_u} \sum_{-k \leq c \leq k, c \neq 0} \big(\log \sigma(\hat{\mathbf{l}}'_c \cdot \mathbf{l}^t_i) + \\ \sum_{h} E_{k'} \log \sigma(-\hat{\mathbf{l}}'_{k'} \cdot \mathbf{l}^t_i)\big),\end{aligned} \quad (3)$$

where $l_{k'}$ is the sampled negative POI, h is the number of negative samples, $\sigma(\cdot)$ is the sigmoid function, and $E(\cdot)$ means to calculate the expectation value for all generated negative samples. Here we adopt the same strategy in [22], namely using a unigram distribution, to draw the negative samples.

4.2 Geographically Hierarchical Pairwise Ranking

We propose the geographically hierarchical pairwise preference ranking model, which incorporates the geographical influence into a pairwise ranking model. The check-in activity is observed as a kind of implicit feedback similar to the web clicks [14, 20]. To learn this implicit feedback, we leverage the Bayesian personalized ranking (BPR) criteria [26] to learn the user preference on POIs. BPR is a pairwise ranking model, which learns the pairwise user preference based on the assumption that users prefer the visited POIs than the unvisited. In our geographically hierarchical pairwise ranking model, we discriminate the unvisited POIs using POIs' geographical information. Previous studies [2, 33, 37] observe that users prefer the POIs nearby the visited than POIs far away, we can discriminate the unvisited POIs and define *neighboring POI* and *non-neighboring POI* as follows.

DEFINITION 4 (NEIGHBORING POI AND NON-NEIGHBORING POI). *For each check-in $\langle u, l_i\rangle$, the neighboring POI is the POI whose distance from l_i is less than or equal to a threshold s, while the non-neighboring POI is the POI whose distance is more than s.*

Furthermore, for each check-in $\langle u, l_i\rangle$, we define a hierarchical pairwise preference relation: the user prefers the visited POI l_i than the unvisited neighboring POI l_{ne}, and prefers the unvisited neighboring POI l_{ne} than the unvisited non-neighboring POI l_{nn}. Denote $d(l_i, l_j)$ as the distance of two POIs l_i and l_j, we represent the hierarchical pairwise preference relation for check-in $\langle u, l_i\rangle$ as follows,

$$l_i >_{u, d(l_i, l_{ne}) \leq s} l_{ne} \ \vee \ l_{ne} >_{u, d(l_i, l_{nn}) > s} l_{nn}. \quad (4)$$

Suppose L is the set of POIs, and L_u is the visited POIs of user u, the hierarchical pairwise preference relation set for a sequence S_u satisfying Eq. (4) is defined as follows,

$$\begin{aligned}D_{S_u} = \{(u, l_i, l_{ne}) \vee (u, l_{ne}, l_{nn}) | l_i \in S_u, d(l_i, l_{ne}) \leq s, \\ d(l_i, l_{nn}) > s, l_{ne}, l_{nn} \in L \setminus L_u\}.\end{aligned} \quad (5)$$

Now learning the geographically hierarchical pairwise ranking model is equivalent to model the preference relations in D_{S_u}. Here we employ the MF model to formulate the preference score function. We use $\mathbf{l}^t_i = \mathbf{l}_i \oplus \mathbf{t}_s$ to represent the temporal POI latent vector, which is consistent with the temporal POI embedding model. In addition, we define $\hat{\mathbf{u}} = \mathbf{u} \oplus \mathbf{u}$, then the score function can be formulated as,

$$f(u, t_s, l_i) = \hat{\mathbf{u}} \cdot \mathbf{l}^t_i. \quad (6)$$

Next, we use the sigmoid function to formulate the pairwise preference probability. Suppose $\Pr(l_i >_u l_n)$ denotes the probability of user u prefers POI l_i than l_n, and $\sigma(\cdot)$ is the sigmoid function. Then, each pair in the preference set can be formulated as,

$$\Pr(l_i >_u l_n) = \sigma(f(u, t_s, l_i) - f(u, t_s, l_n)) = \sigma(\mathbf{u} \cdot (\mathbf{l}_i - \mathbf{l}_n)). \quad (7)$$

Thus, learning the geographically hierarchical pairwise ranking model is equivalent to maximize the following function,

$$\mathcal{L}_{GPR} = \sum_{S_u \in S} \sum_{(u, l_i, l_n) \in D_{S_u}} \log \sigma(\mathbf{u} \cdot (\mathbf{l}_i - \mathbf{l}_n)), \quad (8)$$

where S is a set containing all sequences S_u for all users and D_{S_u} is hierarchical pairwise preference relations on sequence S_u.

4.3 Geo-Teaser Model

We propose the Geo-Teaser model as a unified framework to recommend POIs combining the temporal embedding model and the pairwise ranking model. Learning the Geo-Teaser model is equivalent to maximize $\mathcal{L}_{TPE}$ and $\mathcal{L}_{GPR}$ together,

$$\mathcal{O} = \underset{\mathbf{U},\mathbf{L},\mathbf{T}}{\arg\max}\, \alpha \cdot \mathcal{L}_{TPE} + \beta \cdot \mathcal{L}_{GPR}, \tag{9}$$

where α and β are the hyperparameters to trade-off the sequential modeling and the preference learning modules. We expect to obtain the user, POI, and temporal state representations through learning the temporal POI embeddings and geographically pairwise preference relations in the Geo-Teaser model.

Substituting $\mathcal{L}_{TPE}$ and $\mathcal{L}_{GPR}$ with Eq. (3) and Eq. (8) respectively, then we can learn the Geo-Teaser model through the following objective function,

$$\begin{aligned} \underset{\mathbf{U},\mathbf{L},\mathbf{T}}{\arg\max} \sum_{S_u \in S} \sum_{l_i \in S_u} \Big(\sum_{-k \le c \le k, c \ne 0} & \alpha \log \sigma(\mathbf{l}'_c \cdot \mathbf{l}_i) + \\ \sum_{h} \alpha E_{k'} \log \sigma(-\mathbf{l}'_{k'} \cdot \mathbf{l}_i) & + \\ \sum_{D_{S_u}} \beta \log(\sigma(\mathbf{u} \cdot (\mathbf{l}_i - \mathbf{l}_n))) \Big). & \end{aligned} \tag{10}$$

4.4 Learning

We use an alternate iterative update procedure and employ stochastic gradient descent (SGD) to learn the objective function. To learn the model, for each sampled training instance, we separately calculate the derivatives for $\mathcal{L}_{TPE}$ and $\mathcal{L}_{GPR}$, and then update the corresponding parameters along the ascending gradient direction,

$$\Theta^{t+1} = \Theta^t + \eta \times \frac{\partial \mathcal{O}(\Theta)}{\partial \Theta}, \tag{11}$$

where Θ is the training parameter and η is the learning rate.

Specifically, for a check-in $\langle u, l_i \rangle$, we calculate the stochastic gradient decent for $\mathcal{L}_{TPE}$. First, we get the updating rule for the context POI l_c,

$$\begin{aligned} \mathbf{l}_i &\leftarrow \mathbf{l}_i + \alpha\eta(1 - \sigma(\hat{\mathbf{l}}'_c \cdot \mathbf{l}^t_i))\mathbf{l}'_c \\ \mathbf{t}_i &\leftarrow \mathbf{t}_i + \alpha\eta(1 - \sigma(\hat{\mathbf{l}}'_c \cdot \mathbf{l}^t_i))\mathbf{l}'_c \\ \mathbf{l}'_c &\leftarrow \mathbf{l}'_c + \alpha\eta(1 - \sigma(\hat{\mathbf{l}}'_c \cdot \mathbf{l}^t_i))(\mathbf{l}_i + \mathbf{t}_i). \end{aligned} \tag{12}$$

Then, we update the negative sample l'_k as follows,

$$\begin{aligned} \mathbf{l}_i &\leftarrow \mathbf{l}_i - \alpha\eta\sigma(\hat{\mathbf{l}}'_{k'} \cdot \mathbf{l}^t_i)\mathbf{l}'_{k'} \\ \mathbf{t}_i &\leftarrow \mathbf{t}_i - \alpha\eta\sigma(\hat{\mathbf{l}}'_{k'} \cdot \mathbf{l}^t_i)\mathbf{l}'_{k'} \\ \mathbf{l}'_{k'} &\leftarrow \mathbf{l}'_{k'} - \alpha\eta\sigma(\hat{\mathbf{l}}'_{k'} \cdot \mathbf{l}^t_i)(\mathbf{l}_i + \mathbf{t}_i). \end{aligned} \tag{13}$$

To update $\mathcal{L}_{GPR}$, we calculate the stochastic gradient decent for each preference pair (u, l_i, l_n) in D_{S_u}[3]. Denote $\delta = 1 - \sigma(\mathbf{u} \cdot \mathbf{l}_i - \mathbf{u} \cdot \mathbf{l}_n)$, we update the parameters as follows,

$$\begin{aligned} \mathbf{u} &\leftarrow \mathbf{u} + \beta\eta\delta(\mathbf{l}_i - \mathbf{l}_n) \\ \mathbf{l}_i &\leftarrow \mathbf{l}_i + \beta\eta\delta\mathbf{u} \\ \mathbf{l}_n &\leftarrow \mathbf{l}_n - \beta\eta\delta\mathbf{u}. \end{aligned} \tag{14}$$

Algorithm 1 shows the details of learning the Geo-Teaser model. S is the set of all sequences, and S_u is a sequence of user u. $\mathbf{U}$, $\mathbf{L}$, and $\mathbf{T}$ are feature matrices of the user, POI, and temporal state.

[3]The pair of (u, l_i, l_n) happens in two cases: (u, l_i, l_{ne}) and (u, l_{ne}, l_{nn}) as shown in Alg. 1.

Algorithm 1: Learning algorithm for the Geo-Teaser model

Input: S
Output: $\mathbf{U}$, $\mathbf{L}$, $\mathbf{T}$
1 Initialize $\mathbf{U}$, $\mathbf{L}$, $\mathbf{L}'$, and $\mathbf{T}$ (uniformly at random)
2 **for** *iterations* **do**
3   **for** $S_u \in S$ **do**
4   **for** $\langle u, l_i \rangle \in S_u$ **do**
5   **for** *each context POI* l_c **do**
6         $\mathbf{l}_i \leftarrow \mathbf{l}_i + \alpha\eta(1 - \sigma(\hat{\mathbf{l}}'_c \cdot \mathbf{l}^t_i))\mathbf{l}'_c$
7         $\mathbf{t}_i \leftarrow \mathbf{t}_i + \alpha\eta(1 - \sigma(\hat{\mathbf{l}}'_c \cdot \mathbf{l}^t_i))\mathbf{l}'_c$
8         $\mathbf{l}'_c \leftarrow \mathbf{l}'_c + \alpha\eta(1 - \sigma(\hat{\mathbf{l}}'_c \cdot \mathbf{l}^t_i))(\mathbf{l}_i + \mathbf{t}_i)$
9   **for** $\{k' \sim P_{nc_c}\}$ **do**
10           $\mathbf{l}_i \leftarrow \mathbf{l}_i - \alpha\eta\sigma(\hat{\mathbf{l}}'_{k'} \cdot \mathbf{l}^t_i)\mathbf{l}'_{k'}$
11           $\mathbf{t}_i \leftarrow \mathbf{t}_i - \alpha\eta\sigma(\hat{\mathbf{l}}'_{k'} \cdot \mathbf{l}^t_i)\mathbf{l}'_{k'}$
12           $\mathbf{l}'_{k'} \leftarrow \mathbf{l}'_{k'} - \alpha\eta\sigma(\hat{\mathbf{l}}'_{k'} \cdot \mathbf{l}^t_i)(\mathbf{l}_i + \mathbf{t}_i)$
13   **end**
14   **end**
15   Uniformly sample m unvisited POIs
16   **for** $(u, l_i, l_{ne}) \in D_m$ **do**
17         $\delta = 1 - \sigma(\mathbf{u} \cdot \mathbf{l}_i - \mathbf{u} \cdot \mathbf{l}_{ne})$
18         $\mathbf{u} \leftarrow \mathbf{u} + \beta\eta\delta(\mathbf{l}_i - \mathbf{l}_{ne})$
19         $\mathbf{l}_i \leftarrow \mathbf{l}_i + \beta\eta\delta\mathbf{u}$; $\mathbf{l}_{ne} \leftarrow \mathbf{l}_{ne} - \beta\eta\delta\mathbf{u}$
20   **end**
21   **for** $(u, l_{ne}, l_{nn}) \in D_m$ **do**
22         $\delta = (1 - \sigma(\mathbf{u} \cdot \mathbf{l}_{ne} - \mathbf{u} \cdot \mathbf{l}_{nn}))$
23         $\mathbf{u} \leftarrow \mathbf{u} + \beta\eta\delta(\mathbf{l}_{ne} - \mathbf{l}_{nn})$
24         $\mathbf{l}_{ne} \leftarrow \mathbf{l}_{ne} + \beta\eta\delta\mathbf{u}$; $\mathbf{l}_{nn} \leftarrow \mathbf{l}_{nn} - \beta\eta\delta\mathbf{u}$
25   **end**
26   **end**
27   **end**
28 **end**

$\mathbf{L}'$, an assistant learning parameter, is the output layer POI matrix in Skip-Gram model. We use the standard way [22] to learn the POI representations in the sequences, as shown from line 5 to line 14 in Algorithm 1. Next, we exploit the Bootstrap sampling to generate m unvisited POIs and then classify the unvisited POIs as neighboring POIs and non-neighboring POIs according to their distances from the visited POI l_i. Then, we establish the pairwise preference set D_m for each check-in $\langle u, l_i \rangle$. Here $D_m = \{(u, l_i, l_{ne}) \vee (u, l_{ne}, l_{nn}) | d(l_i, l_{ne}) \le s, d(l_i, l_{nn}) > s, l_{ne}, l_{nn} \in L \setminus L_u\}$. Then we learn the parameters for each instance in D_m, shown from line 15 to line 25 in Algorithm 1.

After learning the Geo-Teaser model, we get the latent feature representations of users, POIs, and temporal states. Then, we can estimate the check-in possibility of user u over a candidate POI l at temporal state t_s according to the preference score function. Furthermore, we use the Eq. (6) for score estimation. Finally, we rank the candidate POIs and select the top N POIs with the highest estimated possibility values for each user.

Scalability. For one check-in, learning the temporal embedding model costs $O(k \cdot h \cdot d)$, where k, h, and d denote the context window size, the number of negative samples, and the latent vector dimension, respectively. For the pairwise preference learning from line 15 to 25 in Algorithm 1, we sample m unvisited POIs, which can generate maximum $O(m^2)$ pairwise preference tuples. For each check-in, the learning procedures cost $O(m^2 \cdot d)$. Therefore, the complexity of our model is $O\big((k \cdot h + m^2) \cdot d \cdot |C|\big)$, where C is the set of all check-ins. For k, h, m, and d are fixed hyperparameters, the proposed model can be treated as linear in $O(|C|)$. Furthermore, in order to make our model more efficient, we turn to the asynchronous stochastic gradient descent (ASGD) [25] and parallelly run the algorithm in an unlock way. As the check-in frequency distribution of POIs in LBSNs follows a power law [31], this results in a long tail of infrequent POIs, which guarantees to employ the ASGD to parallel the parameter updates.

5. EXPERIMENTAL EVALUATION

We conduct experiments to seek the answers to the following questions: 1) how the Geo-Teaser model performs comparing with state-of-the-art recommendation methods? 2) how each component (i.e., the various temporal characteristics and geographical influence) affects the model performance? 3) how the parameters affect the model performance? [4].

5.1 Experimental Setting

Two real-world datasets are used in the experiment: one is from Foursquare provided in [8] and the other is from Gowalla in [37]. Table 1 demonstrates the statistical information of the datasets. In order to make our model satisfactory to the scenario of recommending for future check-ins, we choose the first 80% of each user's check-ins as training data, the remaining 20% for test data, following [3, 35].

5.2 Performance Metrics

In this work, we compare the model performance through *precision* and *recall*, which are generally used to evaluate a POI recommendation system [6, 13]. To evaluate a top-N recommendation system, we denote the precision and recall as P@N and R@N, respectively. In our POI recommendation task, P@N measures the ratio of recovered POIs to the N recommended POIs, and R@N means the ratio of recovered POIs to the set of POIs in the test data. Then we calculate the average precision and recall over all users for evaluation. Supposing $L_{visited}$ denotes the set of correspondingly visited POIs in the test data, and $L_{N,rec}$ denotes the set of recommended POIs, the definitions of P@N and R@N are formulated as follows,

$$P@N = \frac{1}{|U|}\sum_{u\in U}\frac{|L_{visited}\cap L_{N,rec}|}{N}, \tag{15}$$

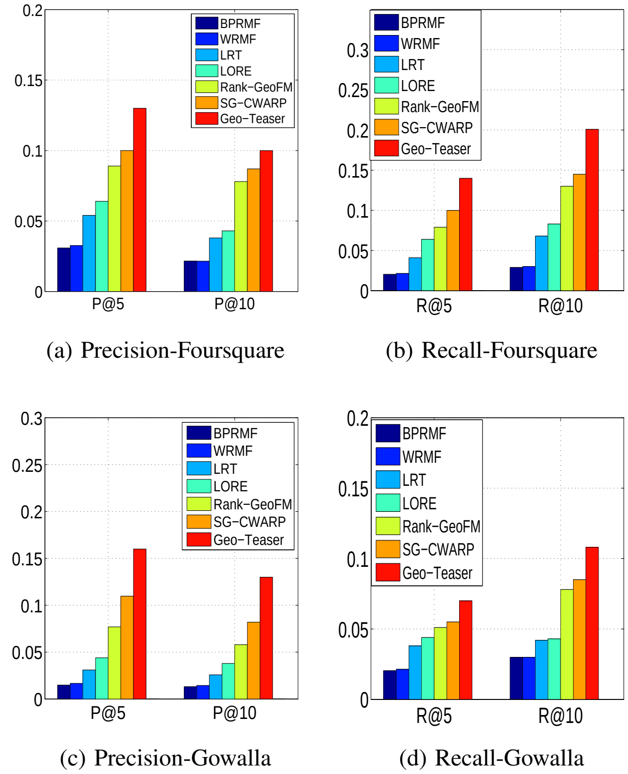
$$R@N = \frac{1}{|U|}\sum_{u\in U}\frac{|L_{visited}\cap L_{N,rec}|}{|L_{visited}|}. \tag{16}$$

5.3 Model Comparison

Prior work [14, 20] observes that treating the check-ins as implicit feedback is better to model the user preference. Hence we compare our model with WRMF [10, 24] and BPRMF [26], which are state-of-the-art collaborative filtering models designed for capturing the implicit feedback. To illustrate the effectiveness of our model, we compare it with four state-of-the-art POI recommendation methods: LRT [6], LORE [35], Rank-GeoFM [13], and SG-CWARP [18].

- ***BPRMF*** [26]: ***B**ayesian **P**ersonalized **R**anking **M**atrix **F**actorization* (*BPRMF*) is a popular pairwise ranking method that models the implicit feedback data to recommend top-N items.

[4]The source code is available from `https://github.com/shenglin1987/geo_teaser`



(a) Precision-Foursquare (b) Recall-Foursquare

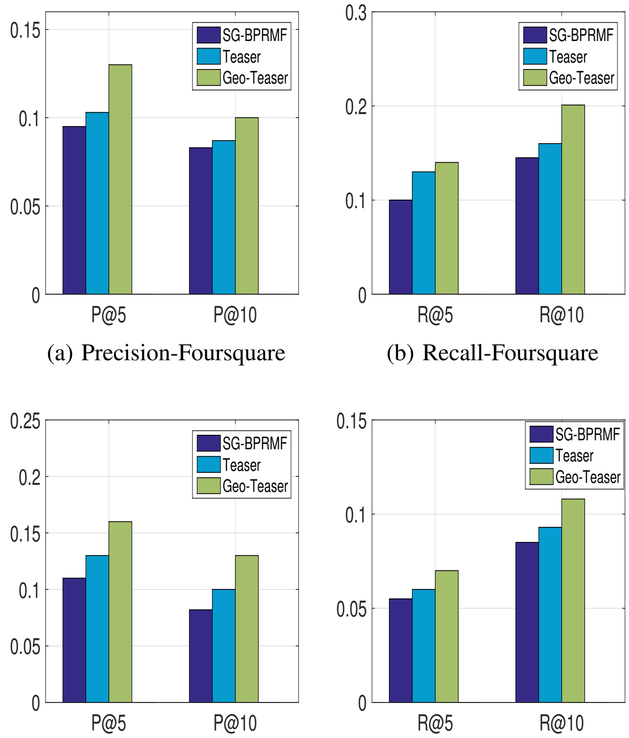
(c) Precision-Gowalla (d) Recall-Gowalla

Figure 5: Model comparison

- ***WRMF*** [10, 24]: ***W**eighted **R**egularized **M**atrix **F**actorization* (*WRMF*) model is designed for implicit feedback ranking problem. We set the weight mapping function of user u_i at POI l_j as $w_{i,j} = (1+10\cdot C_{i,j})^{0.5}$, where $C_{i,j}$ is the number of check-ins, following the setting in [20].
- ***LRT*** [6]: ***L**ocation **R**ecommendation framework with **T**emporal* effects model (*LRT*) is a state-of-the-art POI recommendation method, which captures the temporal effect in POI recommendation.
- ***LORE*** [35]: *LORE* is state-of-the-art model that exploits the sequential influence for location recommendation. Compared with other work [3, 30], *LORE* employs the whole sequence's contribution, not only the successive check-ins sequential influence.
- ***Rank-GeoFM*** [13]: *Rank-GeoFM* is a ranking based geographical factorization method, which incorporates the geographical and temporal influence in a latent ranking model.
- ***SG-CWARP*** [18]. *SG-CWARP* is the latest work, which leverages the word2vec framework to model the check-ins for sequential contexts.

5.4 Experimental Results

In the following, we demonstrate the experimental results on precision and recall, denoted as P@N and R@N, for the top N POI recommendation task. Since the model comparison results are consistent with different values of N, e.g., 1, 5, 10, and 20, we show representative results at 5 and 10 following [6, 7]. All models achieve the best performances at appropriate parameter settings.



(a) Precision-Foursquare (b) Recall-Foursquare

(c) Precision-Gowalla (d) Recall-Gowalla

Figure 6: Demonstration of model component functions

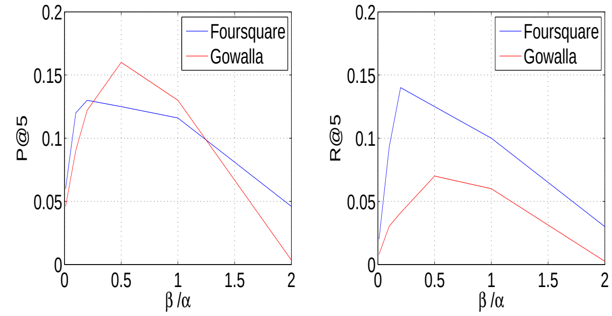
5.4.1 *Performance Comparison*

Figure 5 illustrates the experimental results of different models. We discover that the proposed Geo-Teaser model achieves better performance than all the baselines. Compared with Rank-GeoFM that is a state-of-the-art model incorporating the geographical influence and temporal influence, Geo-Teaser achieves improvements at least 28% on both datasets for all metrics. This verifies the effectiveness of our sequential modeling and as well as the validity of means for incorporating various temporal characteristics and geographical influence. SG-CWARP is the best baseline competitor, which verifies the advantage of modeling the sequential pattern through Skip-Gram model over Markov chain model, namely the LORE model. Our Geo-Teaser model outperforms the SG-CWARP at least 20% on both datasets for all metrics, which verifies our strategy of incorporating various temporal characteristics and geographical influence to improve POI recommendation. In addition, we observe that models perform better on Gowalla than Foursquare for *precision*, but worse for *recall*. The reason lies in that each user's test data size in Gowalla is bigger than Foursquare. As shown in Table 1, the average check-ins for each user in Gowalla is about two times of Foursquare. According to the metrics in Eq. (15) and Eq. (16), the result is reasonable.

5.4.2 *Model Discussion*

In this section, we explore how each component, i.e., the various temporal characteristics and geographical influence, affects the model performance. The Geo-Teaser model improves the SG-CWARP in two aspects, capturing the various temporal characteristics and geographical influence. Ignoring the various temporal characteristics and geographical influence, we propose the **SG-BPRMF** model as the basic version of our proposed Geo-Teaser model. The SG-BPRMF uses the Skip-Gram model to model the sequence and BPRMF to capture the user preference, which is equivalent to SG-CWARP. Furthermore, we incorporate the various temporal characteristics into SG-BPRMF and propose the **Teaser** model. In the following, we compare the SG-BPRMF, Teaser, and Geo-Teaser to show how the various temporal characteristics and geographical influence affect the model.

Figure 6 shows the model performances. We observe that Teaser model improves SG-BPRMF at least about 10% on both datasets for all metrics, which indicates that incorporating the various temporal characteristics improves the model performance. Moreover, the Geo-Teaser model improves the Teaser model at least about 15% on both datasets. It means our strategy of incorporating geographical influence by discriminating the unvisited POIs is valid.



(a) P@5 on Geo-Teaser (b) R@5 on Geo-Teaser

Figure 7: Parameter effect on α and β

5.4.3 *Parameter Effect*

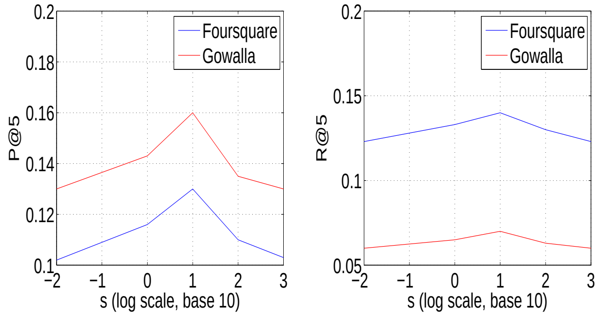
In this section, we show how the three important hyperparameters, α, β, and s affect the model performance. α and β balance the sequential influence and the user preference. s shows the sensitivity of our geographical model.

We tune α and β to see how to trade-off the sequential modeling and user preference learning, shown in Figure 7. Both α and β appear together with the learning rate η in the parameter update procedures. It is not necessary to separately tune the three parameters. We are able to absorb the learning rate η into α and β. In other words, we set $\alpha \leftarrow \alpha \cdot \eta, \beta \leftarrow \beta \cdot \eta$. We avoid to tune the learning rate η, but turn to control the update step size through tuning α and β. Hence α and β should be small enough to guarantee convergence. Assuming the same value for α and β, we tune α to change the learning rate. The model gets the best performance when $\alpha = 0.05$. Then we set $\alpha = 0.05$, and change β to see how the model performance varies with $\frac{\beta}{\alpha}$. Geo-Teaser attains the best performance if $\frac{\beta}{\alpha} \in [0.25, 0.5]$.

In the Geo-Teaser model, we classify the unvisited POIs as neighboring POIs and non-neighboring POIs to constitute a new preference set according to a threshold distance s. Here we choose different values of s to see how this parameter affects the model performance, as shown in Figure 8. Here s is calculated in the kilometer. We observe that the Geo-Teaser model achieves the best performance at $s = 10$.

6. CONCLUSION AND FURTHER WORK

We study the problem of POI recommendation in this paper. We propose the temporal POI embedding model to capture the check-ins' sequential contexts and the various temporal characteristics on



(a) P@5 (b) R@5

Figure 8: Parameter effect on distance threshold s

different days. Moreover, we propose the geographically hierarchical pairwise ranking model to improve the recommendation performance through incorporating geographical influence. Finally, we propose the Geo-Teaser model as a uniform framework combining the two parts to recommend POIs. Experimental results on two datasets, Foursquare and Gowalla, show that our model outperforms state-of-the-art models. The proposed Geo-Teaser model improves at least 20% on both datasets for all metrics compared with SG-CWARP model.

Our future work may be carried out as follows: 1) Since we only consider the sequence of one day in this paper, we may discuss other scenarios in the future, for instance, sequences consisted of consecutive check-ins whose interval is under a fixed time threshold, e.g., four hours or eight hours. 2) We may subsume more information such as users' comments and social relations in this system to improve performance.

7. ACKNOWLEDGMENTS

We would like to thank the anonymous reviewers for their valuable comments and suggestions to improve the quality of the paper. The work described in this paper was partially supported by the Research Grants Council of the Hong Kong Special Administrative Region, China (No. CUHK 14203314 and No. CUHK 14205214 of the General Research Fund), and 2015 Microsoft Research Asia Collaborative Research Program (Project No. FY16-RES-THEME-005).

8. REFERENCES

[1] S. S. Bryan Perozzi, Rami Alrfou. Deepwalk (online learning of social representations). *SIGKDD*, 2014.
[2] C. Cheng, H. Yang, I. King, and M. R. Lyu. Fused matrix factorization with geographical and social influence in location-based social networks. In *AAAI*, 2012.
[3] C. Cheng, H. Yang, M. R. Lyu, and I. King. Where you like to go next: Successive point-of-interest recommendation. In *IJCAI*, 2013.
[4] E. Cho, S. A. Myers, and J. Leskovec. Friendship and mobility: User movement in location-based social networks. In *SIGKDD*, 2011.
[5] S. Feng, X. Li, Y. Zeng, G. Cong, Y. M. Chee, and Q. Yuan. Personalized ranking metric embedding for next new poi recommendation. In *IJCAI*, 2015.
[6] H. Gao, J. Tang, X. Hu, and H. Liu. Exploring temporal effects for location recommendation on location-based social networks. In *RecSys*, 2013.
[7] H. Gao, J. Tang, X. Hu, and H. Liu. Content-aware point of interest recommendation on location-based social networks. In *AAAI*, 2015.
[8] H. Gao, J. Tang, and H. Liu. gSCorr: Modeling geo-social correlations for new check-ins on location-based social networks. In *CIKM*, 2012.
[9] J. He, X. Li, L. Liao, D. Song, and W. K. Cheung. Inferring a personalized next point-of-interest recommendation model with latent behavior patterns. In *AAAI*, 2016.
[10] Y. Hu, Y. Koren, and C. Volinsky. Collaborative filtering for implicit feedback datasets. In *ICDM*, 2008.
[11] Q. V. Le and T. Mikolov. Distributed representations of sentences and documents. In *ICML*, 2014.
[12] H. Li, R. Hong, S. Zhu, and Y. Ge. Point-of-interest recommender systems: A separate-space perspective. In *ICDM*, pages 231–240. IEEE, 2015.
[13] X. Li, G. Cong, X.-L. Li, T.-A. N. Pham, and S. Krishnaswamy. Rank-geofm: A ranking based geographical factorization method for point of interest recommendation. In *SIGIR*, pages 433–442. ACM, 2015.
[14] D. Lian, C. Zhao, X. Xie, G. Sun, E. Chen, and Y. Rui. GeoMF: Joint geographical modeling and matrix factorization for point-of-interest recommendation. In *SIGKDD*, 2014.
[15] P. Liu, X. Qiu, and X. Huang. Learning context-sensitive word embeddings with neural tensor skip-gram model. In *IJCAI*, 2015.
[16] Q. Liu, S. Wu, L. Wang, and T. Tan. Predicting the next location: A recurrent model with spatial and temporal contexts. In *AAAI*, 2016.
[17] X. Liu, Y. Liu, K. Aberer, and C. Miao. Personalized point-of-interest recommendation by mining users' preference transition. In *CIKM*, 2013.
[18] X. Liu, Y. Liu, and X. Li. Exploring the context of locations for personalized location recommendations. In *IJCAI*, 2016.
[19] Y. Liu, Z. Liu, T.-S. Chua, and M. Sun. Topical word embeddings. In *AAAI*, 2015.
[20] Y. Liu, W. Wei, A. Sun, and C. Miao. Exploiting geographical neighborhood characteristics for location recommendation. In *CIKM*, 2014.
[21] T. Mikolov, Q. V. Le, and I. Sutskever. Exploiting similarities among languages for machine translation. *arXiv preprint arXiv:1309.4168*, 2013.
[22] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In *NIPS*, 2013.
[23] T. Mikolov, W.-t. Yih, and G. Zweig. Linguistic regularities in continuous space word representations. In *HLT-NAACL*, pages 746–751, 2013.
[24] R. Pan, Y. Zhou, B. Cao, N. N. Liu, R. Lukose, M. Scholz, and Q. Yang. One-class collaborative filtering. In *ICDM*, 2008.
[25] B. Recht, C. Re, S. Wright, and F. Niu. Hogwild: A lock-free approach to parallelizing stochastic gradient descent. In *NIPS*, 2011.
[26] S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme. BPR: Bayesian personalized ranking from implicit feedback. In *UAI*, 2009.
[27] D. Tang, B. Qin, and T. Liu. Learning semantic representations of users and products for document level sentiment classification. In *ACL*, 2015.

[28] D. Tang, B. Qin, T. Liu, and Y. Yang. User modeling with neural network for review rating prediction. In *IJCAI*, 2015.

[29] M. Xie, H. Yin, H. Wang, F. Xu, W. Chen, and S. Wang. Learning graph-based poi embedding for location-based recommendation. In *CIKM*, pages 15–24. ACM, 2016.

[30] J. Ye, Z. Zhu, and H. Cheng. What's your next move: User activity prediction in location-based social networks. In *SDM*, 2013.

[31] M. Ye, P. Yin, W.-C. Lee, and D.-L. Lee. Exploiting geographical influence for collaborative point-of-interest recommendation. In *SIGIR*, 2011.

[32] H. Yin, Y. Sun, B. Cui, Z. Hu, and L. Chen. Lcars: a location-content-aware recommender system. In *SIGKDD*, 2013.

[33] Q. Yuan, G. Cong, Z. Ma, A. Sun, and N. M. Thalmann. Time-aware point-of-interest recommendation. In *SIGIR*, 2013.

[34] J.-D. Zhang and C.-Y. Chow. Geosoca: Exploiting geographical, social and categorical correlations for point-of-interest recommendations. In *SIGIR*, pages 443–452. ACM, 2015.

[35] J.-D. Zhang, C.-Y. Chow, and Y. Li. LORE: Exploiting sequential influence for location recommendations. In *SIGSPATIAL*, 2014.

[36] J.-D. Zhang, C.-Y. Chow, and Y. Zheng. Orec: An opinion-based point-of-interest recommendation framework. In *CIKM*, 2015.

[37] S. Zhao, I. King, and M. R. Lyu. Capturing geographical influence in POI recommendations. In *ICONIP*, 2013.

[38] S. Zhao, M. R. Lyu, and I. King. Aggregated temporal tensor factorization for point-of-interest recommendation. In *ICONIP*. Springer, 2016.

[39] S. Zhao, T. Zhao, H. Yang, M. R. Lyu, and I. King. Stellar: spatial-temporal latent ranking for successive point-of-interest recommendation. In *AAAI*, 2016.