An Experimental Evaluation of Point-of-interest Recommendation in Location-based Social Networks

Yiding Liu[1] Tuan-Anh Nguyen Pham[2] Gao Cong[3] Quan Yuan[4]

[1,2,3]Nanyang Technological University
[4] University of Illinois at Urbana-Champaign
{[1]liuy0130@e.,[2]pham0070@e.,[3]gaocong@}ntu.edu.sg,[4]qyuan@illinois.edu

ABSTRACT

Point-of-interest (POI) recommendation is an important service to Location-Based Social Networks (LBSNs) that can benefit both users and businesses. In recent years, a number of POI recommender systems have been proposed, but there is still a lack of systematical comparison thereof. In this paper, we provide an all-around evaluation of 12 state-of-the-art POI recommendation models. From the evaluation, we obtain several important findings, based on which we can better understand and utilize POI recommendation models in various scenarios. We anticipate this work to provide readers with an overall picture of the cutting-edge research on POI recommendation.

1. INTRODUCTION

With the prominence of location-aware social media, people can easily share their content associated with locations. For example, Foursquare has more than 50 million active users and more than 8 billion check-ins to Points-of-Interests (POIs) had been made by 2016[1], and Yelp has around 21 million users and 102 million reviews on businesses with geographical coordinates[2].

With the availability of vast amount of users' visiting history, the problem of POI recommendations has been extensively studied. It has been found that 60%–80% of users' visits are in POIs that were not visited in the previous 30 days [34]. POI recommendations can greatly help users to find new POIs of their interests, which is beneficial to both users and businesses. However, compared with other recommendation problems (e.g., product, movie), POI recommendations face new challenges as follows:

- **Rich contexts**. First, user's mobility preference is affected by geographical distance: users usually visit POIs within a small number of activity regions (e.g., near home or work place). Second, users may visit same POIs everyday (e.g., home, work place). Third, users' preference is time-dependent. For example, a user is very likely to visit different places in early morning and late night. Fourth, users' visiting preferences might be affected by their social ties. Other types of context may include reviews on POIs, social posts on POIs, etc.
- **Data scarcity problem**. POI recommendations suffer from much worse data scarcity problem than other recommendation problems. The number of POIs visited by a user is usually only a small portion of all the POIs. For example, the density of the data used in experimental studies for POI recommendations is usually around 0.1%, while the density of Netflix data for movie recommendations is 1.2% [2].

POI recommendations received extensive research attention in the last five years, and many approaches have been proposed. Those studies differ in problem settings, recommendation models and evaluation data. The papers that present the newly proposed methods often report on experimental studies that suggest that the proposed methods perform better than some selected baselines on certain datasets. However, it is not clear whether they perform better on different types of data (e.g., more sparse) or different types of users (e.g., users with very few historical data). Worse still, these newly proposed methods are usually compared with other methods using similar framework (e.g., matrix factorization), and the new methods are not empirically compared with each other.

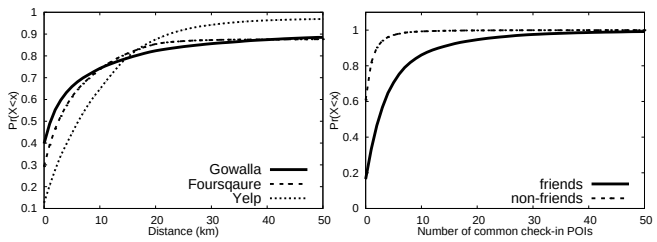
Furthermore, these proposed methods may make use of different types of context information, and adopt different frameworks to capture user preferences. There is a lack of empirical study on different methods of utilizing the same type of context information or capturing user preference for POI recommendations. This state of affairs makes it difficult to decide which method is the most suitable in a particular setting. Therefore, there is a clear need for a benchmark that offers in-depth insight into the performance of the existing POI recommendation methods.

To meet the need, we design an evaluation procedure to evaluate 12 representative POI recommendation models, including those recent proposals, aiming to gain a general picture of POI recommendation models from multiple aspects. Specifically, we experiment these models on datasets of different sources, and different sparsity, as well as users with different sizes of historical data. This evaluation offers new insight on the relative merit of these POI recommendation methods, and the applicable scenarios of these models. We also evaluate the different recommendation techniques for user preference modeling in POI recommendations, such as Matrix Factorization, and modeling methods for context information, such as geographical context. This evaluation will offer insights of which method performs better for each component, for designing more accurate POI recommendation methods in the future. This paper contributes the first all-around evaluation for 12 representative POI recommendation models.

The rest of this paper is organized as follows: Section 2 first gives an introduction of POI recommendations. Subsequently, re-

[1]https://foursquare.com/about
[2]http://www.yelp.com/about

Proceedings of the VLDB Endowment, Vol. 10, No. 10



(a) Spatial influence (b) Social influence

Figure 1: Spatial and social influence.

presentative models in our evaluation are categorized and presented in Section 3. Sections 4 and 5 present the experiments for POI recommendation models, from which some notable findings are uncovered. Finally, we review related work in Section 6.

2. POI RECOMMENDATION

Given a set of POIs $\mathcal{L}$, and a set of users $\mathcal{U}$ each associated with a set of POIs $\mathcal{L}^u$ visited by the user, the problem of POI recommendations is to recommend for each user $u \in \mathcal{U}$ new POIs, i.e., in the set of $\mathcal{L}/\mathcal{L}^u$, that are likely to be visited by user u. POI recommendations are significantly influenced by rich contexts such as geographical distance, social relations and time. To demonstrate their influence, we next show some statistical analysis results on the datasets from three LBSNs, i.e., Gowalla, Foursquare and Yelp. The details of these data are introduced in Section 4.1.

Observation 1: spatial influence. We consider user's consecutive check-ins as transitions between POIs, and compute the distribution of transition distances of users. Figure 1(a) shows the cumulative distribution functions (CDFs) of transition distances in Gowalla, Foursquare and Yelp. We can see that all the three curves rise dramatically when the distance is small. In Gowalla and Foursquare, 90% of users' transition distances are less than 50km. These indicate that users tend to visit nearby POIs.

Observation 2: social influence. We choose the Gowalla users whose check-ins are within Austin, Texas, and compute the numbers of common check-in POIs between friends and between randomly sampled non-friends. For friends and non-friends, the average numbers of common check-in POIs are 5.69 and 0.91, respectively. Particularly, Figure 1(b) shows the CDFs of the number of common check-in POIs. We can see that more than 60% non-friends have no common check-in POI, while the number is only 16% between friends. Moreover, around 85% friends have fewer than 10 common check-in POIs and over 80% non-friends have only 1 or no check-in POI in common. These indicate that most friends have small overlapping on their check-in POIs, but the overlapping is significantly larger than non-friends.

Observation 3: temporal influence. On the one hand, two users may behave differently with respect to time. For example, one often checks in restaurants during lunch time, while the other likes bars and often checks in at midnight. On the other hand, different POIs have different opening hours and peak hours (e.g. restaurants vs. bars), and thus their check-in patterns over time are also different.

3. MODELS FOR EVALUATION

In this section, we introduce 12 POI recommendation models included in the evaluation. They represent the state-of-the-art methods. They cover (i) four popular recommendation techniques and (ii) five types of context information such as geographical influence. These models are summarized in Table 1. Next, we group them based on their recommendation techniques, and introduce how they model and incorporate context information.

3.1 Matrix Factorization Models

Matrix Factorization (MF) [17] decomposes the check-in matrix $\mathbf{C} \in \mathbb{R}^{M \times N}$ into user matrix $\mathbf{U} \in \mathbb{R}^{M \times \mathcal{K}}$ and POI matrix $\mathbf{L} \in \mathbb{R}^{N \times \mathcal{K}}$, where M, N and $\mathcal{K}$ are the number of users, POIs and latent factors, respectively. Latent features of each user i and POI j are represented by $\mathbf{u}_i$ and $\mathbf{l}_j$. The recommendation score of user i for POI j is thus modeled as the inner product $\hat{C}_{ij} = \mathbf{u}_i \mathbf{l}_j^\top$, and the objective function is formulated as:

$$\min_{\mathbf{U},\mathbf{L}} ||\mathbf{C} - \mathbf{U}\mathbf{L}^\top||_F^2 + \lambda_1 ||\mathbf{U}||_F^2 + \lambda_2 ||\mathbf{L}||_F^2,$$

where $||\cdot||_F$ is the Frobenius norm of a matrix, λ_1 and λ_2 are regularization parameters. We next introduce the MF-based models considered in the evaluation.

3.1.1 LRT

LRT [11] is a time-enhanced MF model. Based on the observation that user's check-in behavior varies with time, LRT models each user by different latent vectors for different time slots, and the final recommendation score is computed from all the latent vectors.

Temporal influence. To model users' preferences at different time, LRT factorizes a check-in matrix $\mathbf{C}^{(t)}$ for each time slot t separately, where $t \in \{0, 1, ..., 23\}$ is an hour in a day. Furthermore, inspired by the intuition that users' interests in close time slots tend to be similar, a regularization term is added into the objective function of MF, formulated as $\sum_{t=1}^{T} \sum_{i=1}^{m} \psi_i(t, t-1) ||\mathbf{u}_i^{(t)} - \mathbf{u}_i^{(t-1)}||_2^2$, where $\psi_i(t, t-1)$ is the similarity between $\mathbf{C}_{i,}^{(t)}$ and $\mathbf{C}_{i,}^{(t-1)}$. LRT sums up the recommendation scores of all time slots as the final score, i.e., $\hat{C}_{ij} = \sum_t \mathbf{u}_i^{(t)} \mathbf{l}_j$.

3.1.2 IRenMF

IRenMF [30] is based on Weighted Matrix Factorization (WMF) [16, 35]. The intuitions behind IRenMF are (i) user has similar preferences on neighboring POIs (location-level influence) [15], and (ii) POIs in the same geographical region may share similar user preferences (region-level influence).

Geographical influence. For modeling location-level influence, the recommendation score of user i on POI j further includes the influence of the neighboring POIs $\mathcal{N}(l_j)$. Formally, the model defines the recommendation score $\hat{C}_{ij}$ as $\hat{C}_{ij} = \alpha \mathbf{u}_i \mathbf{l}_j^\top + (1 - \alpha) \frac{1}{Z(l_j)} \sum_{l_k \in \mathcal{N}(l_j)} Sim(l_j, l_k) \mathbf{u}_i \mathbf{l}_k^\top$, where $\alpha \in [0, 1]$ is used to control the influence of neighboring POIs, $Sim(l_j, l_k)$ is a distance-based weight between l_j and l_k, and $Z(l_j)$ is a normalization term.

For modeling region-level influence, IRenMF first clusters all POIs into G regions based on their geographical locations. By assuming the latent factors of POIs from the same region share the same sparsity pattern, they add a lasso penalty in the objective function as $\sum_{g=1}^{G} \sum_{k=1}^{\mathcal{K}} \omega_g ||\mathbf{L}_{(g)}^k||_2$, where $\mathbf{L}_{(g)}^k$ contains k^{th} latent factors of all POIs in region g and ω_g is a weight assigned to $\mathbf{L}_{(g)}^k$.

3.1.3 GeoMF

GeoMF [24] is a geographical WMF model. In order to capture the spatial clustering phenomenon (i.e., POIs visited by same users are likely to be in the same region [46]), GeoMF integrates geographical influence by modeling users' activity regions and the influence propagation on geographical space.

Geographical influence. GeoMF divides the whole geographical space into R grids, each of which represents a geographical region. For each POI, its influence is propagated to surrounding regions, attracting nearby users to visit. In particular, two matrices are introduced, namely user activity areas $\mathbf{X} \in \mathbb{R}^{M \times R}$ and POI influence

areas $\mathbf{Y} \in \mathbb{R}^{N \times R}$. The entry $Y_{jr} = \frac{1}{\sigma} K(\frac{d(r,j)}{\sigma})$ denotes POI j's influence on region r, where $d(r, j)$ is the distance between POI j and region r, $K(\cdot)$ is standard normal distribution and σ is the standard deviation. X_{ir} denotes the possibility of user i appearing in region r. User i's geographical preference on POI j is estimated by $\mathbf{x}_i \mathbf{y}_j^\top$, and the final recommendation score is $\hat{C}_{ij} = \mathbf{u}_i \mathbf{l}_j^\top + \mathbf{x}_i \mathbf{y}_j^\top$.

3.1.4 RankGeoFM

RankGeoFM [22] is an ranking-based MF model that (i) learns users' preference rankings for POIs, and (ii) includes the geographical influence of neighboring POIs.

Geographical influence. RankGeoFM uses another latent matrix $\mathbf{U}^{(2)}$ to represent users' geographical preferences, in addition to user preference matrix $\mathbf{U}^{(1)}$. The recommendation score thus is computed as $\hat{C}_{ij} = \mathbf{u}_i^{(1)} \mathbf{l}_j^\top + \mathbf{u}_i^{(2)} \cdot \sum_{l_k \in \mathcal{N}(l_j)} w_{jk} \mathbf{l}_k^\top$. The first term models the user preference score, while the second term models the geographical influence score that a user likes a POI because of its neighbors, where $\mathcal{N}(l_j)$ refers to the neighboring POIs of j and w_{jk} is the distance-based weight assigned to POI k.

3.1.5 ASMF

ASMF [20] is a two-step POI recommendation framework that (i) learns potential locations from users' friends and (ii) incorporates potential locations into WMF to overcome cold-start problem.

Social influence. For each user i, ASMF considers the locations that are visited by three types of friends, i.e., social friends, location friends and neighboring friends, as his/her potential locations pot_i, and assigns a small value $\alpha \in [0, 1]$ to C_{ik}, where $k \in pot_i$.

Categorical influence. ASMF uses a category-based weight when computing the recommendation score, i.e., $\hat{C}_{ij} = (Q_{ic_j} + \epsilon) \mathbf{u}_i \mathbf{l}_j^\top$, where c_j is the category of POI j, Q_{ic_j} is the preference of user i on c_j and ϵ is a tuning parameter.

Geographical influence. A distance-based geographical score p_{ij}^G is fused with the result of WMF as the overall recommendation score, i.e., $\hat{C}_{ij} \propto p_{ij}^G \times \hat{C}_{ij}$, where p_{ij}^G is computed based on the distance distribution between users' home and their check-in POIs.

3.2 Poisson Factor Models

Poisson Factor Model (PFM) [31] is a probabilistic model that factorizes the user-POI check-in matrix $\mathbf{C}$ as $\mathbf{C} \sim Poisson(\mathbf{U}\mathbf{L}^\top)$. We include two PFM-based models in our evaluation.

3.2.1 MGMPFM

MGMPFM [6] is a fusion model combining the outputs of PFM and a geographical modeling method, namely Multi-center Gaussian Model (MGM).

Geographical influence. Based on the observation that user's check-ins are usually distributed around several centers, such as home and workplace, MGM learns regions of activity for each user using multiple Gaussian distributions.

Recommendation. The recommendation score is defined as $P_{ij} = P(C_{ij}) \cdot P(l_j|R_i)$, where $P(C_{ij}) \propto \mathbf{u}_i \mathbf{l}_j^\top$ is the output of PFM, $P(l_j|R_i)$ is computed by MGM, and R_i is the regions of user i.

3.2.2 GeoPFM

The idea of GeoPFM [26] is that user's geographical preference and interest preference are mutually affected, and user's preference is related to both of them. Hence, GeoPFM jointly learns both geographical preference and interest preference for users.

Geographical influence. Latent regions are integrated into PFM, represented by two-dimensional Gaussian distributions on the spatial space, and each user has a multinomial distribution on regions.

3.3 Link-based Models

LFBCA [41] is a link-based model that constructs a graph to model LBSN users and their relations. In the graph, both **user preference** and **social influence** are modeled by different types of edges. Particularly, users with similar check-in behaviors are linked to model the "similarity relations" and edges representing "friendship relations" are added to connect friends in the graph. Based on the constructed graph, the Bookmark-Coloring Algorithm algorithm (BCA) [3] is executed for each user to compute his/her similarity to every other users, and then the User-based Collaborative Filtering (UCF) [1] is performed based on the similarities.

3.4 Hybrid Models

Hybrid model combines the outputs of two or more recommendation methods and each method models user preference or a type of context information. For example, MGMPFM (see Section 3.2.1) is a hybrid model of PFM and MGM. We introduce more hybrid models included in our evaluation next. Note that for recommendations, USG linearly combines the context influence, while other models use the multiplication of context influence as the final recommendation scores.

3.4.1 USG

USG [46] models user preference, social influence and geographical influence simultaneously for POI recommendations.

User preference. The model adopts UCF to model user preference, where the similarity between two users is computed based on their common check-ins.

Social influence. To exploit the social influence, USG proposes Friend-based Collaborative Filtering (FCF) to make POI recommendations based on similar friends. The similarity between friends is based on their common check-in POIs and common friends.

Geographical influence. Given a POI j and a user i, geographical influence is estimated as the probability of visiting j based on the user's historical visited POIs $\mathcal{L}_i$, i.e., $P_{ij}^g = \prod_{l_k \in \mathcal{L}_i} Pr(d(l_j, l_k))$, where $d(l_j, l_k)$ is the distance between l_j and l_k. $Pr(\cdot)$ estimates the probability that users travel a distance of $d(l_j, l_k)$.

3.4.2 iGSLR

iGSLR [54] exploits geographical preference and social influence for POI recommendations.

Social influence. Similar to USG, iGSLR also uses FCF to leverage friends' check-ins, where the similarity between friends is computed based on the distance of their residences. In our datasets, since residence locations of users are not available, we take users' most frequent check-in POIs as their residences.

Geographical influence. For each user, iGSLR learns a distance distribution from his/her check-in history using Kernel Density Estimation (KDE). The probability of user i visiting a new POI j is thus estimated based on the KDE values of the distances between POI j and the POIs visited by user i.

3.4.3 LORE

Different from other models, LORE [57] considers sequential influence, in addition to social and geographical influence.

Social influence. FCF is adopted to model social influence, where social similarities are defined as in iGSLR.

Geographical influence. For each user, LORE models a check-in probability distribution over a two-dimensional space using KDE. The geographical probability of visiting a new POI is then estimated based on its location on the check-in probability distribution.

Sequential influence. LORE employs additive Markov chain (AMC) [40] to exploit sequential influence between POIs. The sequential probability of a user visiting a POI is based on the transition probability between all the user's visited POIs and the target POI.

3.4.4 GeoSoCa

GeoSoCa [55] models three types of context information, namely geographical, social and categorical correlations.

Geographical influence. GeoSoCa also uses two-dimensional KDE for geographical modeling. Different from LORE, where σ is shared by all users, GeoSoCa adds a local (i.e., user-dependent) bandwidth to make the geographical modeling more personalized.

Social influence. GeoSoCa estimates a power-law distribution (denoted as $f_S(x_{ij})$) of users' social check-in frequency. The social check-in frequency x_{ij} refers to the check-in frequency on POI j made by user i's friends. GeoSoCa uses the cumulative distribution of $f_S(x_{ij})$ as the social influence in recommendations.

Categorical influence. Similar to social influence modeling, GeoSoCa estimates a power-law distribution (denoted as $f_C(y_{ic})$) for users' categorical check-in frequency. The categorical check-in frequency y_{ic} denotes the check-in frequency of user i on all the POIs with category c. The cumulative distribution of $f_C(y_{ic})$ is used as categorical influence in recommendations.

4. EVALUATION SETTING

4.1 Datasets

Our experiments are conducted on three public datasets.

Gowalla dataset. The Gowalla check-in data[3] was generated worldwide from February 2009 to October 2010. We filter out those users with fewer than 15 check-in POIs and those POIs with fewer than 10 visitors. The filtered dataset comprises 18,737 users, 32,510 POIs, 1,278,274 check-ins. The sparsity of user-POI check-in matrix is 99.865%.

Foursquare dataset. The Foursquare data[4] [44] includes check-in data from April 2012 to September 2013. We use the records generated within United States (except Alaska and Hawaii) and eliminate those users with fewer than 10 check-in POIs, as well as those POIs with fewer than 10 visitors. The filtered dataset contains 24,941 users, 28,593 POIs and 1,196,248 check-ins. The sparsity of user-POI check-in matrix is 99.900%.

Yelp dataset. The Yelp data[5] contains a large number of geo-tagged businesses (considered as POIs) and reviews within several cities. We eliminate those users with fewer than 10 check-in POIs, as well as those POIs with fewer than 10 visitors. This yields a dataset with 30,887 users, 18,995 POIs and 860,888 reviews. The sparsity of user-POI check-in matrix is 99.860%.

We partition each dataset into training set, tuning set and test set. For each user, we use the earliest 70% check-ins as the training data, the most recent 20% check-ins as the test data and the remaining 10% as the tuning data. For each model, we tune the parameters based on the tuning data to find the optimal values that maximize Pre@10 and Rec@10 (see Section 4.2), and subsequently use them in the test data. The parameter settings are provided in Appendix B of the full version [29].

4.2 Evaluation Metrics

To evaluate the models, we use 4 widely-used metrics, i.e., precision (Pre@K), recall (Rec@K), normalized discounted cumulative gain (nDCG@K) and mean average precision (MAP@K). To our knowledge, none of the previous work uses all the 4 metrics for experiments. The formal definition of the metrics is included in Appendix A of the full version [29].

4.3 Performance Evaluation Procedure

To systematically evaluate all the models, we devise an all-around evaluation procedure with the following four components.

4.3.1 Evaluation on Different Types of Data

To evaluate the effect of different data properties on accuracy, we design the following two experiments.

Different datasets. We evaluate all the models on Gowalla, Foursquare and Yelp data. We vary K from 5, 10, 20 to 50.

Data density. To investigate the effect of training data density, for each of Foursquare and Gowalla datasets, we generate training sets with different density levels, i.e. 0.0010, 0.0008, 0.0006, 0.0004 and 0.0002, by randomly eliminating non-zero entries of the check-in matrix (to make it sparser) or randomly moving data from tuning set to training set (to make it denser). Note that the density of Foursquare dataset is 0.0010, and thus we can only generate training data with density less than 0.0010.

4.3.2 Evaluation for Different Types of Users

We design three experiments to study the effect of different types of user properties on recommendation models.

Number of check-in POIs of users. The number of check-in POIs is expected to affect the accuracy of recommendation models. We divide users into groups based on the number of check-in POIs in training data. Particularly, we divide Gowalla users into five groups: "<15", "15–30", "30–50", "50–100" and ">100", which contain 6164, 7201, 2979, 1672 and 721 users, respectively. We also divide Foursquare users into five groups: "<10", "10–20", "20–30", "30–50" and ">50", which contain 6045, 9689, 4882, 3341 and 984 users, respectively. We train the models using all users and evaluate them on different groups of users separately.

Activity range of users. As geographical factor is important in POI recommendations, we design this experiment to study the effect of users' activity range on recommendation models. In particular, we divide users of both Gowalla and Foursquare data into five groups: "<10", "10–50", "50–200", "200–800" and ">800" based on the average distance between their check-in POIs (in kilometers), which reflects the activity range. In Gowalla, there are 3664, 3263, 3536, 4213 and 4061 users in the five groups, respectively. In Foursquare, there are 6814, 3189, 3419, 5939 and 5580 users in the five groups, respectively. Note that the numbers of check-in POIs in these groups are similar, and thus will not affect the results.

4.3.3 Evaluation for Different Modeling Methods

As discussed in Section 3.4 , these hybrid POI recommendation models differ in how they model user preferences and each type of context information, particularly the geographical and social influence. It is very useful if we can evaluate the individual component of these methods to answer questions such as how good is the user preference modeling component for POI recommendations. We design the following experiments to compare the user preference, geographical and social components used in the models. The names of these individual modeling components are shown in Table 1.

Comparing geographical modeling methods. We evaluate 6 types of geographical modeling methods used in these models. Moreover, we also evaluate the accuracy for users with different numbers of check-in POIs. Note that it is difficult to isolate the geographical

[3] http://snap.stanford.edu/data/loc-gowalla.html

[4] https://sites.google.com/site/yangdingqi/home/foursquare-dataset

[5] Yelp dataset challenge round 7 (access date: Feb 2016), https://www.yelp.com/dataset_challenge

Table 1: Summary of the POI recommendation models in our evaluation.

Models	Methodology					Information-used					
	CF	MF	PFM	Link	Hybrid	User Pref.	Geographical	Social	Temporal	Sequential	Categorical
USG [46]	✓				✓	✓(UCF)	✓(PD)	✓(CI-/CN-FCF)			
MGMPFM [6]			✓		✓	✓(PFM)	✓(MGM)				
LRT [11]		✓				✓(MF)			✓		
iGSLR [54]	✓				✓		✓(1dKDE)	✓(D-FCF)			
LFBCA [41]				✓		✓(LBCA)		✓(FBCA)			
LORE [57]	✓				✓		✓(2dKDE)	✓(D-FCF)		✓	
IRenMF [30]		✓				✓(WMF)	✓				
GeoMF [24]		✓				✓(WMF)	✓				
RankGeoFM [22]		✓				✓(BPRMF)	✓				
GeoPFM [26]			✓			✓(PFM)	✓				
GeoSoCa [55]					✓		✓(AKDE)	✓(SC)			✓
ASMF [20]		✓				✓(WMF)	✓	✓			✓

and social parts from those MF-based models and GeoPFM, and thus we do not include them here.

Comparing social modeling methods. We evaluate 5 types of social modeling methods used in these models. We also evaluate the accuracy for users with different numbers of friends.

Comparing user preference modeling methods. We evaluate 7 types of user preference modeling methods, without any context information (e.g., geographical and social information) included.

4.3.4 Scalability Evaluation

Scalability is also a important dimension for the practical interest of a recommender system. Therefore, we evaluate both the training and querying (i.e., recommendation) scalability of the models. To the best of our knowledge, this is the first work to evaluate both training and querying scalability of POI recommendation models.

Time complexity analysis. We analyze the training and querying time complexity of different models and the results are summarized in Appendix F of the full version [29].

Training scalability. To explore the training scalability, we use 20%, 40%, 60% and 80% of the Gowalla dataset as the training sets to test the training time of 7 models (i.e., MGMPFM, LRT, LFBCA, IRenMF, GeoMF, RankGeoFM and GeoPFM).

Querying scalability. To explore the querying scalability, we divide Gowalla users into five groups: "<15", "15–30", "30–50", "50–100" and ">100" based on the number of users' check-in POIs, and test the average querying time of the models for each group.

5. EVALUATION RESULTS

In this section, we show the experimental results[6]. Note that for some experiments, results on nDCG and MAP are similar with precision and recall, and we do not show them for saving space.

5.1 Performance on Different Types of Data

5.1.1 Performance on Different Datasets

Figures 2, 3 and 4 depict the overall comparison of the 12 models with respect to top-K recommendations on Gowalla, Foursquare and Yelp, respectively. Note that the Foursquare data does not have social information and thus we only report results for those methods without utilizing social information. Additionally, GeoSoCa and ASMF are only evaluated on Yelp since Gowalla and Foursquare do not have the categorical information it needs.

For each model, the accuracy are similar on Gowalla and Foursquare data. However, all the models perform worse on Yelp. For example, the Pre@5 of RankGeoFM on Gowalla and Foursquare are 0.069 and 0.063, respectively, while the value is only 0.032 on Yelp. This might be because that the activity range of the Yelp users are larger than the Gowalla and Foursquare users (see Figure 1(a)), which makes it difficult to model users' geographical preferences (which is discussed in Section 5.2.2). Considering the relative performances, most of the models perform consistently in the three datasets. We discuss their relative performances as follows.

Hybrid models. Among hybrid models, USG exhibits better recommendation quality than iGSLR, LORE and GeoSoCa. Take the results on Yelp as an example, in terms of Pre@5, USG outperforms the other three by 105.93%, 6.71% and 43.29%, respectively. One possible reason is that although iGSLR, LORE and GeoSoCa leverage geographical, social, sequential and categorical information to indirectly characterize user preference, they still miss direct modeling of user preference as USG does (i.e., using UCF). Therefore, for hybrid models, it would be better to utilize other information on top of user preference modeling for POI recommendations.

IRenMF, GeoMF and RankGeoFM. They are the top-3 best models, and RankGeoFM normally performs the best. Compared with USG, RankGeoFM and IRenMF achieve approximate 14% and 10% improvement, respectively, on Gowalla in terms of all the 4 metrics and different K values. GeoMF also outperforms USG by 5%-10% and 15%-20% on Gowalla and Yelp, respectively. All the three models are designed for implicit feedback data. This indicates that modeling user's check-ins as implicit feedback is more appropriate in POI recommendations.

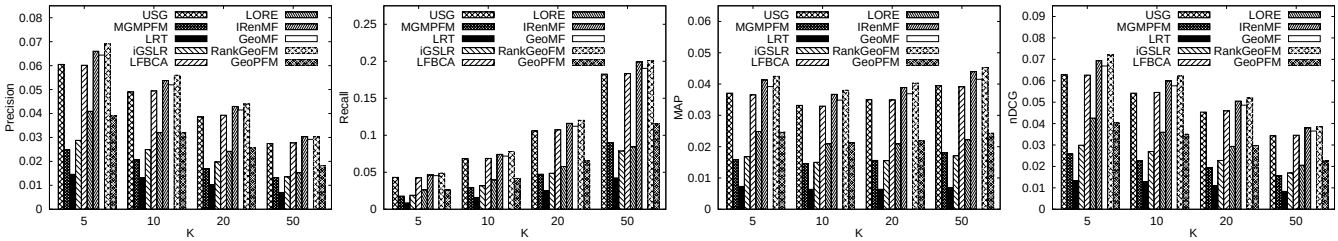
Comparing IRenMF and GeoMF. IRenMF is better than GeoMF on Gowalla and Foursquare data, but worse on Yelp data. This might be because that IRenMF assumes users tend to visit those POIs near their visited locations, while GeoMF considers those POIs as negative samples. Figure 1(a) shows that the distances between users' check-in POIs in Gowalla and Foursquare are smaller than Yelp, which means users are more likely to visit nearby POIs in Gowalla and Foursqaure. Thus, the assumption of IRenMF is more likely to hold on Gowalla and Foursquare than Yelp, which leads to their different performances on these datasets.

ASMF. ASMF performs similarly to USG (<5% difference). However, it is not as good as the other geographical-enhanced MF models. For example, GeoMF is 8%–23% better than ASMF on all the evaluation metrics and K values. This might be because ASMF focuses on utilizing social information, while learning geographical influence might be a better way to improve MF model.

LRT. LRT gives the worst performance among these models. It only considers temporal information, without modeling geographical and social influence. In addition, partitioning the check-in matrix based on time slots makes the data sparser, causing negative effects on learning user preferences.

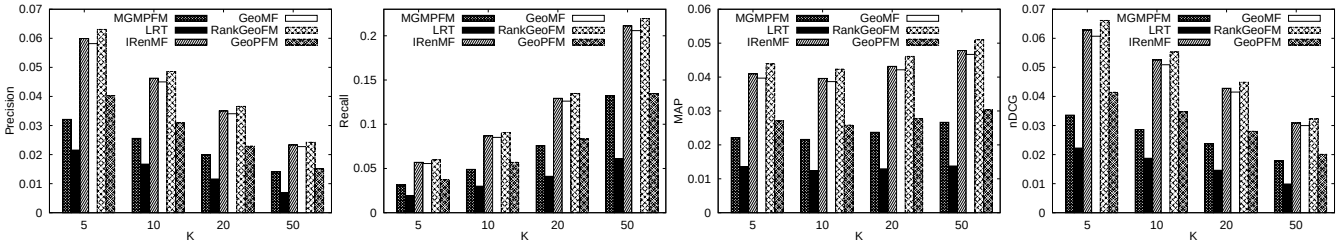
LFBCA. Ye et al. [46] show that the performance of a link-based method is not as good as USG. However, in our evaluation, LFB-

[6]The related datasets and source code are available at: http://spatialkeyword.sce.ntu.edu.sg/eval-vldb17/



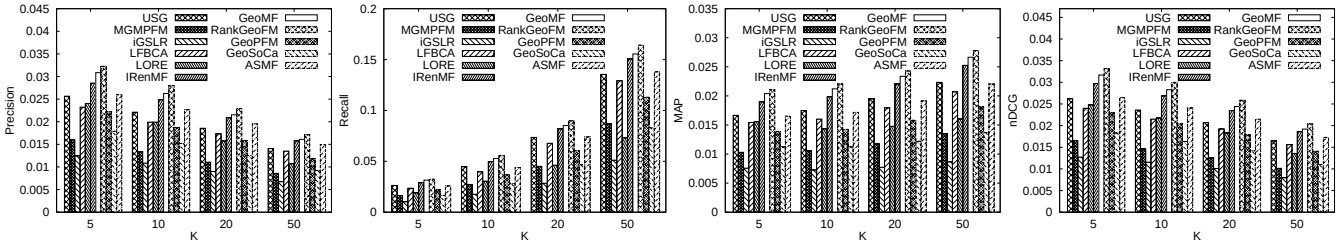
(a) Precision (b) Recall (c) MAP (d) nDCG

Figure 2: Varying K on Gowalla.



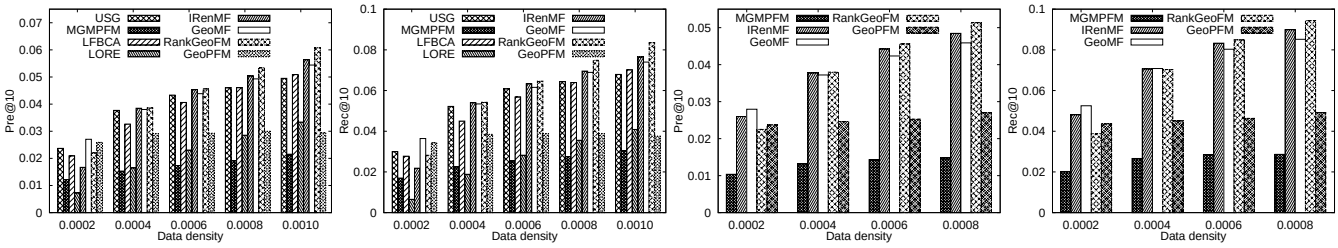
(a) Precision (b) Recall (c) MAP (d) nDCG

Figure 3: Varying K on Foursquare.



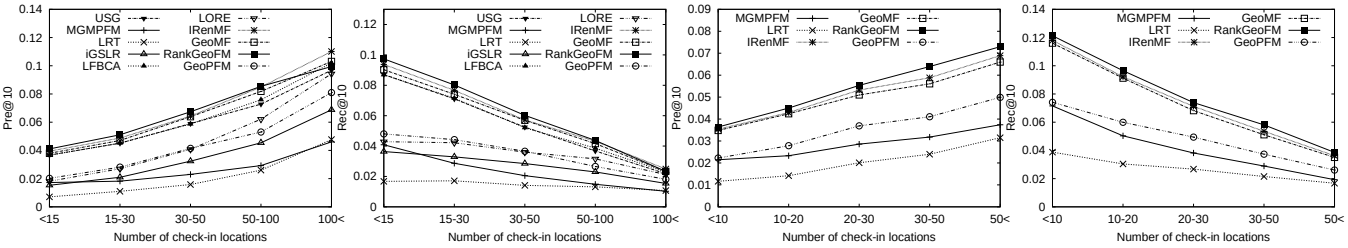
(a) Precision (b) Recall (c) MAP (d) nDCG

Figure 4: Varying K on Yelp.



(a) Pre@10 - Gowalla (b) Rec@10 - Gowalla (c) Pre@10 - Foursquare (d) Rec@10 - Foursquare

Figure 5: Varying data sparsity.

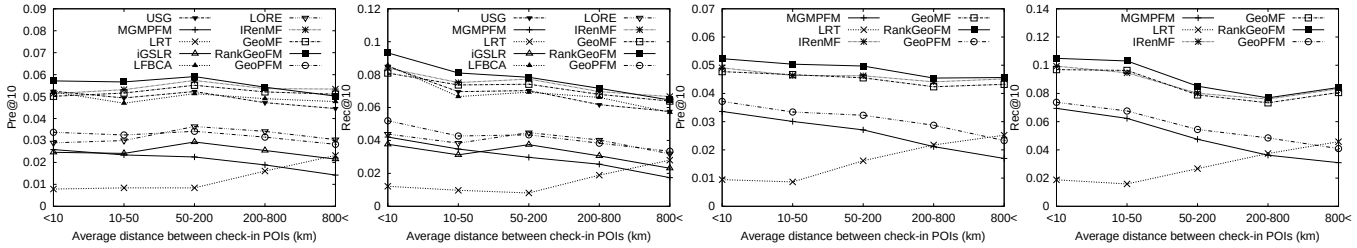


(a) Pre@10 - Gowalla (b) Rec@10 - Gowalla (c) Pre@10 - Foursquare (d) Rec@10 - Foursquare

Figure 6: Performance of POI recommendations for users with different numbers of check-in POIs

CA, which is also a link-based method, achieves similar accuracy with USG on Gowalla, and it is 5%–10% worse than USG on Yelp. This is because LFBCA not only utilizes information of friends, but also considers the effects of other spatially similar users. This tells us that, for link-based models, users' relations with respect to their spatial behaviors should also be considered as "links".

Poisson Factor Models. In the experiments, GeoPFM consistently beats MGMPFM. The difference between them is that GeoPFM jointly learns geographical influence and user preference, while MGMPFM simply fuses the outputs of the two components. This implies that joint learning would be a better approach for leveraging context information than separately modeling.



(a) Pre@10 - Gowalla (b) Rec@10 - Gowalla (c) Pre@10 - Foursquare (d) Rec@10 - Foursquare

Figure 7: Performance of POI recommendations for users with different average distance among check-in POIs.

5.1.2 Performance on Different Density

Figure 5 shows the accuracy on Gowalla and Foursquare, under different data densities. LRT and iGSLR are not included here since their performances are not good in the previous experiments. The results on Yelp are qualitatively similar and omitted.

GeoPFM. In general, except for GeoPFM, every model is significantly jeopardized by lower data density. For example, the performances of these models in terms of Pre@10 decrease at least 35% on Gowalla, when the data density drops from 0.0008 to 0.0002. Although GeoPFM is not as good as RankGeoFM, IRenMF and GeoMF when data is dense, it is very robust to low data density. In terms of Pre@10, GeoPFM only experiences 13.66% and 12.25% loss when the density decreases from 0.0008 to 0.0002. One possible explanation is that GeoPFM uses a hierarchical way to profile user preferences, i.e., each user has preferences on latent regions and preferences on the POIs within each region. This might be helpful in overcoming data scarcity problem.

RankGeoFM. RankGeoFM reports the best performance when the density is greater than 0.0008, outplaying the second best one (i.e., IRenMF) by 5%-10%. However, when the density downs to 0.0002, its accuracy declines dramatically, which is worse than GeoMF and GeoPFM. It might be because that RankGeoFM learns to rank the positive examples higher than negative examples, but there are fewer positive examples available to learn the rankings in sparse data. As a consequence, the result is less reliable.

GeoMF. The performance of GeoMF is slightly worse than IRenMF and RankGeoFM at 0.0008 and 0.0010, but turns to be the most effective model when the density is 0.0002, outperforming the second best one by 5%–10% on both datasets. Hence, GeoMF is preferable than the other models for sparse data.

Comparing USG and LFBCA. The results show that USG performs better than LFBCA on sparse data (0.0002 – 0.0006) by at least 7%, while slightly worse on dense data (0.0010) by 2%–3%. This might be because that for user preference modeling, LFBCA also considers indirect similar users, e.g., similar users of the similar users. The indirect users might provide more information when data is dense, while cause bias when data is sparse.

LORE. LORE does not directly model user preference, but it still benefits from higher density. This indicates that context information such as social influence, is also sensitive to data density.

5.1.3 Performance on Other Datasets

In addition, we also conduct extensive experiments on some other datasets. The results on these datasets are qualitatively similar and are provided in Appendix C and D of the full version [29].

5.2 Performance on Different Users

5.2.1 Users with Different Numbers of Check-in POIs

Figure 6 shows the results for different groups of users with different numbers of check-in POIs. The models considering social information are only evaluated on Gowalla, and results on Yelp are similar and are omitted. We make the following observations: (1) USG, LFBCA, IRenMF, GeoMF and RankGeoFM consistently outperform the other models in general, among which IRenMF, GeoMF and RankGeoFM are usually better than the other two by 5%–15% on both datasets. The improvement is even larger for the groups of users with fewer check-in POIs. This observation confirms the superiority of the 5 models for both active and cold-start users. (2) LORE experiences the largest increase as the number of check-in POIs goes up as shown in Figure 6(a). In particular, for the group of users "<15", the Pre@10 of LORE is only half of USG, while it becomes comparable to the Pre@10 of USG for the group of users "100<". It is worth noting that, the increase of other models using geographical, social and temporal information is not as significant as LORE. Thus, LORE's improvement on active users could largely be attributed to its sequential influence modeling.

5.2.2 Users with Different Activity Ranges

Figure 7 shows the results for groups of users with different activity ranges on Gowalla and Foursquare. The results on Yelp are qualitatively similar and are omitted. We make two observations: (1) RankGeoFM, GeoMF, IRenMF, USG and LFBCA outperform the others by a large margin (>30%) on all bases, and RankGeoFM usually performs the best. (2) All the models with geographical modeling experience at least 10% loss from the first group (<10) to the last group (800<) on both datasets in terms of Rec@10. This means larger range of user's activity causes side effects for geographical-enhanced POI recommendations. Geographical modelings may not work well for users with wide activity ranges.

5.2.3 Tourist Users

To evaluate the models for a special type of users, namely tourists, we compare the models with a simple baseline on two additional datasets. We find that the baseline of recommending the most popular POIs performs better than the most of the models we evaluated and comparable to RankGeoFM when recommending for tourists. The details can be found in Appendix E of the full version [29].

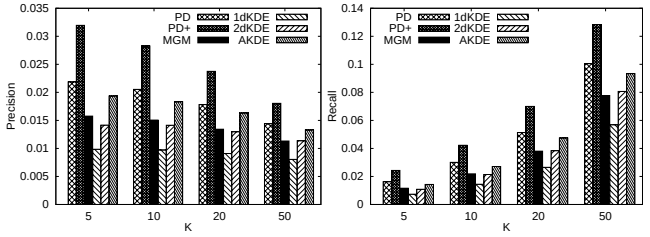
5.3 Different Modeling Methods

5.3.1 Geographical Modeling

Figures 8 and 9 show the evaluation results of 6 geographical modeling methods on Gowalla data. Similar results can be found on Foursquare and Yelp, and thus are omitted.

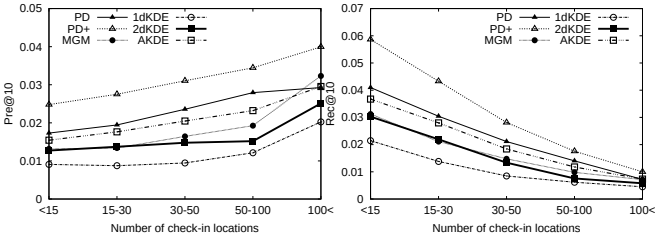
PD & PD+. PD is the geographical component of USG (see Section 3.4.1) and PD+ is an improved version of PD [50]. As shown in Figures 8 and 9, PD+ outperforms all the other models by at least 25% with respect to different K values and user groups, and PD is the second best method in most cases.

MGM. MGM is the geographical component of MGMPFM (see Section 3.2.1). Figure 8 shows that the overall performance of MGM is not satisfactory, while Figure 9 shows that it performs well for users with more than 100 check-in POIs, better than PD by 10% on Pre@10 and only 1% worse on Rec@10. It indicates that when a user has many check-in POIs, the distribution of his/her POIs is



(a) Pre@K - Gowalla (b) Rec@K - Gowalla

Figure 8: Performance of geographical modeling methods.



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

Figure 9: Performance of geographical modeling methods for users with different numbers of check-in POIs.

more likely to be consistent with the assumption of MGM, i.e., following multi-centered Gaussian distribution.

1dKDE. 1dKDE is the geographical component of iGSLR (see Section 3.4.2). It performs the worst in our experiments.

2dKDE & AKDE. 2dKDE and AKDE are the geographical component of LORE and GeoSoCa (see Section 3.4.3 and Section 3.4.4), respectively. AKDE consistently outperforms 2dKDE by 15%–40%. This is because AKDE uses both global and personalized bandwidth of the kernel function for each user, and thus can perform better. Compared to PD, AKDE is 8%–17% worse for the first four user groups (<100), while becoming similar (<2% difference) for the last user group (>100). This means that, similar to MGM, AKDE is also better for active users.

5.3.2 Social Modeling

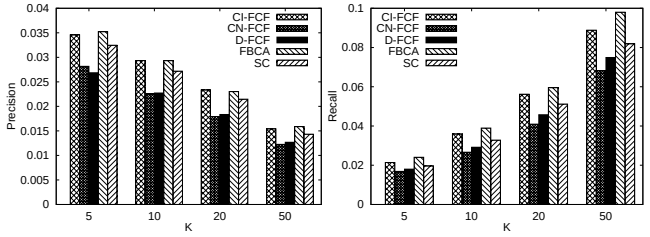
Figures 10 and 11 show the results of 5 social modeling methods:

FBCA. FBCA is the social component of LFBCA (see Section 3.3). In general, FBCA is the best social modeling method, outperforming the second best one, i.e., CI-FCF, by 6%–12% in terms of recall on Gowalla, while their precision values are similar. This indicates that link-based method might be a good choice for modeling social influence. Specifically, Figure 11 shows that FBCA is more powerful for cold-start users but not effective for active users. This is because for users with few friends, FBCA can utilize his/her friends' friends and so on, which is helpful for aggregating more information for cold-start users, while the other models only consider the direct friends of users. However, for active users, FBCA might be inaccurate due to including the effects of too many indirect friends.

CI-FCF & CN-FCF. CI-FCF and CN-FCF are two social components of USG (see Section 3.4.1). From Figure 10, we can see that CI-FCF is more effective than CN-FCF by 22.77% in terms of Pre@5. It means that spatial similarity, which is based on check-in POI and frequency, is better than online similarity such as the number of common friends, for FCF method. This result is consistent with the previous work [46].

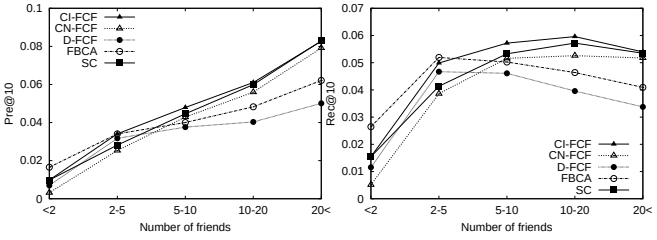
D-FCF. D-FCF is the social component of iGSLR and LORE (see Sections 3.4.2 and 3.4.3). Figure 10 shows that it is the worst among all methods.

SC. SC is the social component of GeoSoCa (see Section 3.4.4). In terms of Pre@10, SC outperforms CN-FCF and D-FCF by 15.23% and 20.87%, while it is outperformed by CI-FCF, which also aggregates social check-ins for recommendations. This means CI-FCF might be a better choice for social check-in aggregation.



(a) Pre@K - Gowalla (b) Rec@K - Gowalla

Figure 10: Performance of social modeling methods.



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

Figure 11: Performance of social modeling methods for users with different numbers of friends.

It is worth noting that, in Figure 11, the recall values of all models increase in the beginning, but slightly decrease afterwards. This is because users with more friends also tend to check-in at more POIs. Specifically, users in the first group (<2) has 21.57 check-in POIs on average, while this value of the last group (20<) is 38.80.

5.3.3 User Preference Modeling

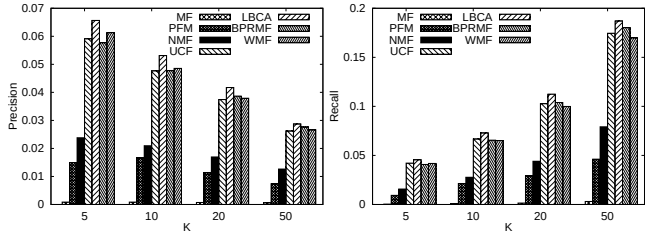
We compare 7 user preference modeling methods employed in POI recommendation models, including User-based Collaborative Filtering (UCF), Location-based Bookmark-coloring Algorithm (LBCA) Matrix Factorization (MF), Poisson Factor Model (PFM), Weighted Matrix Factorization (WMF), Non-negative Matrix Factorization (NMF) [25] and Bayesian Personalized Ranking (BPRMF) [36]. To focus on user preference modeling only, all of these methods are based on user-POI check-in matrix, without utilizing any context information. Figures 12 and 13 depict the accuracy of these methods.

LBCA. LBCA is the link-based model in Section 3.3 without social links in the user-user graph. LBCA is the best method for user preference modeling. For example, it outperforms UCF, WMF and BPRMF by 11%, 7% and 13%, respectively, in terms of Pre@5. One possible reason is that it utilizes both direct and indirect similar users (i.e., similar users of similar users), and thus can aggregate more information for recommendations.

MF. Matrix Factorization, which performs well in traditional recommendation problems, fails for POI recommendations. This is because the data in POI recommendations is much sparser than in traditional recommendation problems, and MF does not fit well with check-ins, which is implicit feedback data. The finding is consistent with the result reported in previous work [22].

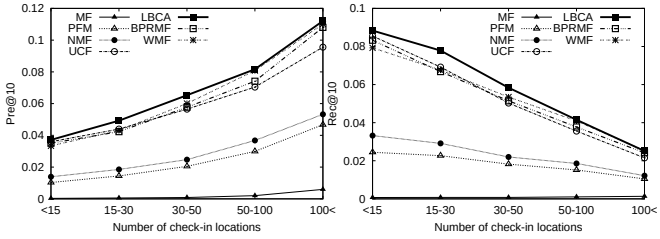
PFM & NMF. They do not perform well for POI recommendations. This also explains why MGMPFM and GeoPFM, which are based on PFM, are not good in our previous experiments.

UCF, WMF & BPRMF. They perform similarly (<3% difference) and are second to the best methods for modeling user preference. For example, they outperform NMF and PFM by over 140% on Pre@5. Moreover, they are better for both cold-start users and active users. This also explains why the models based on these methods, e.g., USG, IRenMF, GeoMF and RankGeoFM, perform good.



(a) Pre@K - Gowalla (b) Rec@K - Gowalla

Figure 12: Performance of user preference modeling methods.



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

Figure 13: Performance of user preference modeling methods for users with different numbers of check-in POIs.

5.4 Scalability

5.4.1 Training Scalability

Table 2 shows the training time of the POI recommendation models on training sets of different sizes. Note that we do not evaluate the training scalability of USG, iGSLR, LORE and GeoSoCa, since they do not need training. For ASMF, the training time on the Gowalla data (without category information) is the same as training WMF and thus is omitted. For LFBCA, we report the training time of the Bookmark-Coloring Algorithm (BCA). To compare the scalability of each model with different training data sizes, we show the relative training time in Figure 14(a), which is defined as the ratio comparing to the running time using 20% training data.

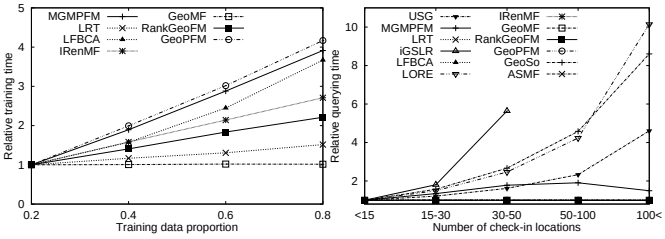
We observe that (1) GeoMF is almost not affected by the size of the training set, with a variation of less than 2%. This is because the training time of GeoMF is based on $||\mathbf{C} - \mathbf{X}\mathbf{Y}^\top||_0$ instead of N_C (Table 7, Appendix F [29]). The value of $||\mathbf{C} - \mathbf{X}\mathbf{Y}^\top||_0$ is not influenced by the size of training data. Therefore, GeoMF is more scalable than the other models. (2) MGMPFM, LRT, IRenMF, RankGeoFM and GeoPFM scale linearly with the size of training data. Among these models, LRT is more scalable than the others. When the size of the training data increases by 300% (from 20% to 80%), the training time of LRT only increases by 50%, while IRenMF and RankGeoFM increase by over 170% and 120%, respectively. MGMPFM and GeoPFM perform similarly and worse than LRT, IRenMF and RankGeoMF. (3) The training time of LFBCA increases super-linearly with the size of training data. Therefore, LFBCA is not preferable for very large datasets.

5.4.2 Querying Scalability

Table 3 shows the average querying time per user of all the models for different user groups, each with different numbers of check-in POIs. In the experiments, querying time represents *the average time of computing the recommendation scores of a user u on all the POIs*. Ranking the POIs is not included in the querying time. For GeoSoCa, since categorical information is not available in the Gowalla data, we test the querying time of its geographical and social components (namely GeoSo) instead. To compare the scalability of different methods with the number of check-in POIs, we show the relative querying time in Figure 14(b), which is defined as the ratio comparing to the time cost for the first user group (i.e., <15).

Table 2: Training time (hour) of the models

Train size	MGM-PFM	L-RT	LF-BCA	IRen-MF	Geo-MF	Rank-GeoFM	Geo-PFM
20%	0.03	0.27	5.23	0.13	7.34	2.84	0.10
40%	0.06	0.32	8.25	0.21	7.38	4.00	0.21
60%	0.09	0.36	12.82	0.28	7.48	5.22	0.32
80%	0.12	0.41	19.20	0.35	7.45	6.29	0.44



(a) Relative training time (b) Relative querying time

Figure 14: Relative training and querying time on Gowalla.

iGSLR. iGSLR has the worst scalability for recommendations. Particularly, it takes only 1.958s for users with fewer than 15 POIs, but 1292.566s for users with more than 100 POIs. This is because the querying time of iGSLR cubically increases with the number of the user's check-in POIs (Table 7, Appendix F [29]). Therefore, iGSLR is not suitable for active users.

Hybrid models vs. Joint-learning models. On the one hand, the querying time of hybrid models usually increases with the number of check-in POIs of a user (except MGMPFM). For example, the querying time of GeoSo increases by 10 times, from 4.977s (for <15) to 42.828s (for 100<). The reason is that the hybrid models separately model geographical information based on users' visited POIs. At recommendation time, the geographical component needs to iterate all the user's visited POIs, and thus the querying time is usually polynomial to the number of visited POIs. On the other hand, the querying time of all the joint-learning models, e.g., IRenMF and GeoMF, is constant for different user groups. This is because these models jointly embed users' preferences and contextual information into fixed-size matrices, and thus the querying time is the same. Therefore, the querying scalability of joint-learning models is usually better than hybrid models.

MGMPFM. Its querying time varies between 3.4s and 6.5s for all 5 user groups, which is more scalable than the other hybrid models. This is because the geographical modeling method of MGMPFM (i.e., MGM) is based on the Gaussian centers $\mathcal{G}_u$ of the user instead of the visited POIs $\mathcal{L}_u$. Even for active users, their check-in POIs are likely to locate around a few number of centers. Hence, MGMPFM can scale better to active users.

5.5 Summary of New Insights

From the evaluation, we observe many interesting findings that have never been reported in any existing work. Those findings are important for understanding POI recommendation models, which helps us to choose and design a suitable model for a particular scenario (e.g., sparse data). We summarize the key findings below.

- RankGeoFM, IRenMF and GeoMF outperform the other models, on different datasets and types of users, and RankGeoFM usually performs the best. Those models (1) are based on implicit feedback models, such as WMF and ranking-based MF, and (2) consider geographical information. In contrast, LRT, which does not possess these two properties, performs worse. Moreover, models without directly modeling user preference based on user-POI check-in matrix (e.g., iGSLR) are not attractive (Figures 2, 3, 4, 6 and 7).
- RankGeoFM is the best model when the check-in data is dense (i.e., the sparsity is larger than 0.0004), and is followed by IRenMF

Table 3: Avarage querying time (second) per user of the models for users with different numbers of check-in POIs

# of POIs	USG	MGMPFM	LRT	iGSLR	LFBCA	LORE	IRenMF	GeoMF	RankGeoFM	GeoPFM	GeoSo	ASMF
<15	3.95	3.40	1.45	1.96	2.05	7.15	0.80	0.28	9.65	0.25	4.98	0.19
15–30	4.79	4.57	1.45	3.53	2.05	10.66	0.80	0.28	9.64	0.25	7.82	0.19
30–50	6.38	6.06	1.45	11.04	2.04	17.63	0.80	0.28	9.61	0.25	13.27	0.19
50–100	9.2	6.48	1.45	56.61	2.04	30.28	0.80	0.28	9.63	0.25	22.79	0.19
100<	18.20	5.08	1.44	1292.57	2.04	72.54	0.79	0.28	9.58	0.25	42.83	0.19

and GeoMF. However, GeoMF is better than all other methods for sparse data (i.e., the sparsity is as low as 0.0002) (Figure 5).

- GeoPFM is the least sensitive model to the sparsity change, while the accuracy of all the other models decreases dramatically when the data becomes sparser (Figure 5).
- LORE experiences the largest increase when users have more POIs, which is mainly attributed to its sequential modeling (Figure 6).
- The accuracy of all geographical models decrease for larger user activity ranges (Figure 7).
- PD+ is the best geographical modeling method for all the user groups, and PD usually performs the second best. This indicates that utilizing power-law distribution is an effective solution for geographical modeling (Figures 8 and 9).
- Methods that model personalized two-dimensional check-in distributions (i.e., AKDE and MGM) outperform PD, only for users with more than 100 check-in POIs. Because personalized two-dimensional distribution requires more data to precisely capture users' behaviors. Thus, it is preferable for active users (Figure 9).
- FBCA and CI-FCF are the two best models for social modeling. FBCA performs the best for users with 1 friend, while CI-FCF is better for users with more than 5 friends. FCF methods based on common friends (i.e., CN-FCF) and geographical distance (i.e., D-FCF) do not show promising performances (Figures 10 and 11).
- LBCA performs the best (by over 7%) for user preference modeling, followed by UCF, WMF and BPRMF, among which the difference is within 6%. These 4 models are better than the other methods. This also explains why the models based on these preference modeling methods (e.g., RankGeoFM and GeoMF) outperform the other models. Hence, to develop new POI recommendation models in the future, it would be more promising to extend LBCA, UCF, WMF or BPRMF for recommendations (Figures 12 and 13).
- For training scalability, GeoMF is the most scalable model, whose training time is constant to the size of training data; LFBCA scales super-linearly to the size of training data; The other models scale linearly to the size of training data (Figure 14(a)).
- For querying scalability, joint-learning models (i.e., LRT, IRenMF, GeoMF, RankGeoMF and GeoPFM) are usually more scalable than hybrid models (e.g., iGSLR and GeoSoCa), with respect to the number of check-in POIs of the user (Figure 14(b)).

6. RELATED WORK

In this section, we review the existing POI recommendation studies. We first categorize them based on the methods they use, and then introduce other POI recommendation problems with different settings. The details of the related work on other POI recommendation problems are included in Appendix G of the full version [29]. For the first time, Table 4 presents a comprehensive overview of these studies from recommendation problems, recommendation models, and context information used.

6.1 POI Recommendation

Collaborative Filtering. In one of early work on POI recommendations [46], Ye et al. applied User-based CF (UCF) and Friend-based CF (FCF) to model user preference in their model (included in our evaluation). Subsequently, UCF and FCF have been utilized in many POI recommendation models [45, 46, 54, 57, 58]. Additionally, item-based CF (ICF) is also used for POI recommendations [19, 37] and ICF is shown to perform worse than UCF [46].

Link-based Methods. In POI recommendations, Ying et al. [49] propose a HITS-based model that considers the links between users and POIs, and Noulas et al. [34] include both user-user friendships and user-POI links. Wang et al. [41] further include relations between spatially similar users as the edges in user-user graph, and use them for recommendations.

Factorization Models. Different MF methods have been leveraged for POI recommendations, such as MF [11], WMF [30, 24, 20], BNMF [25] and ranking-based MF [22]. In addition, Bhargava et al. [4] also apply tensor decomposition to model user preference.

Probabilistic Models. Probabilistic models are usually represented graphically to describe the interplay among variables, such as the mutual effects between geographical influence and user interests in POI recommendations. On the one hand, Poisson Factor Model (PFM) is first adopted for POI recommendation [6] and then extended to include context information [26]; On the other hand, spatial topic models are used in modeling user's latent interests and integrating users' context preferences [18, 14, 25, 59].

In our evaluation, most representative and state-of-the-art models in each type of methods are considered.

6.2 Other POI Recommendation Problems

We next introduce three variants of the POI recommendation problem, which take additional information as part of input. They aim to fulfill more specific needs of users. More details can be found in Appendix G in the full version [29].

Next POI Recommendation. Given a user and his/her current location, next POI recommendation aims at *recommending new POIs that are likely to be visited by the user in the next time interval (e.g., in the next 6 hours)* [7, 18, 9, 60, 13, 28]. Most of the next POI recommendation models employ sequential information between consecutive check-ins to recommend. Additionally, models developed for location prediction can also be applied to next POI recommendation [32, 38, 51, 53, 33, 43].

Time-aware POI Recommendation. Considering that user preference varies with time, given a user a time (e.g., 5 p.m.), time-aware POI recommendation *returns new POIs that are most likely to be visited at the time to users* [50, 52, 22, 8].

In-town/Out-of-town POI Recommendation. Given a user's hometown and current location or city, in-town/out-of-town POI recommendation *returns new POIs to the user, using different recommendation strategies when the user is in-town or out-of-town* [10, 47, 48, 42]. Except for the earliest work [10], all the other models for in-town/out-of-town POI recommendation rely on content information (e.g., tags and categories).

There are other types of POI recommendation, such as category-aware [27] and requirement-aware [51, 53] POI recommendation.

Acknowledgment This work was carried out at the Rapid-Rich Object Search (ROSE) Lab at the Nanyang Technological University, Singapore. The ROSE Lab is supported by the National Research Foundation, Prime Ministers Office, Singapore, under its IDM Futures Funding Initiative and administered by the Interactive and Digital Media Programme Office. The

Table 4: Summary of existing POI recommendation papers (sorted by publication years)

		Problem					Methodology					Information-used					
	Related work	POI rec.	Time-aware	Next POI	In-/Out-of-town	Others	Link-based	CF	Factorization	Probabilistic	Hybrid	Geographical	Social	Textual	Categorical	Sequential	Temporal
1	Ye et al. - GIS'10 [45]	✓						✓					✓				
2	Ye et al. - SIGIR'11 [46]	✓						✓			✓	✓	✓				
3	Noulas et al. - SocialCom-PASSAT'12 [34]	✓					✓						✓				
4	Cheng et al. - AAAI'12 [6]	✓								✓	✓	✓	✓				
5	Levandoski et al. - ICDE'12 [19]	✓						✓				✓					
6	Gao et al. - RecSys'13 [11]	✓							✓								✓
7	Hu et al. - RecSys'13 [14]	✓								✓		✓		✓			
8	Zhang et al. - GIS'13 [54]	✓						✓			✓	✓	✓				
9	Wang et al. - GIS'13 [41]	✓					✓					✓	✓				
10	Yuan et al. - SIGIR'13 [50]		✓					✓				✓					✓
11	Liu et al. - KDD'13 [25]	✓							✓			✓		✓			
12	Liu et al. - CIKM'13 [27]					✓			✓			✓			✓		
13	Ference et al. - CIKM'13 [10]				✓			✓				✓	✓				
14	Kurashima et al. - WSDM'13 [18]	✓		✓						✓		✓		✓			
15	Cheng et al. - IJCAI'13 [7]			✓					✓			✓				✓	
16	Yin et al. - KDD'13 [47]				✓					✓		✓		✓			
17	Yuan et al. - KDD'13 [51]					✓				✓		✓		✓		✓	
18	Zhang et al. - GIS'14 [57]	✓						✓			✓	✓	✓			✓	
19	Liu et al. - CIKM'14 [30]	✓							✓			✓					
20	Yuan et al. - CIKM'14 [52]		✓				✓					✓					✓
21	Lian et al. - KDD'14 [24]	✓							✓			✓					
22	Ying et al. - TIST'14 [49]	✓					✓						✓		✓		
23	Sarwat et al. - TKDE'14 [37]	✓						✓				✓					
24	Li et al. - SIGIR'15 [22]	✓	✓						✓			✓					✓
25	Zhang et al. - SIGIR'15 [55]	✓									✓	✓	✓		✓		
26	Zhang et al. - TIST'15 [56]	✓									✓	✓				✓	✓
27	Zhang et al. - CIKM'15 [58]	✓						✓			✓	✓	✓				
28	Liu et al. - TKDE'15 [26]	✓								✓		✓					
29	Gao et al. - AAAI'15 [12]	✓							✓					✓			
30	Lian et al. - ICDM'15 [23]	✓							✓					✓			
31	Li et al. - ICDM'15 [21]	✓							✓			✓	✓				
32	Zhao et al. - ICDE'15 [59]	✓								✓		✓		✓	✓		
33	Feng et al. - IJCAI'15 [9]			✓						✓		✓				✓	
34	Wang et al. - KDD'15 [42]				✓					✓		✓		✓	✓		
35	Yin et al. - CIKM'15 [48]				✓					✓		✓		✓	✓		✓
36	Yuan et al. - TOIS'15 [53]					✓				✓		✓		✓		✓	
37	Chen et al. - AAAI'15 [5]					✓		✓				✓					✓
38	Li et al. - KDD'16 [20]	✓							✓			✓	✓		✓		
39	Zhao et al. - AAAI'16 [60]			✓					✓			✓					✓
40	He et al. - AAAI'16 [13]			✓					✓			✓				✓	
41	Liu et al. - KDD'16 [28]			✓						✓						✓	✓

work is also supported in part by a Tier-2 grant (MOE-2016-T2-1-137) awarded by Ministry of Education Singapore.

7. REFERENCES

[1] G. Adomavicius and A. Tuzhilin. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE TKDE*, 17(6):734–749, 2005.

[2] R. M. Bell and Y. Koren. Lessons from the netflix prize challenge. *ACM SIGKDD Explorations Newsletter*, 9(2):75–79, 2007.

[3] P. Berkhin. Bookmark-coloring algorithm for personalized pagerank computing. *Internet Mathematics*, 3(1):41–62, 2006.

[4] P. Bhargava, T. Phan, J. Zhou, and J. Lee. Who, what, when, and where: Multi-dimensional collaborative recommendations using tensor factorization on sparse user-generated data. In *WWW*, pages 130–140. IW3C2, 2015.

[5] X. Chen, Y. Zeng, G. Cong, S. Qin, Y. Xiang, and Y. Dai. On information coverage for location category based point-of-interest recommendation. In *AAAI*, pages 37–43, 2015.

[6] 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.

[7] C. Cheng, H. Yang, M. R. Lyu, and I. King. Where you like to go next: Successive point-of-interest recommendation. In *IJCAI*, volume 13, pages 2605–2611, 2013.

[8] R. Deveaud, M.-D. Albakour, C. Macdonald, I. Ounis, et al. Experiments with a venue-centric model for personalisedand time-aware venue suggestion. In *CIKM*, pages 53–62. ACM, 2015.

[9] 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.

[10] G. Ference, M. Ye, and W.-C. Lee. Location recommendation for out-of-town users in location-based social networks. In *CIKM*, pages 721–726. ACM, 2013.

[11] H. Gao, J. Tang, X. Hu, and H. Liu. Exploring temporal effects for location recommendation on location-based social networks. In *RecSys*, pages 93–100. ACM, 2013.

[12] H. Gao, J. Tang, X. Hu, and H. Liu. Content-aware point of interest recommendation on location-based social networks. In *AAAI*, pages 1721–1727, 2015.

[13] 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.

[14] B. Hu and M. Ester. Spatial topic modeling in online social media for location recommendation. In *RecSys*, pages 25–32. ACM, 2013.

[15] L. Hu, A. Sun, and Y. Liu. Your neighbors affect your ratings: on geographical neighborhood influence to rating prediction. In *SIGIR*, pages 345–354. ACM, 2014.

[16] Y. Hu, Y. Koren, and C. Volinsky. Collaborative filtering for implicit feedback datasets. In *ICDM*, pages 263–272. IEEE, 2008.

[17] Y. Koren, R. Bell, C. Volinsky, et al. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37, 2009.

[18] T. Kurashima, T. Iwata, T. Hoshide, N. Takaya, and K. Fujimura. Geo topic model: joint modeling of user's activity area and interests for location recommendation. In *WSDM*, pages 375–384. ACM, 2013.

[19] J. J. Levandoski, M. Sarwat, A. Eldawy, and M. F. Mokbel. Lars: A location-aware recommender system. In *ICDE*, pages 450–461. IEEE, 2012.

[20] H. Li, Y. Ge, H. Richang, and H. Zhu. Point-of-interest recommendations: Learning potential check-ins from friends. In *KDD*, pages 975–984. ACM, 2016.

[21] 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.

[22] 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.

[23] D. Lian, Y. Ge, F. Zhang, N. J. Yuan, X. Xie, T. Zhou, and Y. Rui. Content-aware collaborative filtering for location recommendation based on human mobility data. In *ICDM*, pages 261–270. IEEE, 2015.

[24] 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 *KDD*, pages 831–840. ACM, 2014.

[25] B. Liu, Y. Fu, Z. Yao, and H. Xiong. Learning geographical preferences for point-of-interest recommendation. In *KDD*, pages 1043–1051. ACM, 2013.

[26] B. Liu, H. Xiong, S. Papadimitriou, Y. Fu, and Z. Yao. A general geographical probabilistic factor model for point of interest recommendation. *IEEE TKDE*, 27(5):1167–1179, 2015.

[27] X. Liu, Y. Liu, K. Aberer, and C. Miao. Personalized point-of-interest recommendation by mining users' preference transition. In *CIKM*, pages 733–738. ACM, 2013.

[28] Y. Liu, C. Liu, B. Liu, M. Qu, and H. Xiong. Unified point-of-interest recommendation with temporal interval assessment. In *KDD*, pages 1015–1024. ACM, 2016.

[29] Y. Liu, T.-A. N. Pham, G. Cong, and Q. Yuan. An experimental evaluation of point-of-interest recommendation in location-based social networks (full version). In *http://spatialkeyword.sce.ntu.edu.sg/eval-vldb17/*, 2017.

[30] Y. Liu, W. Wei, A. Sun, and C. Miao. Exploiting geographical neighborhood characteristics for location recommendation. In *CIKM*, pages 739–748. ACM, 2014.

[31] H. Ma, C. Liu, I. King, and M. R. Lyu. Probabilistic factor models for web site recommendation. In *SIGIR*, pages 265–274. ACM, 2011.

[32] A. Monreale, F. Pinelli, R. Trasarti, and F. Giannotti. Wherenext: a location predictor on trajectory pattern mining. In *KDD*, pages 637–646. ACM, 2009.

[33] C. I. Muntean, F. M. Nardini, F. Silvestri, and R. Baraglia. On learning prediction models for tourists paths. *ACM TIST*, 7(1):8, 2015.

[34] A. Noulas, S. Scellato, N. Lathia, and C. Mascolo. A random walk around the city: New venue recommendation in location-based social networks. In *SOCIALCOM-PASSAT*, pages 144–153. IEEE, 2012.

[35] R. Pan, Y. Zhou, B. Cao, N. N. Liu, R. Lukose, M. Scholz, and Q. Yang. One-class collaborative filtering. In *ICDM*, pages 502–511. IEEE, 2008.

[36] S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme. Bpr: Bayesian personalized ranking from implicit feedback. In *UAI*, pages 452–461. AUAI Press, 2009.

[37] M. Sarwat, J. J. Levandoski, A. Eldawy, and M. F. Mokbel. Lars*: An efficient and scalable location-aware recommender system. *IEEE TKDE*, 26(6):1384–1399, 2014.

[38] S. Scellato, M. Musolesi, C. Mascolo, V. Latora, and A. T. Campbell. Nextplace: a spatio-temporal prediction framework for pervasive systems. In *International Conference on Pervasive Computing*, pages 152–169. Springer, 2011.

[39] K.-C. Toh and S. Yun. An accelerated proximal gradient algorithm for nuclear norm regularized linear least squares problems. *Pacific Journal of optimization*, 6(615-640):15, 2010.

[40] O. Usatenko. *Random finite-valued dynamical systems: additive Markov chain approach*. Cambridge Scientific Publishers, 2009.

[41] H. Wang, M. Terrovitis, and N. Mamoulis. Location recommendation in location-based social networks using user check-in data. In *SIGSPATIAL*, pages 374–383. ACM, 2013.

[42] W. Wang, H. Yin, L. Chen, Y. Sun, S. Sadiq, and X. Zhou. Geo-sage: a geographical sparse additive generative model for spatial item recommendation. In *KDD*, pages 1255–1264. ACM, 2015.

[43] Y. Wang, N. J. Yuan, D. Lian, L. Xu, X. Xie, E. Chen, and Y. Rui. Regularity and conformity: Location prediction using heterogeneous mobility data. In *KDD*, pages 1275–1284. ACM, 2015.

[44] D. Yang, D. Zhang, and B. Qu. Participatory cultural mapping based on collective behavior data in location-based social networks. *ACM TIST*, 7(3):30, 2016.

[45] M. Ye, P. Yin, and W.-C. Lee. Location recommendation for location-based social networks. In *SIGSPATIAL*, pages 458–461. ACM, 2010.

[46] M. Ye, P. Yin, W.-C. Lee, and D.-L. Lee. Exploiting geographical influence for collaborative point-of-interest recommendation. In *SIGIR*, pages 325–334. ACM, 2011.

[47] H. Yin, Y. Sun, B. Cui, Z. Hu, and L. Chen. Lcars: a location-content-aware recommender system. In *KDD*, pages 221–229. ACM, 2013.

[48] H. Yin, X. Zhou, Y. Shao, H. Wang, and S. Sadiq. Joint modeling of user check-in behaviors for point-of-interest recommendation. In *CIKM*, pages 1631–1640. ACM, 2015.

[49] J. J.-C. Ying, W.-N. Kuo, V. S. Tseng, and E. H.-C. Lu. Mining user check-in behavior with a random walk for urban point-of-interest recommendations. *ACM TIST*, 5(3):40, 2014.

[50] Q. Yuan, G. Cong, Z. Ma, A. Sun, and N. M. Thalmann. Time-aware point-of-interest recommendation. In *SIGIR*, pages 363–372. ACM, 2013.

[51] Q. Yuan, G. Cong, Z. Ma, A. Sun, and N. M. Thalmann. Who, where, when and what: discover spatio-temporal topics for twitter users. In *KDD*, pages 605–613. ACM, 2013.

[52] Q. Yuan, G. Cong, and A. Sun. Graph-based point-of-interest recommendation with geographical and temporal influences. In *CIKM*, pages 659–668. ACM, 2014.

[53] Q. Yuan, G. Cong, K. Zhao, Z. Ma, and A. Sun. Who, where, when, and what: A nonparametric bayesian approach to context-aware recommendation and search for twitter users. *ACM TOIS*, 33(1):2, 2015.

[54] J.-D. Zhang and C.-Y. Chow. igslr: personalized geo-social location recommendation: a kernel density estimation approach. In *SIGSPATIAL*, pages 334–343. ACM, 2013.

[55] 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.

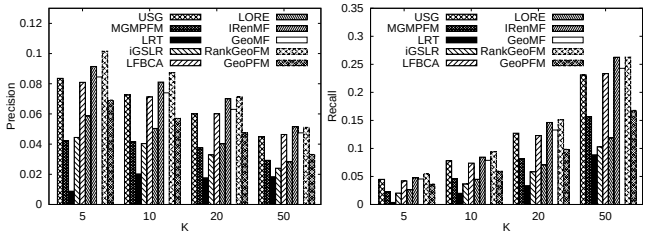
[56] J.-D. Zhang and C.-Y. Chow. Spatiotemporal sequential influence modeling for location recommendations: A gravity-based approach. *ACM TIST*, 7(1):11, 2015.

[57] J.-D. Zhang, C.-Y. Chow, and Y. Li. Lore: Exploiting sequential influence for location recommendations. In *SIGSPATIAL*, pages 103–112. ACM, 2014.

[58] J.-D. Zhang, C.-Y. Chow, and Y. Zheng. Orec: An opinion-based point-of-interest recommendation framework. In *CIKM*, pages 1641–1650. ACM, 2015.

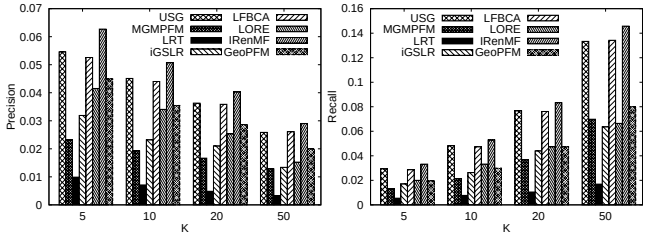
[59] K. Zhao, G. Cong, Q. Yuan, and K. Q. Zhu. Sar: A sentiment-aspect-region model for user preference analysis in geo-tagged reviews. In *ICDE*, pages 675–686. IEEE, 2015.

[60] 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.



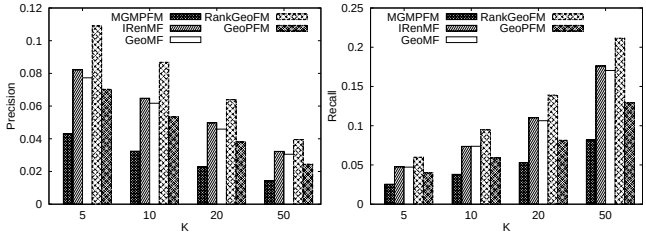
(a) Pre@K - Gowalla (dense) (b) Rec@K - Gowalla (dense)

Figure 17: Varying K on Gowalla (dense).



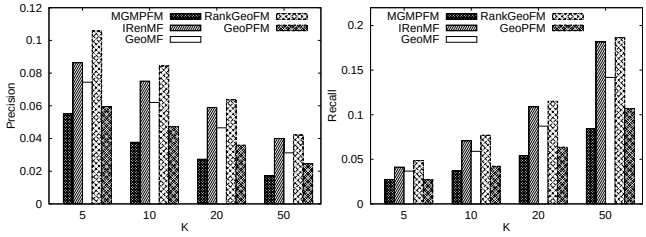
(a) Pre@K - Gowalla (large) (b) Rec@K - Gowalla (large)

Figure 18: Varying K on Gowalla (large).



(a) Pre@K - Gowalla (b) Rec@K - Gowalla

Figure 15: Varying K on Foursquare Singapore.



(a) Pre@K - Gowalla (b) Rec@K - Gowalla

Figure 16: Varying K on Foursquare Tokyo.

APPENDIX

A. EVALUATION METRICS

Pre@K and Rec@K. Given the top-K returned POIs for user u, Pre@K and Rec@K are defined as: $Pre@K = \frac{tp_u}{tp_u+fp_u}$ and $Rec@K = \frac{tp_u}{tp_u+tn_u}$, respectively, where tp_u is the number of recommended POIs that are visited by u (i.e., correct recommendations); fp_u is the number of recommended POIs that are not visited by u (i.e., incorrect recommendations); tn_u is the number of POIs visited by u but not in the top-K recommendations. The average of precision (recall) values of all users is reported.

nDCG@K. For each user, nDCG@K is defined as: $nDCG@K = \frac{DCG@K}{IDCG@K}$, where $DCG@K = \sum_{i=1}^{K} \frac{2^{rel_i}-1}{\log_2(i+1)}$. IDCG@K is the DCG@K value when the recommended POIs are ideally ranked, and rel_i refers to the graded relevance of the result ranked at the position i. nDCG@K is in the range 0 to 1 and higher value means better results. The average of nDCG values of all users is reported.

MAP@K. MAP@K is the arithmetic mean of top-K average precision (AP@K) over all users, which is computed by $MAP@K = \sum_{u=1}^{M} AP@K/M$, where M is the number of users. For each user, $AP@K$ is defined as $AP@K = \frac{1}{K}\sum_{i=1}^{K} Pre@i$, where $Pre@i$ refers to the precision for top-i recommended POIs.

B. PARAMETER SETTING

The notations of the parameters are correspond to the definitions in their original papers.

USG: $\alpha = 0.1$, $\beta = 0.1$, $\eta = 0.05$.

PFM: $K = 30$, $\alpha = 20.0$, $\beta = 0.2$; **MGM:** $\alpha = 0.2$, $\theta = 0.02$, $d = 15$.

LRT: $K = 100$, $\lambda = 1.0$, $\beta = 2.0$, $\alpha = 2.0$, $T = 24$.

LFBCA: $\alpha = 0.85$, $\beta = 0.7$, $\epsilon = 0.001$.

LORE (AMC): $\Delta T = 1\ day$, $\alpha = 0.05$.

GeoMF: $K = 100$, $\delta = 15$, $\gamma = 0.01$, $\lambda = 10$, $\alpha = 0.01$, $\#iters = 10$, $L = 50 \times 50$.

IRenMF: $K = 100$, $\alpha = 0.4$, $\lambda_1 = \lambda_2 = 0.015$, $\lambda_3 = 1$, $\#NN = 10$, $\#clusters = 50$.

RankGeoFM: $K = 100$, $\alpha = 0.6$, $C = 1$, $\epsilon = 0.3$, $\#NN = 300$.

GeoSoCa (AKDE): $\alpha = 0.5$.

GeoPFM: $K = 100$, $\alpha_U = 5$, $\beta_U = 0.2$, $\alpha_V = 20$, $\beta_V = 0.2$ $\#clusters = 50$.

ASMF: $K = 100$, $\alpha = 0.1$, $\lambda_u = 0.01$, $\lambda_v = 0.01$, $\lambda_q = 1$, $\zeta = 0.003$, $\gamma = 10$, $\epsilon = 0.1$.

C. PERFORMANCES ON DIFFERENT CITIES

Figures 15 and 16 show the results on two city-level data, Singapore and Tokyo. The details of the datasets are given in Table 5. From the figures, we can see that the relative performances of the models are qualitatively similar to the results shown in Figures 2, 3 and 4. RankGeoFM, IRenMF and GeoMF are better than the other models, and RankGeoFM usually performs the best. This indicates that RankGeoFM, IRenMF and GeoMF are better for global-level data (Gowalla and Yelp), country-level data (Foursquare) and city-level data (Singapore and Tokyo).

The results also show that the precision and recall values on the city-level datasets are higher than the global-level and country-level datasets. For example, the Pre@5 value of RankGeoFM is 0.11 on Singapore and Tokyo, while are 0.07, 0.06 and 0.03 on Gowalla, Foursquare and Yelp, respectively. There are two possible explanations: (i) the data density of Singapore and Tokyo data are higher than the Gowalla, Foursquare and Yelp data, and (ii) the activity ranges of the users in the city-level datasets are smaller than those in the global-level and country-level datasets.

D. PERFORMANCE ON OTHER DATASETS

Figure 17 shows the results on a denser dataset, namely Gowalla (dense), whose sparsity is 97.527%, which is significantly smaller than the density of all the other datasets in this paper. More details of the dataset can be found in Table 5. The relative performances of the models on the Gowalla (dense) are similar to on the Gowalla data, except that all the models achieve higher precision and recall

Table 5: Data descriptions

Name	# user	# POI	# check-in	Sparsity
Tourist datasets				
Foursquare (tourist)	11,555	28,295	408,958	99.875%
Florence[7]	3,011	927	30178	98.919%
City datasets				
Singapore [22]	2,321	5,596	95,118	99.268%
Tokyo[8]	2,287	7,055	389,029	99.203%
Other different datasets				
Gowalla (dense)	2,426	2,195	221,487	97.527%
Gowalla (large)	41,297	67,738	2,062,996	99.952%

values, which could largely be attributed to the higher density of the data.

Figure 18 shows the results on a larger scale dataset, namly Gowalla (large), which has over 40,000 users and 60,000 POIs. The details of the dataset are given in Table 5. GeoMF and RankGeoFM run out of memory on this data, and thus their results are not available. The results are qualitatively similar to the results on the Gowalla.

E. PERFORMANCE OF RECOMMENDATIONS FOR TOURISTS

The POI recommendation models are designed to uncover users' interests and preferences, based on which recommendations are made. However, the behaviors of tourists are irregular, which might not follow closely with their interests and preferences. For example, tourists usually tend to visit popular places in a new city. To verify this, we have conducted experiments on the original Foursquare dataset (containing both tourists' and natives' check-ins), a tourist Foursquare dataset (the training and test data are constructed based on a previous study [42]) and a tourist dataset named Florence [33]. In addition to the models that are included in the evaluation paper, we also add a MostFreq baseline in the comparison, which always recommends the top-frequent POIs to a user.

Figure 19 shows that the MostFreq baseline performs poorly on original Foursquare data but comparable to the best models on Foursquare (tourist) data and Florence data. Particularly, on original Foursquare data, RankGeoFM is better than MostFreq by 150% in terms of Pre@5. However, on Foursquare (tourist) and Florence, RankGeoFM is better than MostFreq only by 30% and 13%, respectively. Moreover, MostFreq outperforms all the models except RankGeoFM when K is 20 or 50 on the tourist datasets. This is shows that conventional POI recommendation models cannot properly capture the preferences of tourists. This is because conventional POI recommendation models are not specialized for tourists, whose behaviors are more irregular than the majority users of traditional POI datasets, i.e., native users. Therefore, conventional POI recommendation models are not much better than simple baselines such as MostFreq.

F. TIME COMPLEXITY ANALYSIS

USG. USG recommends by summing up the preferences of all the similar users of u on all the POIs. Since u can have at most M similar users, the querying time complexity is $O(MN)$.

MGMPFM. MGMPFM has two individual components, namely MGM and PFM. The training process of MGMPFM is to train PFM, of which the complexity of $O(rN_CK)$ [31]. When making recommendations for user u, the querying complexity PFM is $O(KN)$, and MGM needs to compute the geographical score of each candidate POI, which is based on the Gaussian centers of u, and thus the querying complexity is $O(|\mathcal{L}_u|N)$. The overall querying complexity is $O((|\mathcal{L}_u|+K)N)$.

LRT. Gao et al. claim that the training complexity of LRT is $O(rMNK)$ [11]. However, due to the sparsity of the check-in matrix, it can be rewrite as $O(rN_CK)$. The recommendation process is to sum up all the recommendation scores in different time slots, where the number of time slots is usually a small constant. Therefore, the querying complexity is $O(KN)$.

iGSLR. iGSLR consists of a geographical component (namely 1d-KDE) and a social component (namely D-FCF). The querying process of 1dKDE is to compare the distances of the candidate POIs to u's visited POIs (totally $|\mathcal{L}_u|N$ values) and the distances between every pair of u's visited POIs (totally $|\mathcal{L}_u|^2$ values). Hence, the querying complexity of 1dKDE is $O(|\mathcal{L}_u|^3N)$. D-FCF needs to iterate all the friends of u to aggregate their preferences on N POIs, and thus the complexity is $O(|\mathcal{F}_u|N)$. The overall querying complexity of iGSLR is $O((|\mathcal{L}_u|^3+|\mathcal{F}_u|)N)$.

LFBCA. In each training iteration of BCA, the PPR value of a user is transferred to other connected user. Due to the fact that a user can have at most M connected users, the overall training time complexity of LFBCA is $O(rM^2)$. After obtaining the PPR values of all users, LFBCA recommends POIs to users in a same way with UCF. Therefore, the querying complexity is $O(MN)$.

LORE. LORE consists of a geographical component (namely 2d-KDE), a social component (namely D-FCF) and a sequential component (namely AMC). The querying complexity of D-FCF is the same to in iGSLR, i.e., $O(|\mathcal{F}_u|N)$. 2dKDE takes u's visited POIs as reference points for each candidate POI to compute kernel function values, and thus its complexity is $O(|\mathcal{L}_u|N)$. AMC computes the transition probability of each visited POI to each candidate POI, whose complexity is $O(|\mathcal{L}_u|N)$. Therefore, the overall querying complexity is $O((|\mathcal{L}_u|+|\mathcal{F}_u|)N)$.

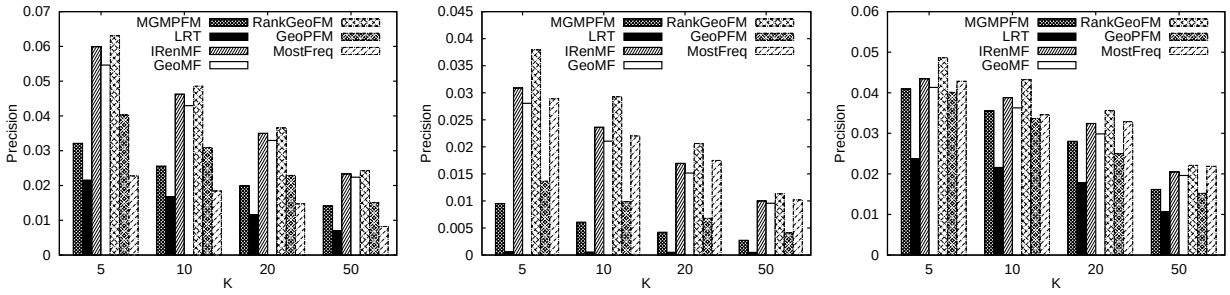
IRenMF. In each training iteration, IRenMF first fixes $\mathbf{L}$ and uses the alternating least squares (ALS) algorithm to update $\mathbf{U}$, of which the time complexity is $O(N_cK^2)$ [16, 35]. Subsequently fixing $\mathbf{U}$, IRenMF uses the accelerated proximal gradient (APG) method [39] to update $\mathbf{L}$. Note that solving the APG optimization problem is also an iterative process and within each iteration, IRenMF performs line search to find a optimal learning rate. The complexity of updating $\mathbf{L}$ is $O(r_Lr_{ls}(NK+N_C))$, where r_L is the number of iterations of APG optimization and r_{ls} is the number of iterations for line search. Therefore, the overall training complexity is $O(r(N_CK^2+r_Lr_{ls}(NK+N_C)))$. When making recommendations, IRenMF includes the user's preference on k nearest POIs. The querying complexity is $O(KkN)$.

GeoMF. The main computation of GeoMF is to iteratively update $\mathbf{U}$, $\mathbf{L}$. In each iteration, the complexity of updating $\mathbf{U}$ and $\mathbf{L}$ is $O(||\mathbf{C}-\mathbf{X}\mathbf{Y}^\top||_0K^2)$, where $||\cdot||_0$ is l_0 norm that used to count the non-zero values in a matrix [24]. When making recommendations, the complexity of computing preference scores and geographical scores are $O(KN)$ and $O(RN)$, respectively. Thus, the querying complexity is $O((K+R)N)$.

RankGeoFM. The main computation of RankGeoFM is to make pairwise comparison of POIs to find incompatibility in the ranking-based learning process. For each positive example from training data, RankGeoFM randomly chooses a POI to construct a negative example and compute the recommendation scores of both examples for comparison. Based on Section 3.1.4, the complexity of computing a recommendation score is $O(Kk)$. Hence, the complexity of making a pairwise comparison till finding incompatibility is $O(Kk\bar{s})$, where $\bar{s}$ is the average number of sampling trials.

[7] http://hpc.isti.cnr.it/ nardini/datasets/LearNext.tar.gz

[8] https://sites.google.com/site/yangdingqi/home/foursquare-dataset



(a) Foursquare - Precision (b) Foursquare (tourist) - Precision (c) Florence - Precision

Figure 19: Varying K on Foursquare, Foursquare (tourist) and Florence.

Table 6: List of notations.

M	the number of users
N	the number of POIs
K	the number of dimensions of the latent space
$\mathbf{C}$	the user-POI check-in matrix
$\mathbf{X}$	the user activity areas matrix (GeoMF)
$\mathbf{Y}$	the POI influence areas matrix (GeoMF)
N_C	the number of non-zero entries in user-POI check-in matrix
r	the number of iterations for learning a model
r_M	the number of iterations for updating $\mathbf{U}$ and $\mathbf{L}$ in the M-step (Geo-PFM)
r_L	the number of iterations for APG optimization (IRenMF)
r_{ls}	the number of iterations for line search (IRenMF)
$\|\mathcal{L}_u\|$	the number of POIs visited by user u
$\|\mathcal{F}_u\|$	the number of friends of user u
$\|\mathcal{G}_u\|$	the number of centers of user u (MGMPFM)
$\|\mathcal{C}_u\|$	the number of categories visited by user u (GeoSoCa)
k	the number of nearby POIs (IRenMF & RankGeoFM)
R	the number of regions (GeoPFM & GeoMF)
$\bar{s}$	the number of trials to find an incompatible example (RankGeoFM)
$\|\|\cdot\|\|_0$	l_0 norm

Table 7: Time complexity of the POI recommendation models.

Models	Training	Querying
USG	N/A	$O(MN)$
MGMPFM	$O(rN_CK)$	$O((\|\mathcal{G}_u\|+K)N)$
LRT	$O(rN_CK)$	$O(KN)$
iGSLR	N/A	$O((\|\mathcal{L}_u\|^3+\|\mathcal{F}_u\|)N)$
LFBCA	$O(rM^2)$	$O(MN)$
LORE	N/A	$O((\|\mathcal{L}_u\|+\|\mathcal{F}_u\|)N)$
IRenMF	$O(r(N_CK^2+r_Lr_{ls}(NK+N_C)))$	$O(Kkn)$
GeoMF	$O(r\|\|\mathbf{C}-\mathbf{X}\mathbf{Y}^\top\|\|_0K^2)$	$O((K+R)N)$
RankGeoFM	$O(rN_CKk\bar{s})$	$O(KkN)$
GeoPFM	$O(rN_CK(r_M+R))$	$O(KN)$
GeoSoCa	N/A	$O((\|\mathcal{C}_u\|+\|\mathcal{F}_u\|+\|\mathcal{L}_u\|)N)$
ASMF	$O(N_CK^2)$	$O(KN)$

On the other hand, the complexity of updating the latent matrices is $O(Kk)$. Thus, the overall training complexity is $O(rN_CKk\bar{s})$. In the querying step, RankGeoFM is similar to IRenMF, considering the user's preference on k nearest POIs. The querying complexity is $O(KkN)$.

GeoPFM. GeoPFM uses a Expectation Maximization (EM) algorithm to estimate the parameters. In E-step, GeoPFM updates the probability distribution of each POI assigned to regions, whose time complexity is $O(N_CKR)$; In M-step, GeoPFM first recomputes the Gaussian distributions of the regions, whose complexity is $O(N)$, and subsequently updates $\mathbf{U}$ and $\mathbf{L}$ iteratively (for r_M iterations), whose complexity is $O(r_MN_CK)$. Therefore, the overall training complexity of GeoPFM is $O(rN_CK(r_M+R))$. The recommendation score of GeoPFM is a geographical-weighted recommendation score of PFM (see Section 3.2.2). The complexity of computing the geographical weight is $O(1)$. Therefore, the querying complexity is $O(KN)$.

GeoSoCa. LORE consists of a geographical component (namely AKDE), a social component (namely SC) and a categorical component (namely CC). The querying complexity of AKDE is same to the 2KDE of LORE, i.e., $O(|\mathcal{L}_u|N)$. SC compute the social check-in frequency of u by aggregating all the check-ins from his/her friends, and thus the complexity is $O(|\mathcal{F}_u|N)$. By using similar method, the complexity of CC to aggregate categorical check-ins is $O(|\mathcal{C}_u|N)$. Therefore, the overall querying complexity is $O((|\mathcal{C}_u|+|\mathcal{F}_u|+|\mathcal{L}_u|)N)$.

ASMF. The time complexity of training ASMF is $O(N_CK)$ [20]. The main computation of ASMF in the query step is the same as MF, i.e., multiplying user and POI latent vectors. Therefore, the time complexity of recommendation is $O(KN)$.

Table 6 shows the definitions of the notations used in the complexity analysis. For the notations that are used in specific models (e.g., the number of nearby POIs used in IRenMF and RankGeoFM, denoted as k), we indicate the model names in the brackets.

Table 7 summarizes the training and querying time complexity of the POI recommendation models. For the training time complexity, we can conclude that: (1) The complexity of LFBCA is significantly higher than the other models; (2) The complexity of MGMPFM, LRT, IRenMF, RankGeoMF and GeoPFM are linear to the number of non-zero values in the check-in matrix (denoted as N_C), i.e., the size of training data. For the querying time complexity, we can conclude that: (1) The complexity of all the models are linear to the number of POIs in the dataset. Because the models need to compute the recommendation score of the given user u on each POI. (2) The complexity of the MF/PFM-based models are linear to the number of dimensions of the latent space (denoted as K). (3) The complexity of the hybrid models are related to the contextual information and the number of context-related observations. For example, LORE models geographical and social information, and thus its querying complexity is related to the number of check-in POIs and the number of friends, i.e., $O((|\mathcal{L}_u|+|\mathcal{F}_u|)N)$; GeoSoCa models geographical, social and categorical information and its querying complexity is $O((|\mathcal{C}_u|+|\mathcal{F}_u|+|\mathcal{L}_u|)N)$. On the contrary, the querying complexity of joint-learning models (i.e., those jointly learn contextual information, such as GeoMF) are independent to the number of context-related observations. Because they already embed contextual information into user preferences in the

training steps and do not need to iterate historical data for querying.

G. RELATED WORK OF OTHER POI RECOMMENDATION PROBLEMS

G.1 Next POI Recommendation

Cheng et al. [7] embed personalized Markov Chain in modeling users' transitions between POIs. Kurashima et al. [18] extend their PLSA-based model for next POI recommendations by considering the Euclidean distance from current location to the candidate POIs. Feng et al. [9] propose a personalized ranking metric embedding method, considering transition probability between POIs as the Euclidean distance between their latent representation in a low-dimensional space. Zhao et al. [60] and He et al. [13] propose ranking-based tensor factorization models for next POI recommendations.

For location prediction, Monreale et al. [32] use a decision tree that is learned from users' trajectory patterns to predict their next visit POIs; Scellato et al. [38] propose a location prediction approach based on nonlinear time series analysis; Yuan et al. [51, 53] propose a unified probabilistic graphical model that can perform location prediction; Muntean et al. [33] employ Gradient Boosted Regression Trees and Ranking SVM on tourism related data to predict the mobility behavior of tourists; Wang et al. [43] design a MF-based method to model users' conforming and regular behaviors for prediction.

G.2 Time-Aware POI Recommendation

Time-aware POI recommendation is first proposed by Yuan et al. [50], and a CF method with time-dependent similarity between users is introduced to solve the problem. Subsequently, Yuan et al. [52] propose a Geographical-Temporal influences Aware Graph method to further improve the recommendation accuracy. Li et al. [22] extend their POI recommendation model (i.e., RankGeoFM in Section 3) by including time-related latent factors, and a tensor decomposition is applied for time-aware recommendations. Deveaud et al. [8] propose a POI-centric method to model the time-aware properties of POIs.

G.3 In-/Out-of-town POI Recommendation

Ference et al. [10] first show that the performances of POI recommendation models for out-of-town users are worse than in-town users by a large margin, and use a unified CF method with different parameter settings for in-town and out-of-town recommendations. Yin et al. [47, 48] use local users' preferences and POI contents for out-of-town users, to overcome users' "cold-start" problem in a new city. Wang et al. [42] design a novel geographical sparse additive generative model and apply different latent topic distributions for both in-town and out-of-town users.