Geographical Relevance Model for Long Tail Point-of-Interest Recommendation

Wei Liu[1,5], Zhi-Jie Wang[1,5], Bin Yao[2,5,6], Mengdie Nie[1,5], Jing Wang[3], Rui Mao[4,6], and Jian Yin[1,5]

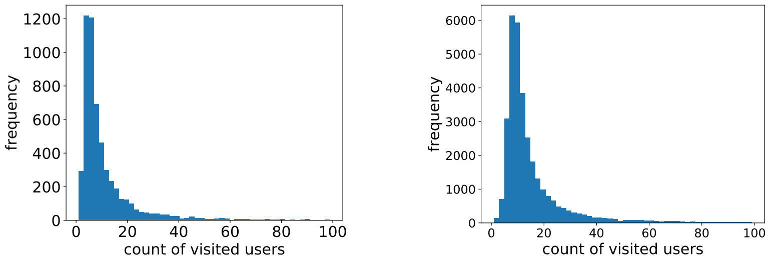
[1] School of Data and Computer Science, Sun Yat-Sen Univeristy, Guangzhou, China
[2] Shanghai Jiao Tong University, Shanghai, China.
[3] Neusoft Institute Guangdong, Foshan, China.
[4] Shenzhen University, shenzhen, China.
[5] Guangdong Key Laboratory of Big Data Analysis and Processing
[6] Guangdong Province Key Laboratory of Popular High Performance Computers
hugh.wei@foxmail.com, wangzhij5@mail.sysu.edu.cn, yaobin@cs.sjtu.edu.cn,
niemengdie@icloud.com, jingyun_wj@163.com, mao@szu.edu.cn,
issjyin@mail.sysu.edu.cn

Abstract. Point of interest (POI) recommendation plays a key role in people's daily life, and has been widely studied in recent years, due to its increasingly applications (e.g., recommending new restaurants for users). One of important phenomena in the POI recommendation community is that, users own a handful of check-ins, while a majority of POIs are visited by few people. This phenomenon is usually called the issue of *data sparsity*, which makes deep impact on the quality of recommendation. Existing works have proposed various models to alleviate the bottleneck of the data sparsity, and most of these works addressed this issue *from the user perspective*. To the best our knowledge, few attention has been made to address this issue *from the POI perspective*. In this paper, we observe that the "blanked" POIs take up a great proportion among all the POIs. It is interesting to investigate whether these blanked POIs can help us improve the quality of recommendation, especially for the long tail POIs (which have a few check-ins, yet have less opportunity to be exposed) recommendation. To this end, this paper proposes a new model, named GRM (geographical relevance model), that wisely uses the geographic information of blanked POIs, addressing the limitations of existing models. Experimental results based on two public datasets demonstrate that our model is effective and competitive. It outperforms state-of-the-art models for the long tail POI recommendation.

Keywords: Long Tail; Relevance Model;Geographical Information; Point-of-Interest Recommendation

1 Introduction

Nowadays, location-based services are widely used in our daily life [12, 38, 40, 23, 39]. For instance, Yelp and Meituan can help individuals discover favourite



(a) Foursquare

(b) Gowalla

Fig. 1. Histogram of the visiting frequency of all POIs (a) distribution on Foursquare. (b) distribution on Gowalla.

foods, shopping malls, hotels, etc. Foursquare and Webchat can assist people discovering fun places, friends' footprint, etc. Almost all these applications incorporate the function of point of interest (POI) recommendation (which is similar to *trip recommendation* [43]). Particularly, POI recommendation can help these applications understand individuals' favourites more deeply, and so it has the potential to provide personalized services for individuals. Besides, it can also help merchants to solicit more potential customers.

In past decades, there are numerous papers studying the problem of POI recommendation [3, 8]. For example, some works used the rich context information (e.g., geographical information [14, 42, 5, 6, 7] and content information [2, 4, 8]) to address data sparsity in POI recommendation. Some works (e.g., [9]) discussed the diversity of POI recommendation. In the existing literature, most of works focused on POI recommendation *from the user perspective* (e.g., [1, 2, 4, 3]). That is, prior works mainly concentrated on developing effective solutions to recommend POIs for individuals (i.e., users). In addition, most of models/methods (e.g., [13, 1]) developed in prior works are inclined to recommend "popular" POIs for users. As such, the *POIs with less check-ins* (known as *long tail POIs*) have less chance to be exposed, incurring a lot of "blanked" POIs. For example, Fig. 1 shows the distribution of check-ins on POIs fro two public datasets "Foursquare and Gowalla" (other details are listed in Table 1). From this Fig. 1, we can see that a few POIs' records are more than 40, yet a majority of POIs have less than 10 records. Clearly, the distribution of check-ins on POIs is **extremely skewed**. This observation is consistent with the findings in recent works [10, 11, 12]. Note that, the frequently visited POIs mostly belong to popular locations, and existing models/systems usually tend to recommend these "popular" POIs. Naturally, this leads most of niche POIs left out (i.e., the blanked POIs appear). It is not hard to understand that the blanked POIs are essentially aroused from the long tail POIs. Thus, the above phenomenon is usually called the *long tail phenomenon.*

Essentially, existing works (e.g., [27, 28]) have already addressed the long tail phenomenon by recommending long tail POIs to users. By doing so, it is benefit for users (e.g., exploring the unfamiliar environment) and also for merchants

Table 1. Details of Two Datasets

Dataset	#User	POI	#Check-in	#Avg.check-in	#Avg.visited	Density
Foursquare	2,321	5,596	194,108	83.6	34.64	1.49%
Gowalla	10,162	24,250	456,988	44.97	18.8	0.18%

(e.g., the promotion of niche POIs contributes to the increase of the revenue). Furthermore, it is worth noting that the good quality of POI recommendation includes not only the accuracy but also other ingredients such as serendipity and novelty. In this regard, recommending long tail POIs to users are also helpful for improving the recommendation quality. It can be seen that, the essence of their works is also from the user perspective, although they recommended long tail POIs instead of "popular" POIs. Specifically, in this paper we attempt to improve the recommendation quality by finding/recommending the best users for each long tail POI. For clarity, we call it **long tail POI recommendation**. To the best of our knowledge, few attention has been made to investigate the long tail POI recommendation (from the POI perspective). It is not hard understand that, this idea should be especially effective for addressing the long tail phenomenon, since it can allow each (current) long tail POIs to match k best users, significantly alleviating the blanked POIs. Loosely speaking, one can roughly conceive that we attempt to fully utilize the blanked POIs to improve the recommendation quality, since our idea is inspired the observation "a lot of blanked POIs appear" and "blanked POIs are essentially aroused from the long tail POIs".

Although the idea above seems to be powerful, there are still many challenges needing to be addressed. (i) For long tail POI recommendation, the recommendation accuracy is also deeply troubled by data sparsity, it could result in the insufficient learning, when the recommender learns user's favour features and POIs' attributive features. (ii) The data structure of user-POI and check-in records is usually extremely skewed. For example, as shown in Table 1, the quantity of POIs is much more than the quantity of users, and the average check-ins of POI is much less. It could be hard to characterize the features of POIs.

To alleviate various challenges and address the long tail recommendation problem, we propose a collaborative filtering approach whose roots lie in pseudo-relevance feedback (a well-known task in the Information Retrieval field) to the long tail POI recommendation problem. To estimate pseudo-relevance feedback accurately and remedy data sparsity, we employ geographical information to obtain a mixed similarity. This approach builds a statistical relevance model of each of the long tail POIs which enables the identification of target users for recommending long tail POIs.

To summarize, our main contribution are as follows:

- We formally define the long tail recommendation problem.
- We present a relevance model to address the long tail POI recommendation problem, in which we combine the similarity with the geographic information to access the "pseudo-relevant" POIs with target POI. Our method

effectively overcomes the data sparsity issues related to users and long tail POIs.
- We conduct extensive experiments to verify the effectiveness of our proposed model. Experimental results consistently demonstrate that the effectiveness of our proposed model.

The rest of this paper is organized as follows: In Section 2, we briefly review related work. In Section 3, we observe the long tail phenomenon in real world datasets. In Section 4, we state the long tail POI recommendation problem formally, and the relevance model is proposed to address the problem. The experimental results are given in Section 5. Finally, we conclude the paper, giving the limitation and future work in Section 6.

2 Preliminaries

In this section, we first review previous works most related to ours (Section 2.1), and then define our problem formally (Section 2.2).

2.1 Related Work

In recent years, *POI recommendation* has attracted much attention due to various applications [18, 17, 20, 21]. In existing works, many researches focused on improving the recommendation accuracy. There are many representative methods such as the memory-based *collaborative filtering* (CF) method [14, 15, 6], the model-based CF methods [16, 5], the weighted matrix factorization based methods [13], etc. Particularly, recent studies have also attempted to adapt implicit feedback data to improve the recommendation quality [1, 17, 18, 8].

It is well known that *data sparsity* makes deep impact on the recommendation quality [5, 18]. To cope with this issue, prior works have proposed many effective techniques by utilizing various context information. The main context information used in the POI recommendation community includes: (**i**) Geographical information [14, 5, 6, 13, 1, 20, 21]. For example, Zhang *et al.* [6] utilized the kernel density estimation to depict individual's personalized geographical distribution, instead of using a universal distribution for all individuals; the personalized *geographical distribution information* is utilized to improve the recommendation accuracy. Ye *et al.* [14] used the power law distribution to character geographical *clustering* [37] phenomenon, and they utilized the characterized *geographic clustering information* to improve the recommendation accuracy. (**ii**) Content information [2, 22, 7, 8, 23, 4]. For instance, Wang *et al.* [7] exploited a generative model to character individual's personal interest and the crowd preference in target region to help user explore unacquainted environment. He *et al.* [8] proposed a two-step method for POI recommendation, which learns individual's category preference first (by a list-wise ranking approach), then selects POI candidates in the recommended category (by spatial influence and category ranking influence). (**iii**) Temporal information [19, 24, 17, 18, 25]. For example, By time of check-in, Yuan et al. [19] and Gao et al. [24] split check-ins into different time bins, learning temporal pattern of individual's preference. Feng et al. [17] and Liu et al. [18] utilized the sequential relationship between two check-ins to recommend POIs for users. (**iv**) Social relationship [36, 14, 27, 10, 2]. For example, Zhang *et al.* [2] they aggregated the check-in frequency of a user's friends on a POI, and modeled the social check-in frequency as a power-law distribution. And (**v**) trajectory information [26, 35]. For example, Wang *et al.* [26] utilized a gravity model to estimate spatial influence using trajectory information.

Among the works mentioned above, the ones highly related to ours could be [14, 5, 6, 13, 1, 20, 21], since both these works and ours utilize the geographic information. Particularly, the work closest to our could be [6], since both their paper and ours: (i) Both these methods belong to collaborative filtering; (ii) With geographical information, some similar probability density functions are utilized to estimate spatial relationship. Nevertheless, our work is different from their work in several aspects at least: (i) Based on collaborative filtering, their work directly utilized similar neighborhoods as candidates, but long tail POI has few recorded users, it's inaccurate to directly employ similar users. Therefore, to remedy data sparsity, our work utilize a relevance model to expand POI's profile first; (ii) other works utilized probability density estimation for user-POI spatial relationship, while we utilize kernel density estimation is to calculate spatial relationship between POIs; (iii) other works employed spatial relationship to represent part of user's rating to POI, while our utilize spatial relationship to expand target POI's neighbors and remedy data sparsity.

Besides the works mentioned above, another type of works [27, 28] are also highly related to ours, since both their papers and ours discuss the long tail issue in POI recommendation. For example, Yin *et al.* [27] represented the user-item information with undirected edge-weighted graph, and extended Hitting Time algorithm to help users find their favourite long tail items. Valcarce *et al.* [28] proposed an item relevance model to help vendors get rid of long tail products. Nevertheless, these models/methods cannot work for our problem, since these works addressed the long tail issue (in POI recommendation) from the user perspective. Instead, in this paper we focus on the long tail issue (in POI recommendation) from the POIs perspective. That is, unlike the traditional POI recommendation, here we are interested in recommending users for long tail POIs, instead of recommending long tail POIs for users. We would like to point that, the work in [44] mentioned the inverted version of the classic POI recommendation problem. Nevertheless, the focus of their paper is still to address how to suggest POIs to users with a better quality. Hence, our work is essentially different from theirs.

2.2 Problem Definition

As discussed before, most of existing works for POI recommendation from the user perspective. That is, they focused on the recommendation task — finding POIs for users. Instead, in this paper we are interested in the inverted version of the classic recommendation task. That is, we want to find best users for each

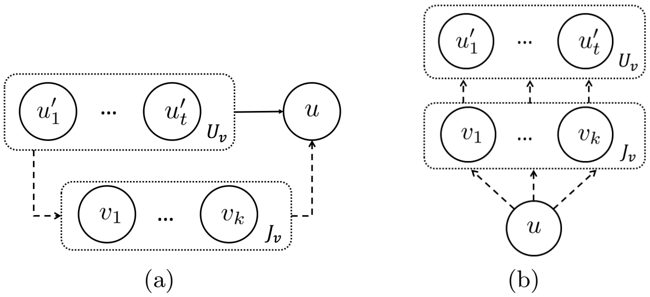


Fig. 2. GRM model. (a) Illustration of Relevance Model R_v for POI v.. (b) Derivation of the estimating process.

long tail POI. Addressing this problem is helpful for alleviating the long tail phenomenon, as mentioned earlier. In what follows, we formally formulate our problem.

Denote by U the set of (all) users, V the set of (all) POIs, and V' the set of long tail POIs, where $V' \subset V$. Similar to [14], let $\boldsymbol{C}$ be the user-POI matrix that represents the relationship between each user and POI. The check-in activity of a user u at a POI v is denoted as $c_{u,v}$. Note that, when $c_{u,v} = 1$, it means u has a check-in at v in the past; otherwise, u has no check-in at v.. For each long tail POI $v \in V'$, we are asked to find a ranked list of n users, L_v^n, that are most likely interested in the long tail POI v. Thus, using the user-POI matrix $\boldsymbol{C}$, our objective is to find a scoring function s': $V' \times U \rightarrow S$. (The set of users visited v is represented by $U_v = \{u'_1, ..., u'_t\}$. Based on U_v, our objective is to compute the probability $p(u|U_v)$ for each user u, and the top-n users would be recommended for long tail POI v, forming a ranked list as L_v^n. Since the long tail POI has few visitors, it is a challenge to estimate $p(u|U_v)$. In the latter section, we would proposed our model to solve it.)

3 Geographical Relevance Model

We first describe the basic idea of our model, and then show how to obtain the important elements used in our model.

▶ **Basic idea.** It is easy to understand that each POI contains some "history" visitors, which can be considered as the *POI profile*. In addition, to measure the relevance, it is necessary to obtain the user's preference. Yet, for the long tail POI, it has few check-in records. Thus, directly obtaining user's preference on the corresponding long tail POI is difficult for traditional collaborative filtering models. To address this issue, our idea is to expand POI profiles via relevant users. Then, the computation of the user preference can be viewed as a process of expanding POI profiles with relevant users, as shown in Fig. 2(a) (see the upper part). Here $U_v = \{u'_1, ..., u'_t\}$ is the set of users who visited the POI v; meanwhile, it also represents the profile of POI v, which shall be used to estimate the probability of a user u.

To expand POI profiles, typically, one can achieve this by employing information from similar (or relevant) POIs. To understand, Fig.2(a) (see the lower part) shows how the corresponding relevance model R_v generates a set of relevant POIs J_v, for a long tail POI v. Note that, throughout this paper we assume users recorded in POI's profile are independent from each other, but dependent on the users recorded in the profiles of POI's neighbors.

▶ **Computing the probability.** To this step, one can easily verify that the long tail POI recommendation can be roughly considered as the problem of computing the probability of a user u under the relevance model of long tail POI v. In what follows, we show how to obtain $p(u|R_v)$ for each user $u \in U$. This step is especially important, since it serves as a vital ingredient of our model. At a high level, our technique for estimating the probability is inspired by [29], in which a relevance-based Language Models is developed for pseudo-relevance feedback. Specifically, the derivation process is as follows:

• Since the probability of next user u (sampled from the relevance model) relies on the set of users $U_v = \{u'_1, ..., u'_t\}$ who visited the POI v, one can have $p(u|R_v) \approx p(u|u'_1, ...u'_t)$. By applying the definition of conditional probability, one can have $p(u|u'_1, ...u'_t) = \frac{p(u,u'_1,...u'_t)}{p(u'_1,...u'_t)}$. Note that, the denominator in the fraction can be ignored, since it remains constant for the same POI v. So, we have

$$p(u|R_v) \approx p(u|u'_1, ...u'_t) = \frac{p(u, u'_1, ...u'_t)}{p(u'_1, ...u'_t)} \propto p(u, u'_1, ...u'_t) \tag{1}$$

• To estimate the relevance model of POI v, R_v, we pick a user u given the prior probability $p(u)$. The sampling probability of users $u'_1, ...u'_t$ will depend on user u , as shown in Eq. 2. To estimate the conditional probability $p(u'|u)$, we sample a user $u' \in U_v$ from the relevant POI's v_j distribution with probability $p(u'|v_j)$, where the selection process of relevant POI will be detailed in latter.

$$p(u, u'_1, ...u'_t) = p(u) \prod_{u' \in U_v} p(u'|u) \approx p(u) \prod_{u' \in U_v} \sum_{v_j \in J_v} p(u'|v_j)p(v_j|u) \tag{2}$$

• By Bayes' Theorem, we have $p(v_j|u) = p(u|v_j)p(v_j)/p(u)$. By combing Formula 2, we have $p(u, u'_1, ...u'_t) \approx p(u) \prod_{u' \in U_v} \sum_{v_j \in J_v} p(u'|v_j) \frac{p(u|v_j)p(v_j)}{p(u)}$. Further, by combing Formula 1, we have

$$p(u|R_v) \propto p(u) \prod_{u' \in U_v} \sum_{v_j \in J_v} p(u'|v_j) \frac{p(u|v_j)p(v_j)}{p(u)} \tag{3}$$

• We next show how to obtain each element in Formula 3. We can see that there are four main elements: $p(u|v_j)$, $p(u'|v_j)$, $p(u)$, and $p(v_j)$. For $p(u)$ and $p(v_j)$, they are considered uniformly ($p(u) = 1/|U|, p(v_j) = 1/|V|$), more elaborate prior probability will be explored in our future work. To compute $p(u|v_j)$, the maximum likelihood estimation of a multinomial distribution over the check-ins can be used for this task. However, this method may suffer from the data sparsity. In our paper, we use Absolute Discounting (AD) [32] to smooth the maximum likelihood estimation $p(u|v_j) = c_{u,v_j} / \sum_{u' \in U_{v_j}} c_{u',v_j}$ with the user probability in the collection. In brief, we use AD to subtract the same constant, δ, from the count of all the seen check-ins. Then, a count (proportional to the probability in the collection) is added to each user. Then, we have

$$p(u|v_j) = \frac{max(c_{u,v_j} - \delta, 0) + \delta |U_{v_j}| p(u|C)}{\sum_{u' \in U_{v_j}} c_{u',v_j}} \tag{4}$$

where $p(u|C)$ is computed as follows:

$$p(u|C) = \frac{\sum_{j \in V} c_{u,v_j}}{\sum_{u' \in U, v_j \in V} v_{u',v_j}} \tag{5}$$

Note that, $p_{(}u'|v_j)$ can be obtained using the same method above.

To this step, it remains to explain how to obtain J_v (i.e., relevant POIs), although we briefly describe it in Fig. 2(a). Note that, J_v is a critical component in our model. We next address how to obtain it in detail.

▶ **Computing relevant POIs J_v.** The key of point for obtaining J_v is the computation of the similarity of v_i and v_j. To achieve this, our method consists of several steps: (i) computing the general similarity of v_i and v_j (in our paper, we use the cosine similarity); (ii) computing the spatial similarity; (iii) combing the "mixed" similarity by combing the above two types of similarities. We next discuss each step.

• Since the similar POIs are the approximation of the real relevant POIs, we call the similar POIs with respect to POI v are the pseudo-relevant POIs in the relevance model R_v. In our paper, we employ kNN algorithm to compute the set of pseudo-relevant POIs based on the check-ins, according to the *pairwise similarity* [33]. Here, we utilize cosine similarity, which yields better results in our experiments. The cosine similarity s between two POIs v_i and v_j is as follows:

$$s(v_i, v_j) = \frac{\sum_{u \in U_{v_i} \cap U_{v_j}} c_{u,v_i} c_{u,v_j}}{\sqrt{\sum_{u \in U_{v_i}} c^2_{u,v_i} \sum_{u' \in U_{v_j}} c^2_{u',v_j}}} \tag{6}$$

• It is easy to understand that, the normalized distance between two POIs can be used as the spatial similarity. However, the spatial similarity is not linear relationship with distance. To obtain POIs' spatial similarity from the distance information and reflect their non-linear relationship, we use the kernel estimation method, which is a non-parametric way to estimate the probability density function of a random variable. The spatial similarity $sp(v_i, v_j)$ is computed as

$$sp(v_i, v_j) = \frac{1}{\sqrt{2\pi} h} e^{-\frac{(d_{v_i v_j})^2}{2h^2}}, \tag{7}$$

where $d_{v_i v_j}$ is the spatial distance between v_i and v_j, the method to select the bandwidth h_* will be introduced in section 4.

• To make our model comprehensive and robust, following prior works [14, 5, 6] we integrates multiple similarity functions into one. Specifically, we here integrate the general similarity and the spatial similarity. For clarity, we denote by $s_m(v_i, v_j)$ the "mixed" similarity, which is computed as follows:

$$\begin{aligned} s_m(v_i, v_j) &= (1-\alpha)s(v_i, v_j) + \alpha \cdot sp(v_i, v_j) \\ 0 &\leq \alpha \leq 1, \end{aligned} \tag{8}$$

where α is a parameter used to balance the weight. Since the above formula is a linear combination of different factors, there is an extra variable α to be inferred. One can easily understand that, when facing new datasets, the variable may need to re-adjust. To address this trouble, our proposed method utilizes the proportion of two factors in exponential space to replace the extra variable. This way, it can avoid this extra variable, as shown below.

$$\begin{aligned} s_m(v_i, v_j) &= \frac{\exp(s(v_i, v_j))}{Z} s(v_i, v_j) + \frac{\exp(sp(v_i, v_j))}{Z} sp(v_i, v_j) \\ Z &= \exp(s(v_i, v_j)) + \exp(sp(v_i, v_j)). \end{aligned} \tag{9}$$

4 Performance Evaluation

In this section, we first cover the experimental settings including datasets, evaluation metrics and benchmark models, and then discuss the experimental results.

4.1 Experimental Settings

In the experiments, we utilize two real datasets to evaluate our proposed method. One is Foursquare[1], and another is Gowalla[2]. The Foursquare dataset is made by 2,321 users on 5,596 POIs from August 2010 to July 2011. In contrast, the Gowalla dataset is produced by 10,162 users on 24,250 POIs from February 2009 to October 2010. Both datasets are very sparse. More details about them can be found in Table 1 (recall Section 1).

Since this paper deals with a novel recommendation task, i.e., an inverted version of the classic POI recommendation problem, in our experiments we consider the POIs with less than 10 visitors as long tail POIs. We randomly select 50% of the visited users as the training set, and the rest of users as a test set. We learn user's preference to POIs from the training set, then we recommend the best candidate users for each long tail POI. The recommendation model is to rate each users unvisited and rank them by the ratings. Then it returns the top-n POIs as the recommendation list (to the POI). By comparing the recommendation list and test set, we assess the model's accuracy. Evaluation metrics include Pre@n and Rec@n, which are computed as

[1] Foursquare is available at https://pan.baidu.com/s/1hrYNwJM

[2] Gowalla is available at https://pan.baidu.com/s/1i4DgFmX

$$\begin{aligned}\text{Pre@}n &= \frac{1}{|V'|}\sum_{v=1}^{|V'|}\frac{|L_v^n \cap T_v|}{n} \\ \text{Rec@}n &= \frac{1}{|V'|}\sum_{v=1}^{|V'|}\frac{|L_v^n \cap T_v|}{|T_v|}.\end{aligned} \tag{10}$$

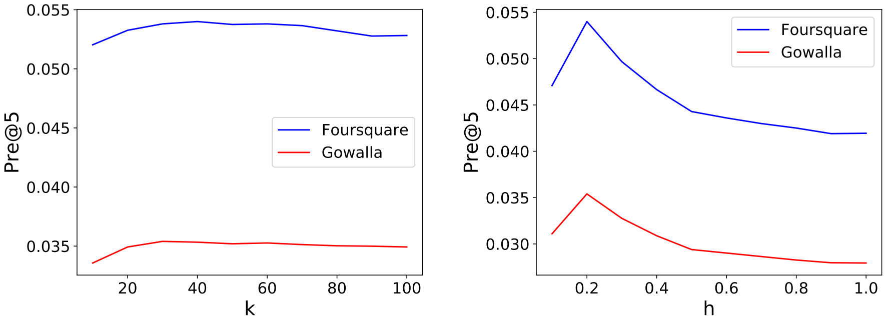
where L_v^n represents the top-n users recommended by the model for POI v, T_v represents the user set really visited POI v. $|V'|$ is the number of long tail POIs. In our experiments, we runs 5 times for each test and report the average value.

To assess the performance of our proposed model, we compare it with five baselines. Since no targeted model exists for the task discussed in this paper, we adapt state-of-the-art models to achieve the task. The details are as follows.

- *Popularity.* This model is a classic and naive recommender model, but is usually used to compare sophisticated models. This model chooses the most popular users for all the POIs in traditional recommendation task. In our task, it implies that, for each POI, it recommends the same set of users.
- *User-based and item-based neighborhood recommenders.* A classic collaborative filtering technique used to compute a set of k nearest neighbors (kNN) for each user or POI. The neighbourhood relationships are computed using pairwise similarities (e.g. Pearson's correlation coefficient). The recommender aims to predict the probability of the target user, based on the check-ins of the neighbourhoods. They conclude user-based (kNN-UB) and item-based (kNN-IB) versions [33]. We use both of version in our experiments. For recommending users to long tail POIs, we generate a recommendation list L_v for each $v \in V'$. This list contains those users $u \in U$ with the largest predicted rating.
- *Hitting time (HT).* This recommender is designed for recommending long tails to users [27], which is contrary to our task. The method overcomes the data sparsity by considering the recommendation task as a random walk in a graph. We utilize this model to build an edge-weighted undirected graph in which the nodes are POIs and users. Each rating is a weight connecting two nodes (i.e., user and POI). Given such a graph, their method computes the hitting time from POI v to target user u, which is the average number of steps that a random walker needs to take (from node v to node u).
- *Rank-GeoFM.* It is a ranking-based model that learns users' preference rankings and includes the geographical influence of neighbouring POIs [1]. This technique is one of the strongest state-of-the-art top-N recommendation model. We use this model to estimate each user's rating on the target POI, and get the recommendation list for each long tail POI.

4.2 Experimental results

To study the impact of bandwidth h in the spatial similarity, we test different settings for this parameter. Fig. 3(b) shows the results of Pre@5 on both



(a) Varying number of neighborhoods k

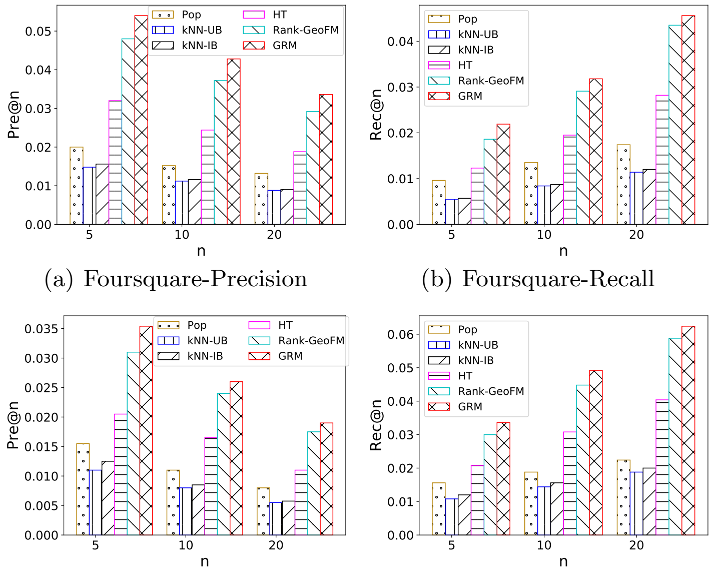
(b) Varying bandwidth h

Fig. 3. Compare the effect of different k and bandwidth on two datasets, Foursquare and Gowalla

datasets. We can see from the figure that GRM first increases and then decreases (for Foursquare), when h increases. Particularly, when $h = 0.2$, its performance reaches the best. One possible reason is that when the distance of two POIs is less than 0.2 km, POI's property is more likely to be similar, but when the distance is more than 0.2 km, the similarity between POI's property will fluctuate drastically. The tendency on Gowalla is similar to that on Foursquare. Furthermore, Fig. 3(a) shows the results by varying k (the number of nearest neighbourhoods). In this set of experiments, we can see that, when k is small, the model has few relevant POIs to expand POI profiles. However, when k exceeds a threshold ($k = 40$ in Foursquare, $k = 30$ in Gowalla), the relevant POIs set would introduce more noise data. At this time, it is not helpful to improve model performance by increasing k. Here the number of nearest neighborhoods k is adopted as 40 in Foursquare, 30 in Gowalla. In the rest of experiments, k is set to 40 and 30 on Foursquare and Gowalla respectively, and parameter h (the bandwidth in the spatial similarity) is set to 0.2, unless stated other wise.

As we expected, the popularity method is poor in our task (cf., Fig. 4). Interestingly, the classic neighbourhood methods (kNN-UB and kNN-IB) perform worse than this naive strategy in some experiments (cf., Fig. 4). It indicates that traditional neighbourhood algorithms could be unsuitable for this task. This is mainly because computing neighbourhoods for long tail POIs is difficult, due to few check-ins. Essentially, this shows the pairwise similarities such as Pearson's correlation coefficient used in neighbourhood method work poor, when only few co-occurrences between vectors are available. On the other hand, it also implies that finding user neighbourhoods who have information about long tail POIs is even more challenging, although finding neighbourhoods for long tail POIs is also challenging.

Furthermore, we observe that the item-based approach (kNN-IB) performs equal or better than the user-based counterpart (kNN-UB) in all the tested scenarios (cf., Figs. 4 and 5). It has been verified in the prior work [33] that, for the



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

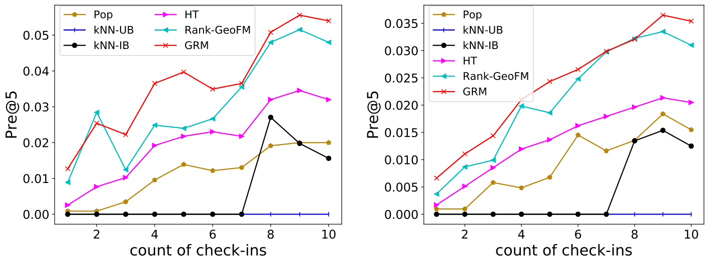
(c) Gowalla-Precision (d) Gowalla-Recall

Fig. 4. Compare the performance of GRM with others approaches on two datasets, Foursquare and Gowalla

traditional recommendation task, item-based approaches tend to achieve better accuracy, compared to most of methods. Fortunately, for the novel recommendation task, we observe from Fig. 4 that the item-based approaches are also not bad.

From Fig. 4, we can see that the performance of the HT method is good. This could be due to the fact that HT computes the average number of steps, which a random walker needs to go from one node to another. This way, this model can generate recommendations for both users and POIs, because all of them are nodes in the same graph connected by check-ins. Thus, this model does not establish any type of difference between users and POIs. The symmetry of this model between both entities is the key of point for achieving the good quality in this novel recommendation task.

Even so, the strongest baseline among these five baseline could be Rank-GeoFM, as shown in Fig. 4. We observe that, Rank-GeoFM produces scores for user and POIs. Thus, the creation of a recommendation list in this task is done by sorting users in the decreased order, with respect to the corresponding POI. The reason Rank-GeoFM produces relatively good recommendation could be that, this method needs no adaptation to our task. Additionally, it utilizes a pairwise ranking method to learn user's and POI's feature vectors. Furthermore, it incorporates geographical information with the influence of nearest geographical neighbourhoods. Last but not least, this method is also symmetric with respect to users and POIs. We conceive that all these properties could be responsible for



(a) Foursquare-Pre@5 (b) Gowalla-Pre@5

Fig. 5. Compare the performance on different long tail POIs on two datasets, Foursquare and Gowalla

the good results of this method. Yet, the symmetry property could be still one of the major reasons.

At the same time, we also test each models' performance for different sparsities. Based on number of check-in records, we divide POIs into ten parts. On Foursquare, we can observe that GRM outperform the rest of baselines consistently, as shown in Fig. 4. The performance of some baselines varies when we change the number of visitors (e.g., Rank-GeoFM outperforms other baselines except when we deal with POIs with very few visitors). Even so, GRM could be more favourable, since it is able to effectively deal with both the high sparsity scenarios and the uncomplicated ones.

In summary, from the above analysis, one can see that, although all the baselines are designed for dealing with the traditional recommendation task — suggesting POIs to users — the results show that some techniques have good performance to the novel task — recommending users to the long tail POIs. Even so, our proposed model GRM is still the best for this task.

5 Conclusions

In this paper, we investigated POI recommendation from the POI perspective. That is, how to recommend potential users for the long tail POIs. We formulated this task formally and designed a novel model to address this problem. Our model is based on the probabilistic collaborative filtering model. We conducted experiments to verify the performance of our proposed model. Experimental results demonstrated that it outperformed a set of representative state-of-the-art recommendation models for our proposed problem. In the future, we would like to incorporate various context information into our model to study the effect of various context information.

References

[1] X. Li, G. Cong, X. Li, T. Pham, S. Krishnaswamy: Rank-geofm: A ranking based geographical factorization method for point of interest recommendation. SIGIR.

433–442 (2015)

[2] J. Zhang, C. Chow: Geosoca: Exploiting geographical, social and categorical correlations for point-of-interest recommendations. SIGIR. 443–452 (2015)

[3] S. Hosseini, H. Yin, M. Zhang, X. Zhou, S. Sadiq: Jointly Modeling Heterogeneous Temporal Properties in Location Recommendation. DASFAA. 490–506 (2017)

[4] H. Yin, X. Zhou, B. Cui, H. Wang, K. Zheng, Q. Nguyen: Adapting to User Interest Drift for POI Recommendation. TKDE. 28(10), 2566–2581 (2016)

[5] C. Cheng, H. Yang, I. King, M. Lyu: Fused Matrix Factorization with Geographical and Social Influence in Location-Based Social Networks. AAAI. 12, 17–23 (2012)

[6] J. Zhang, C. Chow: iGSLR: personalized geo-social location recommendation: a kernel density estimation approach. SIGSPATIAL. 334–343 (2013)

[7] W. Wang, H. Yin, L. Chen, Y. Sun, S. Sadiq, X. Zhou: Geo-sage: A geographical sparse additive generative model for spatial item recommendation. SIGKDD. 1255–1264 (2015)

[8] J. He, X. Li, L. Liao: Category-aware Next Point-of-Interest Recommendation via Listwise Bayesian Personalized Ranking. IJCAI. 1837–1843 (2017)

[9] X. Chen, Y. Zeng, G. Cong, S. Qin, Y. Xiang, Y. Dai: On information coverage for location category based point-of-interest recommendation. AAAI. 37–43 (2015)

[10] A. Noulas, S. Scellato, N. Lathia, C. Mascolo: A Random Walk around the City: New Venue Recommendation in Location-Based Social Networks. SocialCom/PASSAT. 144-153 (2012)

[11] M. Lichman, P. Smyth: Modeling human location data with mixtures of kernel densities. SIGKDD. 35–44 (2014)

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

[13] D. Lian, C. Zhao, X. Xie, G. Sun, E. Chen, Y. Rui: GeoMF: joint geographical modeling and matrix factorization for point-of-interest recommendation. SIGKDD. 831–840 (2014)

[14] M. Ye, P. Yin, W. Lee, D. Lee: Exploiting geographical influence for collaborative point-of-interest recommendation. SIGIR. 325–334 (2011)

[15] J. Levandoski, M. Sarwat, A. Eldawy, M. Mokbel: Lars: A location-aware recommender system. ICDE. 450–461 (2012)

[16] V. Zheng, Y. Zheng, X. Xie, Q. Yang: Collaborative location and activity recommendations with gps history data. WWW. 1029–1038 (2010)

[17] S. Feng, X. and Li, Y. Zeng, G. Cong, Y. Chee, Q. Yuan: Personalized Ranking Metric Embedding for Next New POI Recommendation. IJCAI. 2069–2075 (2015)

[18] Q. Liu, S. Wu, L. Wang, T. Tan: Predicting the Next Location: A Recurrent Model with Spatial and Temporal Contexts. AAAI. 194–200 (2016)

[19] Q. Yuan, G. Cong, A. Sun: Graph-based Point-of-interest Recommendation with Geographical and Temporal Influences. CIKM. 659–668 (2014)

[20] B. Liu, Y. Fu, Z. Yao, H. Xiong: Learning geographical preferences for point-of-interest recommendation. SIGKDD. 1043–1051 (2013)

[21] B. Liu, H. Xiong, S. Papadimitriou, Y. Fu, Z. Yao: A General Geographical Probabilistic Factor Model for Point of Interest Recommendation. TKDE. 27(5), 1167–1179 (2015)

[22] J. Zhang, C. Chow, Y. Zheng: ORec: An Opinion-Based Point-of-Interest Recommendation Framework. CIKM. 1641–1650 (2015)

[23] H. Gao, T. Tang, X. Hu, H. Liu: Content-aware point of interest recommendation on location-based social networks. AAAI. 1721–1727 (2015)

[24] H. Gao, J. Tang, X. Hu, H. Liu: Exploring temporal effects for location recommendation on location-based social networks. RecSys. 93–100 (2013)

[25] Y. Liu, C. Liu, B. Liu, M. Qu, H. Xiong: Unified Point-of-Interest Recommendation with Temporal Interval Assessment. SIGKDD. 1015–1024 (2016)

[26] Y. Wang, N. Yuan, D. Lian, L. Lin,X. Xie, E. Chen, Y. Rui: Regularity and Conformity: Location Prediction Using Heterogeneous Mobility Data. SIGKDD. 1275–1284 (2015)

[27] H. Yin, B. Cui, J. Li, J. Yao, C. Chen: Challenging the long tail recommendation. PVLDB. 5(9), 896–907 (2012)

[28] Valcarce, Daniel and Parapar, Javier and Laro Barreiro: Item-based relevance modelling of recommendations for getting rid of long tail products. KBS. 103(C), 41–51 (2016)

[29] Lavrenko, Victor and Croft, W Bruce: Relevance based language models. SIGIR. 120–127 (2001)

[30] J. Parapar, A. Bellogín, P. Castells, Á. Barreiro: Relevance-based language modelling for recommender systems. IPM. 49(4), 966–980 (2013)

[31] D. Valcarce, J. Parapar, Á. Barreiro: A study of priors for relevance-based language modelling of recommender systems. RecSys. 237–240 (2015)

[32] C. Zhai, J. Lafferty: A study of smoothing methods for language models applied to information retrieval. TOIS. 22(2), 179–214 (2004)

[33] C. Desrosiers, G. Karypis: A comprehensive survey of neighborhood-based recommendation methods. Recommender systems handbook. 107–144 (2011)

[34] Cremonesi, Paolo and Koren, Yehuda and Turrin, Roberto: Performance of recommender algorithms on top-n recommendation tasks. RecSys. 39–46 (2010)

[35] S. Shang, L. Chen, C. Jensen, J. Wen, P. Kalnis: Searching Trajectories by Regions of Interest. TKDE. 29(7), 1549–1562 (2017)

[36] S. Shang, L. Chen, Z. Wei, C. Jensen, J. Wen, P. Kalnis: Collective Travel Planning in Spatial Networks. TKDE. 28(5), 1132–1146 (2016)

[37] S. Shang, K. Zheng, C. Jensen, B. Yang, B. Kalnis, G. Li, J. Wen: Discovery of Path Nearby Clusters in Spatial Networks. TKDE. 27(6), 1505–1518 (2015)

[38] S. Shang, R. Ding, K. Zheng, C. Jensen, P. Kalnis, X. Zhou: Personalized trajectory matching in spatial networks. VLDB J. 23(3), 449–468 (2014)

[39] Z. Wang, D. Wang, B. Yao, M. Guo: Probabilistic Range Query over Uncertain Moving Objects in Constrained Two-Dimensional Space. TKDE. 27(3), 866–879 (2015)

[40] K. Xie, K. Deng, S. Shang, X. Zhou, K. Zheng: Finding Alternative Shortest Paths in Spatial Networks. TODS. 37(4), 29:1–29:31 (2012)

[41] Z. Wang, B. Yao, R. Cheng, X. Gao, L. Zou, H. Guan, M. Guo: SMe: explicit & implicit constrained-space probabilistic threshold range queries for moving objects. GeoInformatica. 20(1), 19–58 (2016)

[42] S. Shang, J. Liu, K. Zheng, H. Lu, T. Pedersen, J. Wen: Planning unobstructed paths in traffic-aware spatial networks. GeoInformatica. 19(4), 723–746 (2015)

[43] S. Shang, R. Ding, B. Yuan, K. Xie, K. Zheng, P. Kalnis: User oriented trajectory search for trip recommendation. EDBT. 156–167 (2012)

[44] S. Feng, G. Cong, Gao, B. An, Y. Chee: POI2Vec: Geographical Latent Representation for Predicting Future Visitors. AAAI. 102–108 (2017)