



# Mix geographical information into local collaborative ranking for POI recommendation

Wei Liu<sup>1</sup> · Hanjiang Lai<sup>1</sup> · Jing Wang<sup>2</sup> · Geyang Ke<sup>1</sup> · Weiwei Yang<sup>1</sup> · Jian Yin<sup>1</sup>

Received: 6 November 2016 / Revised: 13 November 2017 / Accepted: 21 November 2017 /  
Published online: 15 May 2019  
© Springer Science+Business Media, LLC, part of Springer Nature 2019

## Abstract

Point-of-Interest recommendation is a task of personalized ranking prediction on a set of locations for users. The problem of data sparsity is much severe in POI recommendation, as users usually visit only a few POIs concentrated on a limited number of types, relative to the enormous whole POIs. However, he/she has different personalized favors on each type. Based on this phenomenon, we assume the user-POI matrix is locally low-rank instead of globally low-rank, then we put forward to utilize local collaborative ranking (LCR) for POI recommendation, which could mitigate the sparsity of check-in data. Especially, POIs visited by a user always scatter on limited spatial areas, and POI is usually popular in a local scope. There exists spatial local property in users' check-in behavior. Moreover, to represent the spatial local property, we propose spatial similarity in the first time. With spatial similarity, LCR can find more latent neighborhoods in its local matrices and construct the local matrices much accurately. Besides, user's preference to POI includes not only general favor but also spatial favor. So spatial favor is introduced in our model. We utilize spatial similarity and spatial favor to mix geographical information into local collaborative ranking seamlessly, proposing our model MG-LCR (Mix Geographical information into Local Collaborative Ranking). Experiments show that, MG-LCR model can reflect users' preference to POIs more accurately and outperform the state-of-the-art methods.

**Keywords** Local collaborative ranking · Spatial similarity · Spatial favor · Point-of-Interest recommendation

## 1 Introduction

In recent years, we have witnessed the increased development of location-based social networking (LBSN) services, such as Foursquare, Twitter, Yelp and Sina Weibo. LBSNs allow users to explore their surrounding places through the sharing check-in data. The check-in data contains rich knowledge which can be used for various recommendation problems, e.g.,

---

✉ Jian Yin  
issjyin@mail.sysu.edu.cn

Point-of-Interest (POI) recommendation. The task of POI recommendation is to provide personalized recommendations of places of interest to a user, and it returns his/her unvisited places that he/she may be interested. In this paper, we focus on POI recommendation.

Considerable research efforts [4, 12, 15, 20, 37] have been devoted to develop models for providing accurate locations. The existing POI recommendation methods can be categorized into memory-based and model-based systems. Memory-based systems [27, 34] compute the missing rates of user-item pairs from similar users (user-based systems [27, 34]) or similar items (item-based systems). Compared to memory-based systems, model-based systems [2, 4, 12, 14] learn a model based on the patterns recognized from the check-in data. Although successes, a widely acknowledged challenge of POI recommendation is *data sparsity*. The check-in data is very sparse due to a majority of users only visited a very small number of POIs. When we make personalized recommendation for users, the sparse data may cause low accuracy of recommendation system.

Many researchers exploit to use the context information, e.g., geographical coordinates of POIs, for solving the data sparse problem. Lian et al. [12] incorporated the spatial clustering phenomenon [27] into the weighted matrix factorization model and proposed activity area vectors of users and influence area vector of POIs to augment user's and POI's latent factors in matrix factorization model. Li et al. [15] proposed a ranking based geographical factorization model (Rank-GeoFM) for addressing the data sparsity. Their ranking model used both the with and without check-in data, thus the data sparsity can be alleviated. Zhang et al. [37] proposed to exploit the geographical correlations, social correlations and categorical correlations among users and POIs. Wang et al. [20] utilized multi-fold context information to solve the data sparsity problem. In the pipelines of most existing POI methods that use the geographical information, all assume that the matrix of user-POI is globally low-rank. However, since most of the entries in the user-POI matrix are unknown, it is hard to complete the matrix, and it will give suboptimal solutions of the users' and POIs' latent factors in the factorization model. The assumption of globally low-rank thus still has limitations.

Although the check-in data is very sparse, fortunately, the check-in data always has the local structure for the users who live nearby, that is "everything is related to everything else, but near things are more related than distant thing" [27]. This motivates us to recover more unknown values in user-POI matrix by using the local geographical constraint. In this paper, we propose to recommend the unvisited POIs for a user based on three local properties. 1) **Local low-rank matrix**. Instead of assuming that the user-POI matrix is globally low-rank, we explicitly enforce that the user-POI matrix is locally low-rank [10, 11], which the locality is defined in the vicinity of certain row-column combinations. We construct several low-rank sub-matrices, each sub-matrix corresponds to a local region in the user-POI matrix. To calculate the similarity, the anchor point is used as representation for local region. In the local low-rank model, user-POI sub-matrices are much denser than the original matrix. Thus, the sparsity problem can be relieved. 2) **Spatial similarity**. In the real world, there exists spatial locality in users' behaviour, that is people living in the neighbourhood would have the similar behavior. For example, there are two users *A* and *C*. They live closely. Since there are limited number of check-ins for each user, they have no common check-in. This doesn't mean they are not similar with each other any more. As their active areas overlap, maybe a nearby POI visited by *A* will also be visited by *C* in the future. To represent the spatial locality, we propose a new similarity: spatial similarity. The corresponding method is proposed for calculating the spatial similarity between users (the same to POIs). With the spatial similarity, we can calculate the similarity more accurately between user-POI point with anchor point, then we can find more similar user-POI points undiscovered before and make the local matrix denser. 3) **Spatial favor**. When a user chooses a restaurant, the

distance is an important consideration. In this paper, we propose the spatial favor to model user's spatial preference. We utilize a personalized geographical model to get the spatial favor of user to POI. Compared to the existing works [12, 15] that fuse the spatial factor into their models directly, our model can learn the parameter more accurately. Finally, we use the local collaborative ranking (LCR) [11] with these three factors and propose to mix geographical information into local collaborative ranking (MG-LCR) for the sparse data.

Our contributions in this paper can be summarized as follows:

- We propose a local matrix factorization method that explicitly enforces the user-POI matrix to have a locally low-rank structure. Due to sufficient samples are available in the small neighbourhoods, the local matrix approximation can relieve data sparsity.
- We propose to mix geographical information into local collaborative ranking, in which the spatial similarity and the spatial favor are proposed for modelling the geographical correlations. Our spatial similarity can get more similar user-POI points in the local region and make the local matrix denser. We also utilize personalized geographical model to get the spatial favor of user to POI.
- We conduct extensive evaluations on two benchmark datasets for POI recommendation. The empirical results show that the proposed method outperforms other state-of-the-art methods, including LCR, GeoMF, Rank-GeoFM.

The rest of this paper is organized as follows: In Section 2, we firstly propose locally low-rank model to predict users' rating to POI, then we mix geographical information into local collaborative ranking. The experimental results are given in Section 3. In Section 4, we briefly review related work. Finally, we conclude the paper in Section 5.

## 2 The proposed approach

In this paper, we propose to mix geographical information into local collaborative ranking (MG-LCR) for dealing with small amount of check-in data. The MG-LCR has three building blocks for solving the sparsity of data: 1) local low-rank matrix factorization; 2) spatial similarity and 3) spatial favor. In the following, we will firstly introduce the research problem with required notations, and then we will present the details of these three parts, respectively.

### 2.1 Problem definition

Let  $\mathcal{U}$  be the set of all users, and  $\mathcal{V}$  be the set of all POIs. In POI recommendation,  $S$  is the observed implicit feedback data,  $S \subset \mathcal{U} \times \mathcal{V}$  is available. Each POI is associated with latitude and longitude coordinates  $(x, y)$ . We denote  $\mathcal{V}_u^+$  as the set of POIs that user  $u$  have visited:

$$\mathcal{V}_u^+ := \{v \in \mathcal{V} : (u, v) \in S\}. \quad (1)$$

Each  $v$  in  $\mathcal{V}_u^+$  is also associated with frequency information  $f_v$ , which indicates the frequency of user  $u$  visited POI  $v$ . POI recommendation system's task is to recommend POIs that user has not visited.

To predict users' ratings to unvisited POIs, Li et al. [15] and Lian et al. [12] utilized methods based on matrix factorization, which assumes user-POI matrix is globally low-rank. It measures users' preference by their historical check-ins over the entire POI space. They

used matrix factorization to learn the rating matrix  $X : \mathcal{U} \times \mathcal{V}$ . With matrix factorization,  $X$  is approximated by the matrix product of two low-rank latent matrices

$$\hat{X} := UV^T, \tag{2}$$

where  $U : |\mathcal{U}| \times r$  and  $V : |\mathcal{V}| \times r$  and  $r$  is the rank of latent matrix.

### 2.2 Local low-rank matrix factorization

In fact, as user’s preference includes various aspects, he/she may have a special taste on restaurants, and another personalized favor on sceneries. Analogously, each POI’s attribute contains several aspects, it may provide a good service, but its location is remote. In other words, there exists local property in users’ check-in behavior and POIs’ attribute.

Therefore, we utilize an alternative approach to predict users’ ratings to POIs. By assuming user-POI matrix is locally low-rank instead of globally low-rank, we apply local low-rank matrix approximation to estimate ratings in personalized ranking. Thus,  $X$  is approximated by a number of low-rank matrices. Each low-rank matrix represents one local subgroup, and each of these matrices describes the original matrix for a subgroup of user-POI points [1, 24].

$$\hat{X}_{u,v} = \sum_{t=1}^q w_{u,v}^t \hat{X}_{u,v}^t, \tag{3}$$

where  $q$  is the number of low-rank matrices.  $\hat{X}_{u,v}$  represents the estimation of element in original matrix  $X$ ,  $\hat{X}_{u,v}^t$  is the estimation obtained from  $t$ -th low-rank matrix  $X^t$ ,  $w_{u,v}^t$  denotes the weight of low-rank matrix element  $\hat{X}_{u,v}^t$  in  $X_{u,v}$ . To get  $\hat{X}_{u,v}$  and  $w_{u,v}^t$ , we operate as follow:

**Step 1:** Sample  $q$  anchor points  $(u_{t_1}, v_{t_1}), \dots, (u_{t_q}, v_{t_q}) \in \mathbb{R}^{|\mathcal{U}| \times |\mathcal{V}|}$  uniformly from the training set. Each anchor point will be used to construct corresponding sub-matrix. And it is conceivable that more sophisticated adaptive sample techniques will result in a more accurate model.

**Step 2:** Identify neighborhoods surrounding  $t$ -th anchor point to form a sub-matrix. If the similarity between user-POI point  $(u, v)$  with anchor point  $(u_t, v_t)$  is not less than a threshold, then  $(u, v)$  is a neighborhood of  $t$ -th anchor point. To calculate the similarity between user-POI point with anchor point, we use the product of users’ similarity with POIs’ similarity to represent it. Here we use function  $s(u, u_t)$  to indicate the similarity between users in each point, the same to POIs.

$$\begin{aligned} s(u, u_t) &= \arccos \left( \frac{\langle U_u, U_{u_t} \rangle}{\|U_u\| \cdot \|U_{u_t}\|} \right) \\ s(v, v_t) &= \arccos \left( \frac{\langle V_v, V_{v_t} \rangle}{\|V_v\| \cdot \|V_{v_t}\|} \right) \\ s((u, v), (u_t, v_t)) &= s(u, u_t)s(v, v_t), \end{aligned} \tag{4}$$

where  $s(u, u_t)$  describes user similarity, which can be computed solely based on the partially observed user-POI matrix by cosine similarity. However, the partial observed matrix is very sparse, leading to poor estimates of similarity. Therefore, we factorize  $X$  into latent factor  $U, V$  firstly, then we calculate the cosine similarity between users, the same to POIs (omitted here). Moreover, the similarity can also be calculated by other information which will be introduced in the latter section. As mentioned in [10], we also introduce smoothing

kernel into the similarity. Since Epanechnikov kernel’s performance is better than uniform and triangular kernel, we adapt Epanechnikov kernel as smoothing kernel. The similarity between two users becomes:

$$\begin{aligned}
 K_{h_1}(u, u_t) &= \frac{3}{4}(1 - s(u, u_t)^2)\mathbf{1}[s(u, u_t) < h_1] \\
 K'_{h_2}(v, v_t) &= \frac{3}{4}(1 - s(v, v_t)^2)\mathbf{1}[s(v, v_t) < h_2],
 \end{aligned}
 \tag{5}$$

where  $h_1, h_2$  are bandwidth of smoothing kernel. So the similarity between user-POI point with anchor point becomes

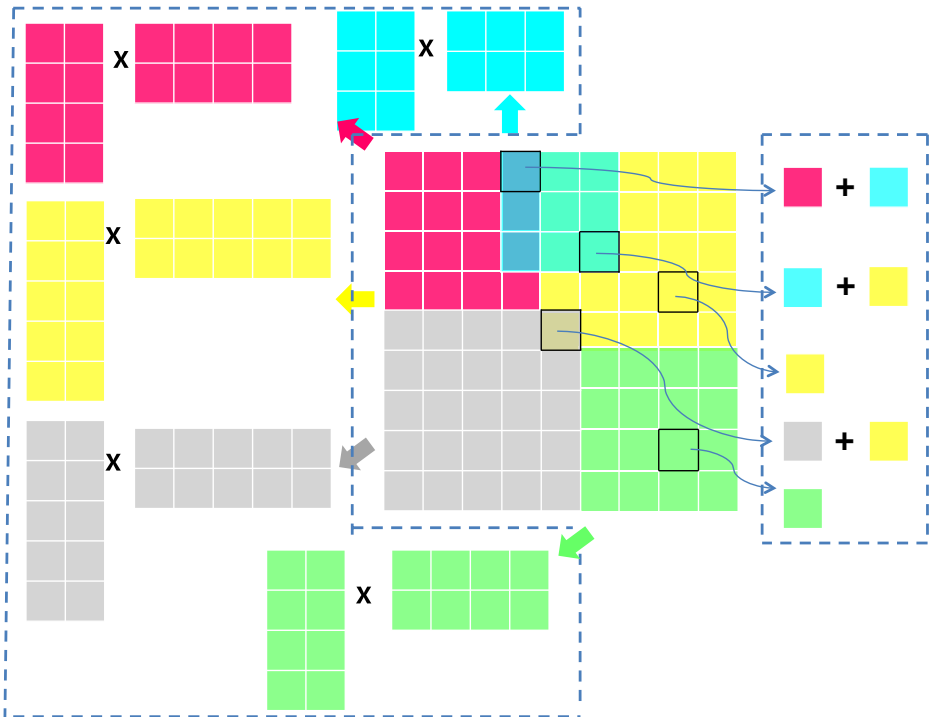
$$s((u, v), (u_t, v_t)) = K_{h_1}(u, u_t)K'_{h_2}(v, v_t).
 \tag{6}$$

**Step 3:** With the similarity  $K(\cdot, \cdot)$ , we can get the weight  $w_{u,v}^t$  as following formula

$$w_{u,v}^t = \frac{s((u, v), (u_t, v_t))}{\sum_{s=1}^q s((u, v), (u_s, v_s))}.
 \tag{7}$$

where  $\hat{X}^t$  can be obtained by the local low-rank matrix, in left part of Figure 1, the sub-matrices are decomposed by matrix factorization:

$$\hat{X}_{u,v}^t = U_u^t V_v^{tT}.
 \tag{8}$$



**Figure 1** Local low-rank matrix factorization. Original matrix is realigned by the anchor points to form sub-matrices, there are five sub-matrices indicated by different colors. Black-edged box indicates the anchor point

**Step 4:** Via local low-rank matrix approximation, the estimation rating of user-POI  $\hat{X}_{u,v}$  will be aggregated, as shown in right part of Figure 1:

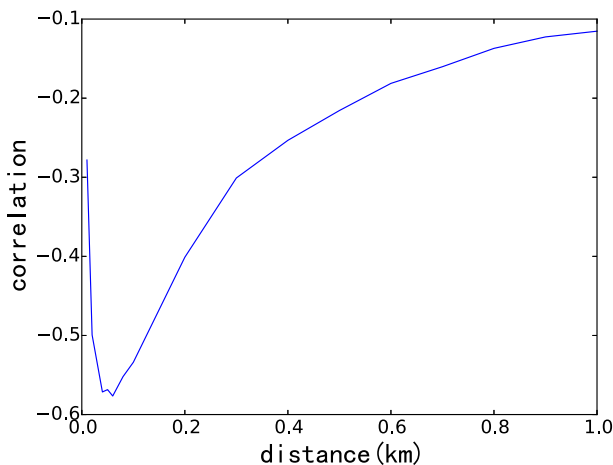
$$\hat{X}_{u,v} = \sum_{t=1}^q \frac{s((u, v), (u_t, v_t))}{\sum_{s=1}^q s((u, v), (u_s, v_s))} [U^t V_t^T]_{u,v}. \tag{9}$$

We calculate the similarity between users (the same to POIs) by latent vector. In fact, besides the general local property, there exists spatial local property for users and POIs in the geographical space. In the following, we first introduce the spatial similarity, then we describe spatial favor in our model. At last, we mix the geographical information into LCR, and provide corresponding learning algorithm.

### 2.3 Spatial similarity

Each user may have different favors in different geographical regions. For example, when user is in Las Vegas, he/she would like casino, having the same favor with local users. While he/she probably likes Lake Powell, if the user is in Phoenix.

At the same time, there is also spatial local property in POIs. Ye et al. [27] mentioned “nearby POIs are more related to each other, which exhibits strong geographical influence.” The adjacent POIs are more or less similar. POIs closed with shopping malls are likely related to shopping. We use data from Yelp Dataset Challenge, including Phoenix, Las Vegas et al. Via latent vector obtained from matrix factorization, we calculate the similarity between POIs by cosine ( $\frac{V_{v_1} \cdot V_{v_2}}{\|V_{v_1}\| \cdot \|V_{v_2}\|}$ ), getting the correlation between distance (POI to POI) and similarity (POI to POI). The  $x$  axis means that POI-POI’s distance is not larger than a corresponding value. In Figure 2, the correlation between distance and similarity is negative, especially when the distance is 100m, the absolute value of correlation is more than 0.55. This indicates when POI to POI’s distance isn’t larger than 100m, they are more likely similar. Thus it can be seen that there exists spatial local property in the geographical



**Figure 2** Correlation between sim ( $v_1, v_2$ ) and distance ( $v_1, v_2$ ) on Yelp. When the POI to POI’s distance is not large than a certain value, we calculate the corresponding correlation value between similarity and distance

space. We can also see when the distance is less than 100m, the absolute value of correlation is not big. This may be because there are not too many POI pairs, of which distance is less than 100m, and there are many noises in the similarities between POIs.

To represent the local property on geographical space, we will introduce spatial similarity, just as similarity for general local property. Next, we will introduce how to compute the spatial similarity. The spatial similarity contains two parts: user’s spatial similarity and POI’s spatial similarity.

*User’s Spatial Similarity* depicts how similar between the active areas of two users. This concept is first proposed in POI recommendation. In earlier works, they used the general similarity to make user-based collaborative filtering recommendation, ignoring that there exists spatial similarity. Through there already exists spatial similarity in other domains, such as image retrieval [6] and biogeography [9], but the spatial similarity in these domains is not same with POI recommendation. Because the check-in data is just some points in spatial area, and each point is associated with frequency information. And we concern their spatial active areas’ relationship, not just their overlapped area. Here we propose a kernel estimation method [13, 34] to calculate it. Defining user  $a$ ’s check-in set as  $L_a = \{(v_1^{(a)}, f_1^{(a)}), (v_2^{(a)}, f_2^{(a)}), \dots, (v_n^{(a)}, f_n^{(a)})\}$ , where  $v_j^{(a)} = \langle x_j, y_j \rangle$ ,  $x_j$  and  $y_j$  represent the latitude and longitude of POI  $v_j^{(a)}$ , which is visited by the user  $f_j^{(a)}$  times. Since user may visit a POI several times, the frequency information is important to depict user’s geographical attribute, and it can reflect the probability of a user occurs in somewhere. Each POI  $v$ ’s spatial similarity with user  $a$  is defined as below:

$$f_s(v|L_a, h) = \frac{1}{|F_a|} \sum_{j=1}^n K_h''(v, v_j^{(a)}) \cdot f_j^{(a)} \tag{10}$$

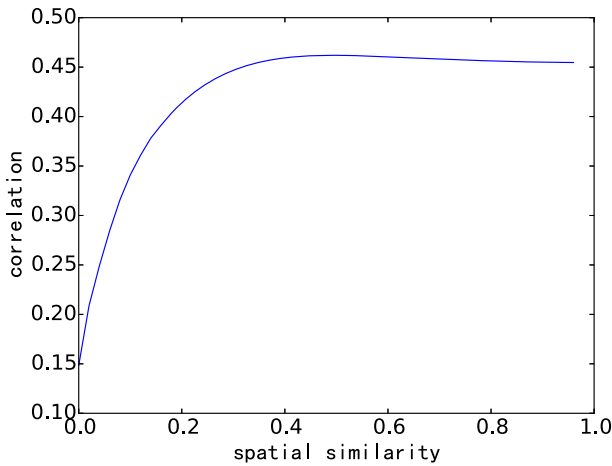
$$K_h''(v, v_j^{(a)}) = \frac{1}{\sqrt{2\pi}h} e^{-\frac{\|v-v_j^{(a)}\|^2}{2h^2}}, \tag{11}$$

where  $|F_a|$  is the sum of all  $f_j^{(a)}$  in  $L_a$ ,  $K_h''$  is a kernel function defined in formula (11),  $h$  is the bandwidth of kernel function,  $\|v - v_j^{(a)}\|^2$  is the spatial distance between  $v$  and  $v_j^{(a)}$ . In a similar way, user  $b$ ’s check-in set is  $L_b = \{(v_1^{(b)}, f_1^{(b)}), (v_2^{(b)}, f_2^{(b)}), \dots, (v_m^{(b)}, f_m^{(b)})\}$ . So the spatial similarity  $sp(a, b)$  between  $a$  and  $b$  will be:

$$\begin{aligned} sp(a, b) &= \frac{1}{|F_b|} \sum_{i=1}^m f_s(v_i^{(b)}|L_a, h) \cdot f_i^{(b)} \\ &= \frac{1}{|F_b||F_a|} \sum_{i=1}^m \sum_{j=1}^n K_h''(v_i^{(b)}, v_j^{(a)}) \cdot f_j^{(a)} \cdot f_i^{(b)} \\ &= \frac{1}{|F_a|} \sum_{j=1}^n f_s(v_j^{(a)}|L_b, h) \cdot f_j^{(a)} \\ &= sp(b, a), \end{aligned} \tag{12}$$

where  $|F_b|$  is the sum of all  $f_i^{(b)}$  in  $L_b$ . From the formula above, we can see the  $sp(a, b)$  is symmetric. So the calculation will be reduced by half.

To prove the assumption that similarity contains general similarity and spatial similarity, we calculate the correlation between user’s general similarity with user’s spatial similarity.



**Figure 3** Correlation between users’ general similarity and spatial similarity on Yelp. When the user to user’s spatial similarity is not large than a certain value, we calculate the corresponding correlation value between general similarity and spatial similarity

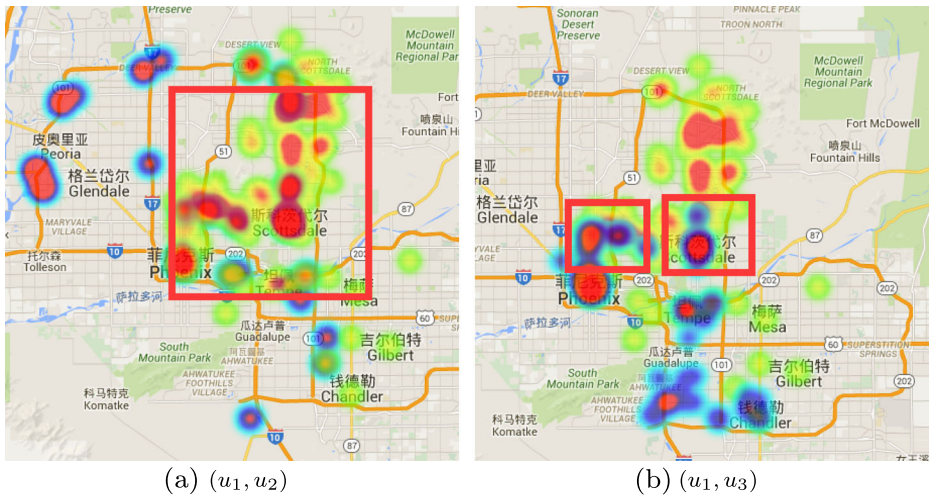
In Figure 3, as the spatial similarity increases, the correlation improves synchronously. This indicates the bigger users’ spatial similarity are, the more similar their preference are. Besides, we show the heatmap of three users’ check-in in Figure 4. In Figure 4(a), user  $u_1$ ’s active area is shown in yellow/green-edge area, user  $u_2$ ’s active area is shown in blue-edge area, their active areas overlap at the center part of the map. Based on the formula of general similarity and spatial similarity, we get general similarity is 0.68, and their spatial similarity is 0.27. In Figure 4(b), user  $u_1$ ’s active area is still shown in yellow/green-edge area, user  $u_3$ ’s active area is shown in blue-edge area. In contrast to  $u_2$ , user  $u_1$ ’s general similarity with  $u_3$ ’s is 0. However, they have two overlapped active areas, and their spatial similarity is 0.25. So the spatial similarity can remedy the insufficient of general similarity. By the two examples, we can see the spatial similarity can depict users’ spatial relationship well. At the same time, the spatial similarity can also reflect users’ similarity properly when general similarity doesn’t work.

*POI’s Spatial Similarity* depicts how similar between two POIs on spatial space. Basically, the normalized distance between two POIs can be used as the spatial similarity. However, the spatial similarity is not linear relationship with distance. Ye et al. [27] utilized a power-law probabilistic model between two POIs’ relationship. But it is based on users’ visited POI set. Here the POI’s spatial similarity is independent with users and is calculated from POI perspective. 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. Defining the POI  $v_i$ ’s coordinate as  $v_i = \langle x_i, y_i \rangle$ , POI  $v_j$ ’s coordinate as  $\langle x_j, y_j \rangle$ , the spatial similarity  $sp(v_i, v_j)$  between  $v_i$  and  $v_j$  will be:

$$sp(v_i, v_j) = \frac{1}{\sqrt{2\pi} h_*} e^{-\frac{(d_{v_i v_j})^2}{2h_*^2}}, \tag{13}$$

where  $d_{v_i v_j}$  is the spatial distance between  $v_i$  and  $v_j$ . How to select the bandwidth  $h_*$  will be introduced in section 2.4.





**Figure 4** Heatmaps of  $(u_1, u_2, u_3)$  on Yelp. In Figure 4(a), user  $u_1$ 's active area is shown in yellow/green-edge area, user  $u_2$ 's active area is shown in blue-edge area, their active areas overlap at the center part of the map. In Figure 4(b), user  $u_1$ 's active area is still shown in yellow/green-edge area, user  $u_3$ 's active area is shown in blue-edge area. There are two overlapped active areas

### 2.4 Spatial favor

As we know, user visits a POI considering not only personalized preference but also the distance factor. As the distance between user and POI becomes smaller, user will probably prefer the nearby POI much more, which we call spatial favor in this paper. To better learn user's personalized preference, we obtain user's spatial favor to POI in advance. Since we cannot get user's current position, we use user's history check-ins to depict user's active area, then calculate distance between target POI with the POIs visited by the user. Ye et al. [27] used power-law distribution to get spatial favor by the distance. However, it needs to learn the parameter of the power-law distribution firstly. Here, we use kernel function to depict user's spatial favor to POI.

$$f_S(v|L_u, h) = \frac{1}{|F|} \sum_{j=1}^n K_h''(v, v_j^u) \cdot f_j \tag{14}$$

$$K_h''(v, v_j^u) = \frac{1}{\sqrt{2\pi}h} e^{-\frac{(d_{vv_j^u})^2}{2h^2}}, \tag{15}$$

where  $v$  is the POI to calculate,  $f_S(v|L_u, h)$  depicts user  $u$ 's spatial favor to POI  $v$ .  $h$  is the bandwidth parameter. Moshe et al. [13] suggested  $h$  be the Euclidean distance of the  $k$ th nearest neighbor to  $v_j^u$  in the training data. However, user's check-in data is very sparse, this method works not very well in the individual's level. Here we utilize fixed  $h$  and perform a grid-search to selecting  $h$  by a validation set. We will use log-probability score function  $LP_u(h)$  (formula is shown below) and precision of top- $k$  for bandwidth selection.

$$LP_u(h) = \frac{1}{n_t} \sum_{r=1}^{n_t} \log f_S(v_r|L_u, h), \tag{16}$$

**Table 1** log-probability scores on validate set, comparing the fixed, adaptive approaches for kernel density estimation on Foursquare check-in data

	Bandwidth	AvgLogL	Precision@5
Fixed	$h = 1$	-1.348	0.031
	$h = 0.1$	-0.873	0.036
	$h = 0.05$	<b>-0.334</b>	<b>0.040</b>
	$h = 0.01$	-0.748	0.037
Adaptive	$k = 2$	-2.674	0.016
	$k = 5$	-0.956	0.032
	$k = 8$	-0.153	0.027

where the  $n_t$  is the number of POIs in the validation set. By Table 1, we can see when  $h = 0.05$ , the performance is best.

### 2.4.1 Mix geographical information into local collaborative ranking

We use local low-rank matrix factorization to predict users' ratings on POIs. The rating can only reflect user's general preference excluding geographical factor. As we know, users tend to visit the POIs around them. Then the spatial favor proposed before can reflect the probability of user appearing at the POI in the spatial perspective. Then we fuse the spatial favor into LCR, getting a new rating of user to POI, which is defined as follows:

$$\hat{X}_{u,v} = \sum_{t=1}^q \frac{s((u, v), (u_t, v_t))}{\sum_{s=1}^q s((u, v), (u_s, v_s))} [U^t V^{tT}_{u,v} + f_S(v|L_u)]. \tag{17}$$

At the same time, when calculating the relationship between user-POI point with anchor points  $s((u, v), (u_t, v_t))$ , we should consider not only the general similarity calculated between latent vectors but also spatial similarity. The similarity  $s(u, u_t), s(v, v_t)$  in formula (5) between users (and POIs) can be updated by the mixed similarity  $s_m(u, u_t), s_m(v, v_t)$  below:

$$\begin{aligned} s_m(u, u_t) &= (1 - \alpha)s(u, u_t) + \alpha \cdot sp(u, u_t) \\ s_m(v, v_t) &= (1 - \alpha)s(v, v_t) + \alpha \cdot sp(v, v_t) \\ 0 &\leq \alpha \leq 1, \end{aligned} \tag{18}$$

where  $\alpha$  is a parameter to balance the weight of spatial similarity. However, this is a linear combination of different factors, and there is an extra variable  $\alpha$  to be inferred. When facing new datasets, the variable needs to be recomputed. Here the proposed method utilizes the proportion of two factors in exponential space to replace the extra variable. As shown below, it can avoid extra variable to infer, the same to POI.

$$\begin{aligned} s_m(u, u_t) &= \frac{\exp(s(u, u_t))}{Z} s(u, u_t) + \frac{\exp(sp(u, u_t))}{Z} sp(u, u_t) \\ Z &= \exp(s(u, u_t)) + \exp(sp(u, u_t)). \end{aligned} \tag{19}$$

### 2.5 Parameter inference

To optimize parameter in the model, we use bayesian analysis of the problem [18], utilizing the likelihood function for  $p(v_i \succ_u v_j | \Theta)$  as objective function, and  $p(\Theta)$  is the prior probability of the model. Our model assumes that a user prefers a visited POI than an unvisited one, and user’s rating to visited POI should be higher than rating to unvisited POI. It learns the POIs’ ranking for each user, so our proposed method is called local collaborative ranking mixed with geographical information. As a result, we will maximize likelihood function  $p(\succ_u | \Theta)$  for all users

$$\prod_{u \in \mathcal{U}} p(\succ_u | \Theta) = \prod_{(u, v_i, v_j) \in \mathcal{U} \times \mathcal{V} \times \mathcal{V}} p(v_i \succ_u v_j | \Theta). \tag{20}$$

In order to get a personalized total rank, we define the individual probability that a user really prefers item  $v_i$  to item  $v_j$  as:

$$\begin{aligned} p(v_i \succ_u v_j | \Theta) &= \sigma(\hat{x}_{uv_i v_j}(\Theta)) \\ \hat{x}_{uv_i v_j}(\Theta) &= \hat{X}_{uv_i}(\Theta) - \hat{X}_{uv_j}(\Theta), \end{aligned} \tag{21}$$

where  $\sigma$  is the logical function:  $\sigma(x) = 1 / (1 + e^{-x})$ ,  $\hat{x}_{uv_i v_j}(\Theta)$  is an arbitrary real function, the model parameter vector  $\Theta$  which captures the special relationship between user  $u$ , item  $v_i$  and item  $v_j$ . Here,  $\hat{X}_{uv_i}(\Theta)$ ,  $\hat{X}_{uv_j}(\Theta)$  is estimated by local collaborative ranking mentioned in Section 2.4.1. In order to complete ranking learning task, we introduce a general prior density  $p(\Theta)$ , its mean is 0, the variance matrix of the normal distribution is  $\sum_{\Theta}$ . Then to reduce the number of unknown parameters, we formulate the maximum posterior (MAP) estimation for the personalized ranking, which is derived below.

$$\begin{aligned} MAP &= \ln p(\Theta | \succ_u) \\ &= \ln p(\succ_u | \Theta) p(\Theta) \\ &= \ln \prod_{(u, v_i, v_j) \in D_s} \ln \sigma(\hat{x}_{uv_i v_j}) p(\Theta) \\ &= \sum_{(u, v_i, v_j) \in D_s} \ln \sigma(\hat{x}_{uv_i v_j}) + \ln p(\Theta) \\ &= \sum_{(u, v_i, v_j) \in D_s} \ln \sigma(\hat{x}_{uv_i v_j}) - \lambda_{\Theta} \|\Theta\|^2 \\ D_s &= \{(u, v_i, v_j) | u \in \mathcal{U} \wedge v_i \in \mathcal{V} \wedge v_j \in \mathcal{V} \setminus \mathcal{V}_u^+\}, \end{aligned} \tag{22}$$

where  $\lambda_{\Theta}$  are model specific regularization parameters.

### 2.6 Learning algorithm

We have just derived an objective function for ranking learning. To learn the objective function, we adopt stochastic gradient descent algorithm. As the objective function is differentiable, the gradient of MAP with respect to the model parameters is:

$$\begin{aligned} \frac{\partial MAP}{\partial \Theta} &= \sum_{(u, v_i, v_j) \in D_s} \frac{\partial}{\partial \Theta} \ln \sigma(\hat{x}_{uv_i v_j}) - \lambda_{\Theta} \frac{\partial}{\partial \Theta} \|\Theta\|^2 \\ &\propto \sum_{(u, v_i, v_j) \in D_s} \frac{-e^{-\hat{x}_{uv_i v_j}}}{1 + e^{-\hat{x}_{uv_i v_j}}} \frac{\partial}{\partial \Theta} \hat{x}_{uv_i v_j} - \lambda_{\Theta} \Theta. \end{aligned} \tag{23}$$

With local collaborative ranking, the target matrix  $X$  is grouped by several local matrices  $X_t$ , each is approximated by the matrix product of two low-rank matrices  $U^t : |\mathcal{U}^t| \times r$  and  $V^t : |\mathcal{V}^t| \times r$ .  $\mathcal{U}^t$  is the user set of local matrix  $X_t$ ,  $\mathcal{V}^t$  is the POI set of local matrix  $X_t$ ,  $r$  is the rank of feature matrix.

$$\begin{aligned} \hat{X}_{u,v} &= w_{u,v}^t \cdot [U^t V_{u,v}^{tT}] \\ w_{u,v}^t &= \frac{s((u, v), (u_t, v_t))}{\sum_{s=1}^q s((u, v), (u_s, v_s))}, \end{aligned} \tag{24}$$

where  $w_{u,v}^t$  represents the weight of local low-rank matrix element  $\hat{X}_{u,v}^t$  in  $\hat{X}_{u,v}$ . Each row  $[U_u^t$  in  $U^t$  can be seen as a feature vector describing a user  $u$  in local matrix and similarly each row  $[V_v^t$  of  $V^t$  describes a POI  $v$ . Thus the prediction formula can also be written as:

$$[U^t V_{u,v}^{tT}] = \sum_{f=1}^r [U_{uf}^t] \cdot [V_{vf}^t]. \tag{25}$$

Then we can get:

$$\frac{\partial}{\partial \Theta} \hat{x}_{uv_i v_j} = \begin{cases} w_{u,v}^t \cdot ([V_{v_i f}^t] - [V_{v_j f}^t]), & \text{if } \Theta = [U^t]_{uf}, \\ w_{u,v}^t \cdot [U_{uf}^t], & \text{if } \Theta = [V^t]_{v_i f}, \\ -w_{u,v}^t \cdot [U_{uf}^t], & \text{if } \Theta = [V^t]_{v_j f}, \\ 0, & \text{else} \end{cases} \tag{26}$$

The stochastic gradient descent algorithm is defined as follow:

**Step 1:** initialize each  $[U^t], [V^t]$  as Gaussian distribution matrix.

**Step 2:** choose the triple  $(u, v_i, v_j)$  randomly (uniformly distributed) from users' check-in set, learn the parameters in  $[U^t]$  and  $[V^t]$  in iteration.

$$\Theta \leftarrow \Theta - \alpha \left( \frac{e^{-\hat{x}_{uv_i v_j}}}{1 + e^{-\hat{x}_{uv_i v_j}}} \cdot \frac{\partial}{\partial \Theta} \hat{x}_{uv_i v_j} + \lambda_{\Theta} \Theta \right). \tag{27}$$

**Step 3:** judge the iteration converged or not. If it converges, then the algorithm finishes, otherwise turn to Step 2.

### 3 Experiment and results

In the experiments, we compare MG-LCR with other recommendation models. First, without considering geographical information, we choose three basic models, weighted matrix factorization (WMF), probabilistic factor model (PFM) and bayesian personalized ranking

(BPR). We use LCR comparing with them. Second, to show the effect of introducing geographical information, we compare MG-LCR with LCR. At last, in order to understand the superiority of our models, we compare MG-LCR with the state-of-art model Rank-GeoFM.

### 3.1 Data sets

Experiments are conducted on two datasets, One is Foursquare<sup>1</sup> check-in data [15], the other is Yelp<sup>2</sup> check-in data. In Foursquare check-in data, there are total 49,062 users, 206,416 POIs and 221,128 check-ins in New York City(NYC). The density of user-POI matrix is  $2.1 \times 10^{-5}$ , average number of visited POIs per user is 4.5, average number of check-ins per POI is 1.07. In Yelp check-in data, there are total 552,000 users, 77,000 POIs and about 2,200,000 check-ins. The density of user-POI matrix is  $5.1 \times 10^{-5}$ , average number of visited POIs per user is 3.98, average number of check-ins per POI is 28.57. We can see that both of the datasets are sparse, and the user-POI matrix in Foursquare is much sparser than Yelp.

### 3.2 Evaluation metrics

Here we use top- $k$  evaluation criterion, for POIs each user accessed, we randomly selected 70% of check-ins as the training set, and 20% as a test set, the rest as validate set. We learn model's parameters from the training set, then we predict where the user may go by recommendation model. The recommendation model is to rate each POI unvisited and rank them by the ratings, then it returns the top- $k$  POIs as recommendation list to the user. By comparing recommendation list and test set, we assess the model's accuracy. Evaluation metrics include  $recall@k$  and  $precision@k$ . The first evaluation metric shows how many percent of POIs user really visited are in the recommendation list for each user. And the second evaluation criterion indicates how many percent of POIs in the recommendation list are really visited by each user. The formulas are defined as below:

$$recall@k = \frac{1}{M} \sum_{u=1}^M \frac{|R_u(k) \cap T_u|}{|T_u|}$$

$$precision@k = \frac{1}{M} \sum_{u=1}^M \frac{|R_u(k) \cap T_u|}{k}. \quad (28)$$

where  $R_u(k)$  represents the top- $k$  POIs recommended by the model for user  $u$ .  $T_u$  represents the POI set really visited by user  $u$ .  $M$  is the number of users. This process is just an experiment of one time. The performance of the final recommendation model is through 5 independent experiments and we get the result by average.

### 3.3 Baseline methods

For POI recommendation, we compare our model with the following baseline methods.

- **Weighted Matrix Factorization(WMF)** WMF is especially designed for matrix factorization based on implicit feedback [8, 17]. WMF considers all the unvisited locations

<sup>1</sup>available at <https://pan.baidu.com/s/1dELJfSP>

<sup>2</sup>available at <https://pan.baidu.com/s/1hrXOpe8>

as negative examples, and it assigns weights of small value to all negative examples, while the weights assigned to positive examples are monotonically increasing respect to frequency.

- **Probabilistic Factor Models(PFM)** Since the distribution of frequency data doesn't obey Gaussian distribution, PFM [2] places Beta distributions as priors on the latent vectors, then PFM defines a Poisson distribution on the frequency.
- **Bayesian Personalized Ranking(BPR)** BPR is the most popular ranking-based matrix factorization, our local collaborative ranking model is also improved based on the Bayesian Personalized Ranking criterion [18].

For integrated models considering geographical information on POI recommendation, the following methods are used as baselines.

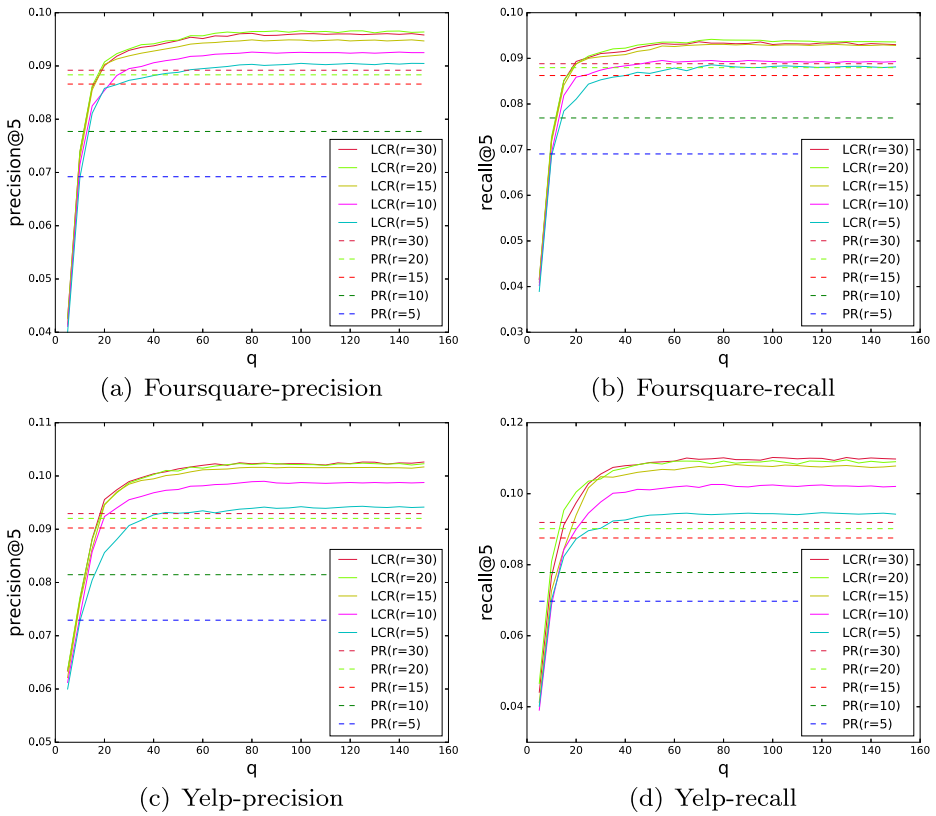
- **GeoMF** GeoMF includes geographical factor into weighted matrix factorization, then it extends the dimensions of latent vectors to learn users' spatial activity and POIs' spatial influences [12].
- **Rank-GeoFM** Rank-GeoFM is the state-of-the-art method on personalized ranking for POI recommendation [15].

### 3.4 Parameter setting

Next, we show the parameter values. We set the bandwidth parameter in Epanechnikov kernel as  $h_1 = h_2 = 0.8$ . Then we compare the performance under different number of anchor points. In Figure 5, the performance is increasing as more anchor points, and when the number is bigger than 100, the performance doesn't increase too much, however, the calculation time will consume more. Then we select 100 anchor points for Foursquare dataset and 100 anchor points for Yelp dataset. At the same time, we compare LCR with Personalized Ranking (PR, also called BPR) at different rank values, including 5,10,15,20,30. Since the performance doesn't increase too much when  $r > 15$ , we don't try any larger ranks. In Figure 5, LCR gets the best performance at  $r = 30$  on two datasets. However, from the experiment results, we observe the increase becomes smaller when the rank exceeds 15. Therefore, to balance the costing time for learning parameters and performance, we set rank as 15 in our models. In the same way, we set the rank of traditional matrix factorization as 15. In spatial favor, we set  $h = 0.05$  as fixed bandwidth for individual's frequency distribution, which is shown in Table 1. We compare the performance of MG-LCR at different values of  $\alpha$  and our improved fusing method with softmax function. In Figure 6, when  $\alpha = 0.2$ ,  $\alpha = 0.3$ , the precision@5 is best on each of two datasets. This indicates spatial similarity is indeed an important factor. And our fusing method in formula 19 can also obtain nearly best performance, it is robust and not effected by different datasets. At the same time, it doesn't need extra calculation to determine the weight parameter, so we adopt the improved fusing method in our later experiment.

### 3.5 Results and discussion

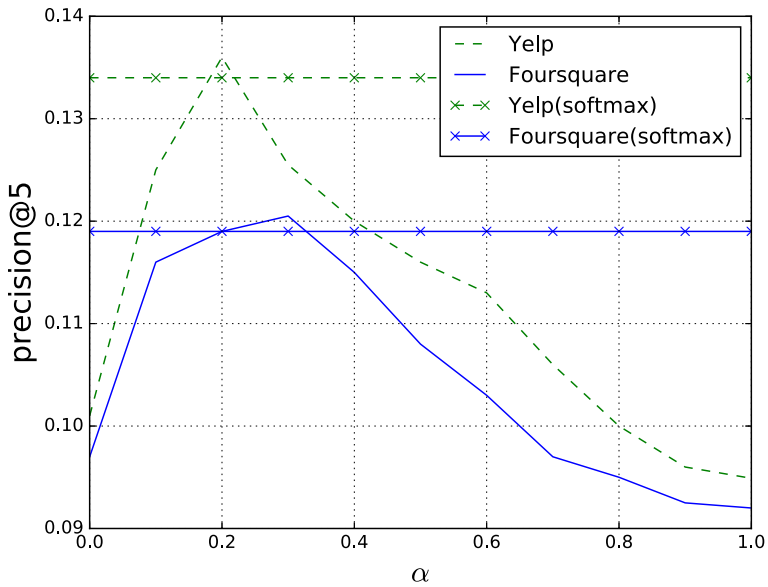
In Figure 7, we compare LCR with BPR, WMF, PFM in different number of top- $k$ . LCR is best all the time. On each number of top- $k$ , WMF is better than PFM. Even through PFM is optimized for frequency matrix, WMF is more suited to POI recommendation where data of check-ins in POI recommendation is implicit feedback. And BPR is better than WMF, PFM. Not only because it is a ranking-based factorization, but also the train set of BPR consists of positive examples, negative examples and missing values to be predicted. However, in WMF



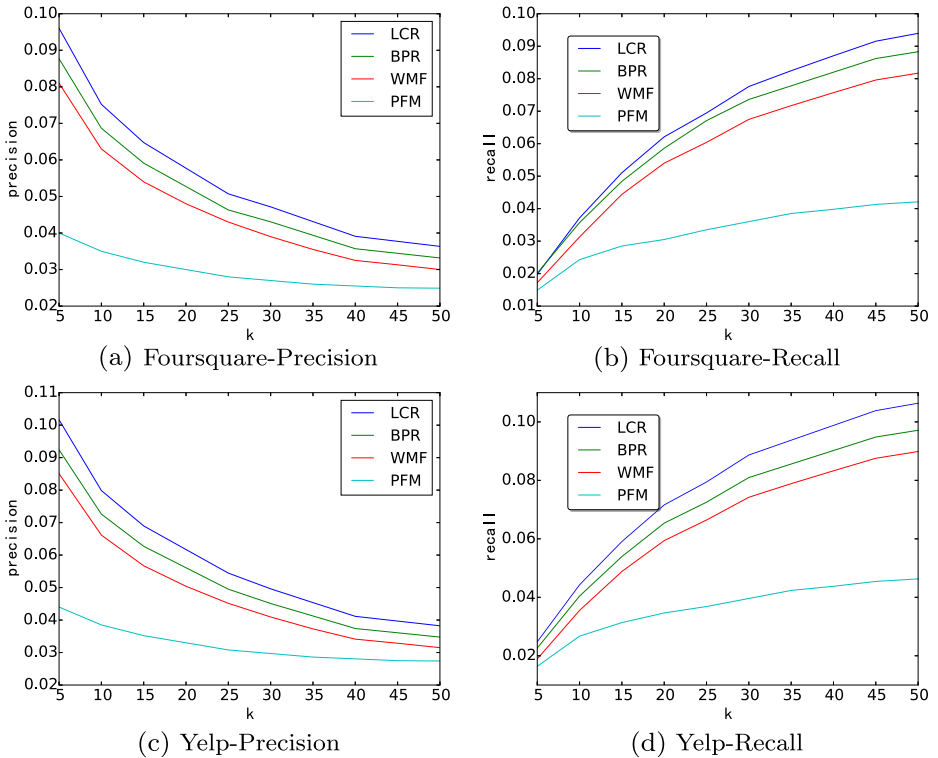
**Figure 5** Compare performance of LCR with PR which considers matrix as global low-rank, at different value of rank on two datasets, Foursquare and Yelp

and PFM, all the missing values are assumed as negative examples, this would bring some false negative examples. We can see that LCR is much more better than BPR all the time. The improvements, in terms of precision@5, are more than 5.10% and 7.20% on Foursquare and Yelp datasets, respectively. This may be caused by the local property existed in check-in matrix, and the local low-rank factorization can capture the local property exactly. This can prove the locally low-rank is much suited than globally low-rank in POI recommendation.

In Figure 8, we compare MG-LCR with LCR+fav, LCR+sim and LCR. LCR+fav is the model mixed spatial favor into local collaborative ranking, without considering spatial similarity. LCR+sim is the model mixed spatial similarity into local collaborative ranking, without considering spatial favor. MG-LCR is the model fused geographical information into local collaborative ranking, which considers not only spatial favor but also spatial similarity in fact. We can see all of MG-LCR, LCR+fav, LCR+sim are better than LCR, this proves geographical information is important in POI recommendation. Considering precision@5, in Foursquare dataset, LCR+fav and LCR+sim improve LCR by 14.9%, 4.98% respectively, in Yelp dataset they improve by 17.5%, 8.30% respectively. The experiments also show LCR+fav is a little better than LCR+sim. This can indicate spatial favor is a little more important than spatial similarity. Then we compare them with MG-LCR, which is the model mixed both spatial favor and spatial similarity into local collaborative ranking.

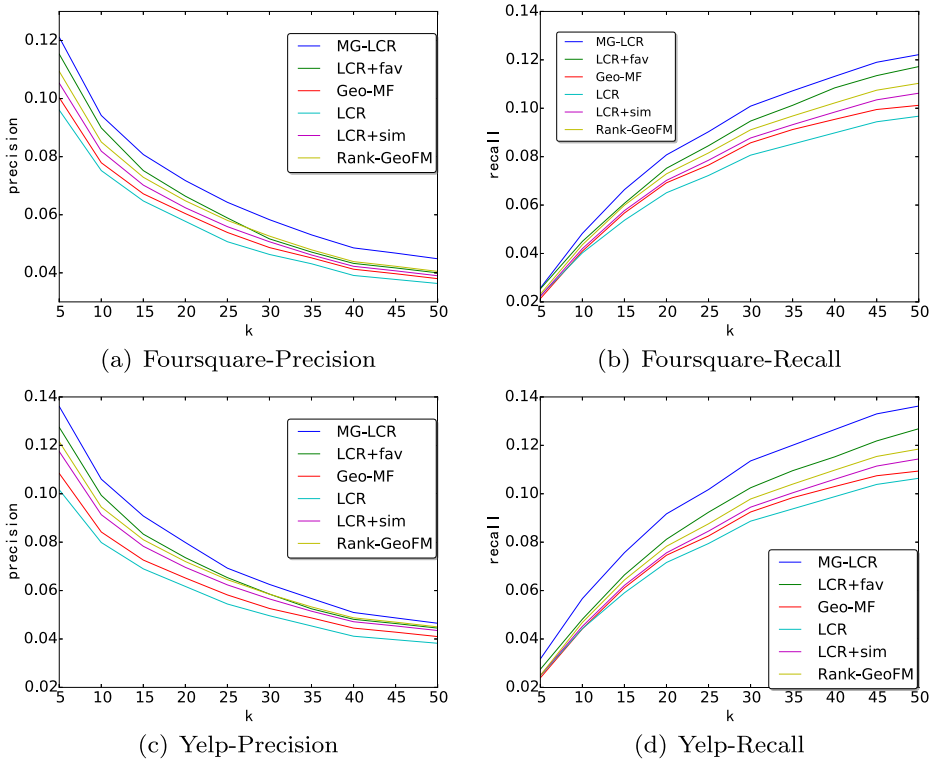


**Figure 6** Compare precision@5 at different value of  $\alpha$  and softmax method on two datasets, Foursquare and Yelp



**Figure 7** Compare the performance of LCR with others approaches on two datasets, Foursquare and Yelp





**Figure 8** Compare the performance of our proposed model MG-LCR with Rank-GeoFM and GeoMF on two datasets, Foursquare and Yelp

We can see that, considering precision@5, MG-LCR is superior to LCR+fav and LCR+sim, improving LCR by 20.6% in Foursquare, 25.4% in Yelp.

At last, we compare them with Rank-GeoFM (the state-of-art method), GeoMF. We can see that MG-LCR, LCR+fav and Rank-GeoFM are all better than GeoMF, this may be caused by that they are optimized for ranking. Further more, LCR+fav is better than Rank-GeoFM, by 4.85% and 5.48%, in terms of precision@5 on each dataset. MG-LCR is much better than Rank-GeoFM, by 11.0% and 11.9%, in terms of precision@5 on each dataset. Maybe because MG-LCR supposes the user-POI matrix is locally low-rank and includes spatial similarity to make sub-matrix more denser, which relieves the sparsity greatly. Moreover, the spatial favor of user to POI can characterise users' geographical attribute better.

### 4 Related work

POI recommendation is a very important research topic in location-based service. The researchers' works mainly focus on predicting users' ratings to locations. In these works, there are mainly two kinds of Collaborative Filtering (CF). One is memory-based CF, such as user-based CF and item-based CF, including Ye et al. [27], Zhang et al. [34] and so on. As social networks and mobility devices become popular, a large number of location-based

data can be achieved, [27, 34] got user similarity from the friendship of users, besides the check-in data, to improve the accuracy of prediction. The other is model-based CF, such as Probabilistic Factor Models (PFM) [2], non-negative matrix factorization (BNMF) [14], matrix factorization (MF) [4], weighted matrix factorization (WMF) [12].

The methods mentioned above are designed for approximating the absolute ratings of the locations. However, POI recommendation is actually optimizing for locations ranking. Traditional ranking-oriented CF algorithms directly generate a preference ordering of items for each user, including CoFiRank [23], ListRank-MF [19]. However, these ranking-oriented CF methods are based on explicit feedback, such as rating. In POI recommendation, the data we possess is implicit feedback, i.e. check-in data. Li et al. [15] used Rank-GeoFM to optimize POIs' rank, they ranked POIs for user by the frequency of check-in. On one hand, it works better for implicit feedback data, and on the other hand, the sparsity problem can be alleviated to some degree. Because both visited (positive examples) and unvisited POIs will contribute to the learning of a ranking function.

Besides the check-in data we can utilize to make POI recommendation, there are many other sources of data can be used to improve the accuracy of ranking.

Above all, the geographical information is proved to be very important in POI recommendation [27]. Ye et al. [27] argued that the geographical proximities of POIs have a significant influence on users' check-in behavior. The probability of POIs visited by user is closely related to the distance between user and POIs. They proposed that each user's check-in activities exist geographical clustering phenomenon. Then they used the power-law distribution to fit the check-in probability. Different from [27], Cheng et al. [2] captured the multi-center characteristics of individual check-in POIs and used the multi-center model for encoding the spatial clustering phenomenon. To make personalized recommendation, not model users' check-in behavior in a universal way, Zhang et al. [34] used a kernel density estimation approach to personalize the geographical influence on users' check-in behaviors. Moreover, to learn individual spatial density model, Lichman et al. [13] proposed a mixture kernel density estimation to interpolate between an individual's data and broader patterns in the population as a whole. Besides, to make use of collaborative filtering to learn users' spatial activity, Lian et al. [12] augmented users' and POIs' latent factors to users' spatial activity factors and POIs' spatial influence factors, then they used weighted matrix factorization to incorporate the clustering phenomenon into human mobility behavior for POI recommendation.

Other types of context utilized in POI recommendation include text [5, 7, 14, 20, 26], time [3, 22, 25, 31, 33, 36], friendship [34] and so on [21]. Yin et al. [28, 32] focused on solving out-of-town problem with content information. Yin et al. [29] proposed geographical locality from users' and POIs' perspectives. Even though it considers spatial similarity, it is still based on global low-rank. Besides, to complete the online recommendation, Yin et al. [30, 31] proposed a clustering-based branch and bound algorithm to prune the POI search space and facilitate fast retrieval of the top- $k$  recommendations.

Different from them, for the first time in this article we introduce local collaborative ranking for POI recommendation, which assumes user-POI matrix is locally low-rank instead of globally low-rank and is more suited for the local property existed in the check-in data. Even though they [10, 11, 16, 35] also considered the locally low-rank exists in user-item matrix, other useful information is ignored, such as geographical information. Unlike them, we introduce the geographical information systematically by the forms of spatial favor and spatial similarity, then we mix them into local collaborative ranking to improve the recommendation accuracy. From theoretical analysis, our model can also work on out-of-town

recommendation scenario. Since the estimated value of user's preference to POI is constructed by the corresponding values of the related local low-rank matrices, we can utilize spatial and content information to find more potential related local low-rank matrices, which are constructed by the out-of-town users and POIs. Therefore, our model can utilize local low-rank matrices of out-of-town to relieve data sparsity in out-of-town recommendation scenario. Besides, it also needs to do more experiments to prove the real effect, and this is a significant work we would conduct in future.

## 5 Conclusions

There are the following differences between our work and the existing works.

First, we put forward to utilize local collaborative ranking for predicting user's rating to POI, which assumes the user-POI matrix is locally low-rank instead of globally low-rank. The local collaborative ranking can relieve the sparsity of data. The experiment also reflects that local collaborative ranking is indeed superior to global low-rank factorization algorithms on POI recommendation.

Second, we analysis geographical relationship between user with POI, user with user, POI with POI systematically in POI recommendation, and mix geographical information into local collaborative ranking seamlessly. By geographical information, we model users' spatial favor to POIs (spatial favor) and spatial similarity between users (and POIs). We consider users' preference to POIs includes general favor and spatial favor. And the geographical influence to users varies greatly. We utilize personalized geographical model to get the spatial favor of user to POI. By experiment, we can see LCR+fav is much better than LCR. Besides, the similarity between users (the same to POIs) includes traditional similarity (obtained by latent factor) and spatial similarity. We make use of kernel density estimation to calculate the spatial similarity between users (and POIs). By spatial similarity, we can get more similar users (and POIs) in subgroup, and make the local matrix denser. The experiment also reflects LCR+sim is indeed better than LCR which only considers general similarity.

Third, experiment shows MG-LCR is much better than state-of-the-art method. At the same time, with good scalability, MG-LCR can incorporate more different types of context information, such as review text. By review text, we can obtain topic similarity  $ts(u, u_t)$  between users (and POIs) and find more potential similar users and POIs with anchor points, making the local sub-matrix much denser. This is a future work we will go on studying.

$$\begin{aligned}
 s_m(u, u_t) &= (1 - \alpha - \beta)s(u, u_t) + \alpha \cdot sp(u, u_t) + \beta \cdot ts(u, u_t) \\
 \alpha &= \frac{\exp(sp(u, u_t))}{Z} \\
 \beta &= \frac{\exp(ts(u, u_t))}{Z} \\
 Z &= \exp(s(u, u_t)) + \exp(sp(u, u_t)) + \exp(ts(u, u_t)). \quad (29)
 \end{aligned}$$

**Acknowledgments** This work is supported by the National Natural Science Foundation of China (61472453, U1401256, U1501252, U1611264), and This work was financially supported by 2016 Characteristic Innovation Project (Natural Science) of Education Department of Guangdong Province of China (2016KTSCX162), and Foshan Science and Technology Bureau Project (2016AG100382).

## References

1. Beutel, A., Ahmed, A., Smola, A.J.: ACCAMS Additive Co-Clustering To Approximate Matrices Succinctly[C]// Proceedings of the 24th International Conference on World Wide Web International World Wide Web Conferences Steering Committee, pp. 119–129 (2015)
2. Cheng, C., Yang, H., King, I., et al.: Fused matrix factorization with geographical and social influence in location-based social networks[C]// Twenty-Sixth AAAI Conference on Artificial Intelligence (2012)
3. Cho, E., Myers, S.A., Leskovec, J.: Friendship and mobility: User movement in location-based social networks. In: SIGKDD (2011)
4. Gao, H., Tang, J., Hu, X., et al.: Exploring temporal effects for location recommendation on location-based social networks[C]// Proceedings of the 7th ACM conference on Recommender systems, pp. 93–100. ACM (2013)
5. Gao, H., Tang, J., Hu, X., et al.: Content-Aware Point of Interest Recommendation on Location-Based Social Networks[C]// AAAI, pp. 1721–1727 (2015)
6. Gudivada, V.N.: Design and evaluation of algorithms for image retrieval by spatial similarity[J]. *Acem Trans. Inf. Syst.* **13**(2), 115–144 (1995)
7. Hu, B., Ester, M.: Spatial topic modeling in online social media for location recommendation[C]//Proceedings of the 7th ACM conference on Recommender systems, pp. 25–32. ACM (2013)
8. Hu, Y., Koren, Y., Volinsky, C.: Collaborative filtering for implicit feedback Datasets[C]// 2008 Eighth IEEE International Conference on Data Mining, pp. 263–272. IEEE Computer Society (2008)
9. Jokimki, J.: Spatial similarity of urban bird communities: a multiscale approach[J]. *J. Biogeogr.* **30**(8), 1183–1193 (2003)
10. Lee, J., Kim, S., Lebanon, G., et al.: Local low-rank matrix approximation[C]// Proceedings of The 30th International Conference on Machine Learning, pp. 82–90 (2013)
11. Lee, J., Bengio, S., Kim, S., et al.: Local collaborative ranking[C]//Proceedings of the 23rd international conference on World wide web, pp. 85–96. ACM (2014)
12. Lian, D., Zhao, C., Xie, X., et al.: GeoMF: joint geographical modeling and matrix factorization for point-of-interest recommendation[C]// Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 831–840. ACM (2014)
13. Lichman, M., Smyth, P.: Modeling human location data with mixtures of kernel densities[C]// Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 35–44. ACM (2014)
14. Liu, B., Fu, Y., Yao, Z., Xiong, H.: Learning geographical preferences for point-of-interest recommendation. In: Proceedings of KDD13, pp 1043C1051. ACM (2013)
15. Li, X., Cong, G., Li, X.L., et al.: Rank-geoFM: A Ranking based Geographical Factorization Method for Point of Interest Recommendation[C]// Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 433–442. ACM (2015)
16. Mackey, L.W., Jordan, M.I., Talwalkar, A.: Divide-and-conquer matrix factorization[C]// Advances in Neural Information Processing Systems, pp. 1134–1142 (2011)
17. Pan, R., Zhou, Y., Cao, B., et al.: One-Class Collaborative Filtering[J]. *IEEE Int Conf Data Min .icdm08.ghth* **29**(5), 502–511 (2008)
18. Rendle, S., Freudenthaler, C., Gantner, Z., et al.: BPR: Bayesian personalized ranking from implicit feedback[C]// Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence, pp. 452–461. AUAI Press (2009)
19. Shi, Y., Larson, M., Hanjalic, A.: List-wise learning to rank with matrix factorization for collaborative filtering[C]// Proceedings of the fourth ACM conference on Recommender systems, pp. 269–272. ACM (2010)
20. Wang, W., Yin, H., Chen, L., et al.: Geo-SAGE: A Geographical Sparse Additive Generative Model for Spatial Item Recommendation[C]// Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1255–1264. ACM (2015)
21. Wang, Y., Yuan, N.J., Lian, D., et al.: Regularity and Conformity: Location Prediction Using Heterogeneous Mobility Data[C]// Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1275–1284. ACM (2015)
22. Wang, W., Yin, H., Sadiq, S., et al.: SPORE: A Sequential personalized spatial item recommender system[C]// IEEE, International Conference on Data Engineering, pp. 954–965. IEEE (2016)
23. Weimer, M., Karatzoglou, A., Le, Q.V., et al.: Maximum margin matrix factorization for collaborative ranking[J]. *Advances in neural information processing systems*, pp. 1–8 (2007)

24. Wu, Y., Liu, X., Xie, M., et al.: CCCF: Improving Collaborative filtering via scalable User-Item Co-Clustering[C]// Proceedings of the ninth ACM international conference on web search and data mining, pp. 73–82. ACM (2016)
25. Xie, M., Yin, H., Xu, F., Wang, H., Chen, W., Wang, S.: Learning Graph-based POI Embedding for Location-based Recommendation[C]// ACM International on Conference on Information and Knowledge Management, pp. 15–24. ACM (2016)
26. Yang, D., Zhang, D., Yu, Z., et al.: A sentiment-enhanced personalized location recommendation system[C]//Proceedings of the 24th ACM Conference on Hypertext and Social Media, pp. 119–128. ACM (2013)
27. Ye, M., Yin, P., Lee, W.C., et al.: Exploiting geographical influence for collaborative point-of-interest recommendation[C]// Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval, pp. 325–334. ACM (2011)
28. Yin, H., Sun, Y., Cui, B., et al.: LCARS: A location-content-aware recommender system[C]// ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 221–229. ACM (2013)
29. Yin, H., Cui, B., Chen, L., et al.: Modeling Location-Based user rating profiles for personalized Recommendation[J]. *Acm Trans. Knowl. Discov. Data* **9**(3), 1–41 (2015)
30. Yin, H., Cui, B., Huang, Z., et al.: Joint Modeling of Users' Interests and Mobility Patterns for Point-of-Interest Recommendation[C]// ACM International Conference on Multimedia, pp. 819–822. ACM (2015)
31. Yin, H., Zhou, X., Shao, Y., et al.: Joint Modeling of User Check-in Behaviors for Point-of-Interest Recommendation[C]// ACM International on Conference on Information and Knowledge Management, pp. 1631–1640. ACM (2015)
32. Yin, H., Zhou, X., Cui, B., et al.: Adapting to user interest drift for POI Recommendation[J]. *IEEE Trans. Knowl. Data Eng.* **28**(10), 2566–2581 (2016)
33. Yuan, Q., Cong, G., Sun, A.: Graph-based point-of-interest recommendation with geographical and temporal influences[C]// Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management, pp. 659–668. ACM (2014)
34. Zhang, J.-D., Chow, C.-Y.: igsrl: Personalized geo-social location recommendation-a kernel density estimation approach. In: Proceedings of GIS13
35. Zhang, Y., Zhang, M., Liu, Y., et al.: Localized matrix factorization for recommendation based on matrix block diagonal forms[C]// Proceedings of the 22nd international conference on World Wide Web International World Wide Web Conferences Steering Committee, pp. 1511–1520 (2013)
36. Zhang, J.D., Chow, C.Y., Li, Y.: LORE Exploiting Sequential influence for location recommendations[C]// Proceedings of the 22nd ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, pp. 103–112. ACM (2014)
37. Zhang, J.D., GeoSoCa, C.C.Y.: Exploiting geographical, social and categorical correlations for point-of-interest recommendations[C]//Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 443–452. ACM (2015)

**Publisher's note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

## Affiliations

Wei Liu<sup>1</sup> · Hanjiang Lai<sup>1</sup> · Jing Wang<sup>2</sup> · Geyang Ke<sup>1</sup> · Weiwei Yang<sup>1</sup> · Jian Yin<sup>1</sup>

Wei Liu  
liuw56@mail2.sysu.edu.cn

Hanjiang Lai  
laihanj@gmail.com

Jing Wang  
jingyun\_wj@163.com

Geyang Ke  
kegeyang1991@gmail.com

Weiwei Yang  
yangww8@mail.sysu.edu.cn

<sup>1</sup> Guangdong Key Laboratory of Big Data Analysis and Processing, The School of Data and Computer Science, Sun Yat-Sen University, Guangzhou 510006, China

<sup>2</sup> Neusoft Institute, Guangdong 5282250, China