

GeoMF: Joint Geographical Modeling and Matrix Factorization for Point-of-Interest Recommendation

Defu Lian^{†‡}, Cong Zhao[†], Xing Xie[‡], Guangzhong Sun[†], Enhong Chen[†], Yong Rui[‡]

[†]University of Science and Technology of China

[‡]Microsoft Research, Beijing, China

[†]{liandefu, congzhao}@mail.ustc.edu.cn,

[‡]{xingx, yongrui}@microsoft.com, [†]{gzsun, cheneh}@ustc.edu.cn

ABSTRACT

Point-of-Interest (POI) recommendation has become an important means to help people discover attractive locations. However, extreme **sparsity** of user-POI matrices creates a severe challenge. To cope with this challenge, viewing mobility records on location-based social networks (LBSNs) as implicit feedback for POI recommendation, we first propose to exploit weighted matrix factorization for this task since it usually serves collaborative filtering with implicit feedback better. Besides, researchers have recently discovered a spatial clustering phenomenon in human mobility behavior on the LBSNs, i.e., individual visiting locations tend to cluster together, and also demonstrated its effectiveness in POI recommendation, thus we incorporate it into the factorization model. Particularly, we augment users' and POIs' latent factors in the factorization model with activity area vectors of users and influence area vectors of POIs, respectively. Based on such an augmented model, we not only capture the spatial clustering phenomenon in terms of two-dimensional kernel density estimation, but we also explain why the introduction of such a phenomenon into matrix factorization helps to deal with the challenge from matrix sparsity. We then evaluate the proposed algorithm on a large-scale LBSN dataset. The results indicate that weighted matrix factorization is superior to other forms of factorization models and that incorporating the spatial clustering phenomenon into matrix factorization improves recommendation performance.

Categories and Subject Descriptors

H.2.8 [Database Management]: Database Applications—*data mining*; H.3.3 [Information Storage and Retrieval]: Information Search and Retrieval—*clustering, information filtering*

Keywords

Weighted Matrix Factorization; Location Recommendation; Kernel Density Estimation; Location-based Social Network

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

KDD'14, August 24–27, 2014, New York, NY, USA.

Copyright 2014 ACM 978-1-4503-2956-9/14/08 ...\$15.00.

<http://dx.doi.org/10.1145/2623330.2623638>.

1. INTRODUCTION

With the popularity of smart mobile devices and the diversification of positioning technologies, it has become easier for people to acquire real-time information regarding their locations. This development has triggered the advent of location-based social networks (LBSNs), such as Foursquare, Jiepan, Facebook Place and so on. This emergence has not only led to location-based socializing becoming a new form of social interaction, but has also helped people speed up familiarization of the surroundings. To achieve the latter goal, Point-of-Interest (POI) recommendation has become one important means.

Recently, POI recommendation has become a popular research topic due to easy access of large-scale mobility records and inclusion of social network information. One of the most important problems in POI recommendation is how to deal with a severe challenge stemming from extreme sparsity of user-POI matrices. To this end, viewing mobility records on the LBSNs, i.e., check-ins, as implicit feedback for POI recommendation, we first propose to exploit weighted matrix factorization for this task because it usually serves collaborative filtering with implicit feedback better than the other approaches [8, 18]. In this algorithm, users and POIs are mapped onto a joint latent space by approximating a user-POI 0/1 rating matrix (each 0/1 in the matrix indicates whether a user has checked in at a POI) in a weighted way, such that a user's preference for a POI is modeled as inner product between them in that space. The reasons that check-ins can be considered as implicit feedback for POI recommendation lies in the following three aspects: first, check-in datasets just include the locations where users have been and therefore they likely prefer. In other words, they just provides *positive* examples. Second, the visit frequencies of a user to her visiting locations reflect her confidences of being fond of them. The visiting patterns of higher frequency indicate the preferences of higher confidence. Third, locations where a user has never visited are either really unattractive or undiscovered but potentially appealing. However, these two cases are usually difficult to differentiate from each other if no extra information is provided.

Fortunately, due to inclusion of geographical information of POIs, some researchers have recently discovered a spatial clustering phenomenon [21] in human mobility behavior on the LBSNs, i.e., individual visiting locations tend to cluster together, and leveraged it to assist POI recommendation. For example, Ye et al. [24] characterized it by a power law distribution on the distance between any pair of visited locations. Instead of making the power law dis-

tributed assumption, Zhang et al. [26] directly performed kernel density estimation to this distribution. To avoid the cost in computing the distance between paired locations, in [2, 12], the authors modeled the spatial clustering phenomenon in terms of geo-clustering and tried to estimate individual spatial distribution. Actually, by modeling the spatial clustering phenomenon, it becomes possible to partially distinguish unattractive locations from undiscovered but potentially appealing ones. In particular, it is much more likely that unvisited POIs near a frequently visited location are really unattractive and this likelihood depends on the visit frequency to that location. This is because if a user often visits a certain POI but has never patronized surrounding POIs, it is highly unlikely for her to visit them in the future. Therefore, weighted matrix factorization will benefit from the introduction of such a phenomenon.

For the sake of joint geographical modeling and matrix factorization, we augment users' and POIs' latent factors of the factorization model with non-negative activity area vectors of users and non-negative influence area vectors of POIs, as shown in Figure 1. These augmenting vectors consist of a set of grids with the same fixed size. Each entry in a user's activity area vector represents the possibility of this user showing up in the corresponding grid while each entry in a POI's influence vector indicates the degree of influence that this POI has on the corresponding grid. The reason for such an explicit augmentation with geographical information is that there is still no evidence showing that the latent space has included geographical information (It is easy for other information, e.g., POI categories, to be incorporated into this framework in a similar manner). In this way, a user's preference for a POI is modeled as inner product between them in the augmented space and thus includes both the interest of the user from the latent space and her preference for the location of the POI. If a user's preference for the location of a POI is non-zero, the activity areas of the user intersect with the influence areas of the POI so that this POI is reachable from the activity areas of the user. In other words, it is possible for the attractiveness of the POI to cross the gap arising from the distance.

More importantly, it is possible to exploit such a representation of users' preference to partially distinguish unattractive locations from undiscovered ones. Let's go back to the previous example, where a user often visited a POI i , but has never patronized POIs nearby. Using such a representation, the user has similar location preference for the POIs around i . However, since this user has never visited nearby POIs, from the perspective of approximation, such a representation requires inner product in the latent space to offset the location preference. Therefore, we have introduced more potentially unattractive locations (*negative* examples) into matrix factorization. Thus, it explains why the incorporation of spatial clustering phenomenon into the factorization model can help to deal with the challenge stemming from the user-POI matrix sparsity.

When only considering augmenting vectors in the proposed model and assuming that influence areas of POIs are fixed in advance, the proposed model is reduced to a non-negative weighted least square problem. Supposing that influence on a grid from a POI is determined by the density at the grid's center of the two-dimensional normal distribution centered at this POI, the reduced model encompasses two-dimensional kernel density estimation on individual visiting

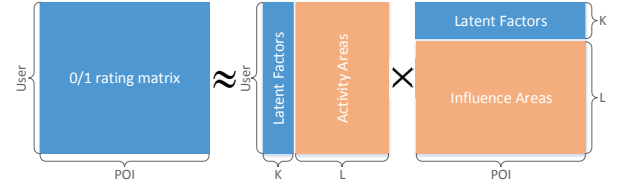


Figure 1: The augmented model for weighted matrix factorization, where the dimension of latent space is K and the number of grid areas is L .

locations. In particular, the estimated density at a location i with respect to a user equals to the averaging density at the location i of the normal distributions centered at each of her visiting locations, or inversely, it equals to the averaging density at each of her visiting locations of the normal distribution centered at the location i . When mapping POIs onto grids and constituting users' activity area vectors with grids and their visited frequency, the estimated density is in proportion to inner product between the augmenting vectors. Thus, two-dimensional kernel density estimation is a special case of the reduced model so that our model captures the spatial clustering phenomenon from the perspective of kernel density estimation (KDE). However, different from KDE, the advantage of the proposed model lies in its consistent objective goal with weighted matrix factorization. Therefore, the modeling of the spatial clustering phenomenon is seamlessly incorporated into matrix factorization.

To summarize, in this paper, we have made the following contributions to POI recommendation.

- According to the analysis to mobility records from a location-based social network for POI recommendation, these mobility records are considered as implicit feedback for recommendation systems. Therefore, we propose to leverage weighted matrix factorization to conduct POI recommendation.
- Due to the existence of accurate geographical locations of POIs, we propose augmenting users' and POIs' latent factors with activity area vectors of users and influence area vectors of POIs. Based on such an augmented model, we can not only capture the spatial clustering phenomenon from the aspect of two-dimensional kernel density estimation, but we are also able to explain why the modeling of the spatial clustering phenomenon can help to cope with the challenge of recommendation arising from the user-POI matrix sparsity. Moreover, since its objective goal is consistent with weighted matrix factorization, the modeling of the spatial clustering phenomenon is seamlessly incorporated into matrix factorization.
- We evaluate the proposed algorithm on a large-scale location-based social network dataset. The results not only show the superiority of weighted matrix factorization on POI recommendation to other forms for factorization models, but they also indicate that incorporating the spatial clustering phenomenon into matrix factorization improves recommendation performance.

2. PRELIMINARY

POI recommendation is commonly studied on a user-POI matrix $\mathbf{C} \in \mathbb{N}^{M \times N}$, where there are M users and N items.

Then each entry $c_{u,i}$ in the matrix \mathbf{C} records the visit frequency of a user u to a POI i , where u and i is reserved indexing users and POIs respectively. Here, following common symbolic notation, upper case bold letters denote matrices, lower case bold letters denote column vectors without any specification, and non-bold letters represent scalars.

2.1 Matrix Factorization

Given this frequency matrix, one possible approach to POI recommendation is low rank matrix factorization with respect to it. This involves mapping users and POIs into a joint latent space with dimension $K \ll \min(M, N)$, such that a user's preference for a POI is modeled as inner product between them in that latent space. The mapping is achieved by approximating the frequency matrix by solving the following optimization problem,

$$\min_{\mathbf{P}, \mathbf{Q}} \|\mathbf{C} - \mathbf{P}\mathbf{Q}^T\|_F^2 \quad (1)$$

where $\mathbf{P} \in \mathbb{R}^{M \times K}$ and $\mathbf{Q} \in \mathbb{R}^{N \times K}$ are users' latent vectors and POIs' latent vectors, respectively. $\|\cdot\|_F$ is the Frobenius norm of matrices, simply the square root of the sum of the squared values in matrices. And this optimization can be uniquely solved by singular value decomposition (SVD) by taking the first K pairs of singular vectors. However, due to the large number of users and POIs, it is more practical to alternate these two kinds of latent factors to solve the least square problem. For the sake of further scalability, it has become a popular solution to simply approximate the observed ratings (frequency), that is, minimizing the following weighted square errors,

$$\min_{\mathbf{P}, \mathbf{Q}} \|\mathbf{W} \odot (\mathbf{C} - \mathbf{P}\mathbf{Q}^T)\|_F^2 \quad (2)$$

where \odot is the Hadamard product operator, i.e., element-wise multiplication of matrices. \mathbf{W} is a binary weighted matrix whose entry $w_{u,i}$ indicates whether a user u has visited a POI i , i.e., $w_{u,i} = \mathbb{I}(c_{u,i} > 0)$. In order to reduce the generalization error (performance on unseen data), regularization terms are usually added to this objective goal to shrinking the latent vectors toward zero, i.e.,

$$\min_{\mathbf{P}, \mathbf{Q}} (\|\mathbf{W} \odot (\mathbf{C} - \mathbf{P}\mathbf{Q}^T)\|_F^2 + \gamma(\|\mathbf{P}\|_F^2 + \|\mathbf{Q}\|_F^2)) \quad (3)$$

2.2 Weighted Matrix Factorization for Implicit Feedback

The previously discussed matrix factorization is designed for rating predictions in a typical scenario of movie recommendation. In such cases, users not only reveal their favorite movies with high ratings, but also their least favorite movies with low ratings. However, in POI recommendation, a user's visit to a location reflects her preference for it. The visit frequency to this location determines the confident level of her preference for it. In particular, a higher visit frequency corresponds to a larger confidence of preference for the location. Therefore, in the frequency matrix, there are only *positive* examples observed so that POI recommendation is considered as a well-known One Class Collaborative Filtering (OCCF) problem [18, 8].

One solution to such an OCCF problem is to randomly sample some negative examples for each user and to assign them smaller weights than positive ones since the confidence in their negative attitude is less than the positive attitude of the positive examples. It is also possible and effective

to consider all the unvisited locations as negative examples when the weights to all negative examples are assigned the same value, i.e., 1. In particular, we set the weighting matrix as follows,

$$w_{u,i} = \begin{cases} \alpha(c_{u,i}) + 1 & \text{if } c_{u,i} > 0 \\ 1 & \text{otherwise} \end{cases} \quad (4)$$

where $\alpha(c_{u,i}) > 0$ is a monotonically increasing function with respect to $c_{u,i}$. Such a setting exactly encodes the observation that the frequency is a confidence of users' preferences. Based on this weighted matrix, the objective function for the implicit feedback is represented as follows:

$$\min_{\mathbf{P}, \mathbf{Q}} (\|\mathbf{W} \odot (\mathbf{R} - \mathbf{P}\mathbf{Q}^T)\|_F^2 + \gamma(\|\mathbf{P}\|_F^2 + \|\mathbf{Q}\|_F^2)) \quad (5)$$

It is similar to Eq (3), except that the frequency matrix \mathbf{C} is substituted with a 0/1 rating matrix \mathbf{R} , where each entry $r_{u,i} \in \{0, 1\}$ indicates whether a user u has visited a POI i . Therefore, this new form of objective actually approximates whether users have visited locations in a weighted way. However, due to the weight setting, the approximation error is summed over all entries in the user-POI matrix. Fortunately, the approximate error can be efficiently reduced via alternative least square and its time complexity for each iteration is in proportion to the total number of visited locations, i.e., the number of non-zero entries in the frequency matrix. We will provide detailed analysis in a later section.

3. JOINT GEOGRAPHICAL MODELING AND MATRIX FACTORIZATION

Weighted matrix factorization works well in most of implicit feedback datasets since it takes all unvisited locations as negative examples and assigns larger weights to positive examples than negative ones. However, in the case of POI recommendation, since POIs are naturally equipped with geographical information, there is still room for improvement in this algorithm. For example, if a user often visits a POI i but never patronizes POIs around i , it is more likely that these unvisited POIs surrounding i will become negative examples than unvisited but distant ones, and this likelihood depends on the visit frequency to the POI i . Although there are some recent studies that leverage this geographical information to assist POI recommendation [24, 2, 12, 26] by modeling the well-known spatial clustering phenomenon, these approaches are almost independent of the procedure for collaborative filtering, particularly, matrix factorization. The incorporation of geographical information into matrix factorization, in particular weighted matrix factorization for implicit feedback datasets, is important, because it may not only help to understand how to recommend POIs in the presence of geographical information, but may also help to explain why the modeling of the spatial clustering phenomenon can help to cope with the challenges of matrix sparsity. More importantly, it may boost the overall recommendation performance. To this end, we propose GeoMF for joint geographical modeling and matrix factorization.

3.1 Representation

Before presenting the GeoMF model, for the sake of convenience, we must first clarify two terms involved: user's activity areas and POI's influence areas. Roughly speaking, a user's activity areas consist of location regions where the

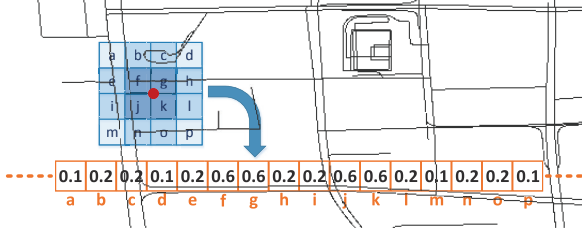


Figure 2: Generating an influence area vector for a POI (the red point).

user will show up and POI's influence areas are those regions to which the influence of this POI can be propagated. As for influence areas, we take Eiffel Tower as an example of a POI. Its influence areas may consist of all the areas in Paris since it is a famous symbol of Paris. More specifically and formally, when we assume the areas are obtained by splitting the whole world into L even grids, denoted as $\mathbb{L} = \{g_1, g_2, \dots, g_L\}$, we have the following definitions:

DEFINITION 1 (USER'S ACTIVITY AREAS). *A user's activity areas are determined by a set of pairs of a grid area $l \in \mathbb{L}$ where the user may show up and a non-negative real value $v \in \mathbb{R}_+$ indicating the possibility of appearing.*

We can represent a user's activity areas as a non-negative vector $\mathbf{x} \in \mathbb{R}_+^L$. Each entry x_l in the vector \mathbf{x} indicates the possibility that this user will appear in the grid $g_l \in \mathbb{L}$.

DEFINITION 2 (POI'S INFLUENCE AREAS). *Influence areas of a POI consist of a collection of pairs of a grid area $l \in \mathbb{L}$ to which the influence of this POI can be propagated and a non-negative real value $v \in \mathbb{R}_+$ indicating the quantity of influence from this POI.*

A POI's influence areas can also be similarly converted into a non-negative vector $\mathbf{y} \in \mathbb{R}_+^L$. Usually, influence areas are different from POI to POI. For simplicity, we assume the influence areas of a POI are fixed in advance and have a normal distribution centered at this POI. In particular, the influence y_l at a grid g_l from a POI i is $y_l = \frac{1}{\sigma} K(\frac{d(i,l)}{\sigma})$, where $K(\cdot)$ is standard normal distribution and σ is the standard deviation. Figure 2 shows an example of such a setting.

The advantage of setting the influence areas in this way is that inner product between \mathbf{x} and \mathbf{y} encompasses two-dimensional kernel density estimation on a user's visiting locations. Specifically, the estimated density of a user u at a POI i equals to $\frac{1}{|\mathbb{P}_u|\sigma} \sum_{j \in \mathbb{P}_u} K(\frac{d(i,j)}{\sigma})$, where \mathbb{P}_u is a set of visited POIs by user u . If these POIs \mathbb{P}_u are mapped into her visited areas of grids $\mathbb{L}_u \subseteq \mathbb{L}$, this estimation becomes $\frac{1}{|\mathbb{L}_u|} \sum_{l \in \mathbb{L}_u} \frac{n_l}{\sigma} K(\frac{d(i,l)}{\sigma})$, where n_l is her visiting frequency to the grid g_l , and thus is in proportion to $\mathbf{x}^T \mathbf{y}$ by setting \mathbf{x} as the visit frequency to the corresponding grid areas. In this way, we actually model the spatial clustering phenomenon using two-dimensional kernel density estimation. However, different from kernel density estimation, a user's activity area vector \mathbf{x} is a variable learned by the subsequent optimization.

We then leverage \mathbf{x} and \mathbf{y} to augment user's and POI's latent factors in the matrix factorization, respectively, as shown in Figure 1. We obtain the estimated preference ma-

trix for our proposed GeoMF model as follows

$$\tilde{\mathbf{R}} = \mathbf{P}\mathbf{Q}^T + \mathbf{X}\mathbf{Y}^T \quad (6)$$

where we stack the activity area vector of each user by row to obtain a users' activity area matrix $\mathbf{X} \in \mathbb{R}^{M \times L}$ and stack the influence area vector of each POI by row to obtain a POIs' influence area matrix $\mathbf{Y} \in \mathbb{R}^{N \times L}$. One reason for such an explicit augmentation with geographical information is that there is still no evidence showing that the latent space has already included them (It is easy for other information, e.g., the POI categories, to be incorporated into this framework in a similar way). In this way, a user's preference for a POI is modeled as an inner product in this augmented space and thus includes both the interest of the user from the latent space and her preference for the locations of the POI. If her preference for the locations of a POI is non-zero, her activity areas intersect with the influence areas of the POI so that the POI is reachable from her activity areas.

Another reason for this kind of augmentation is that it explains why the modeling of the spatial clustering phenomenon can help to deal with the challenge of matrix sparsity. In particular, returning to the example in the first paragraph of this section, where a user has often visited a POI i , but never patronized POIs around i . Using such a representation, this user has similar *non-negative* location preference for the POIs around i . However, since this user has never visited these surrounding POIs, such a representation requires the inner product in the latent space to offset the location preference, from the perspective of approximation (Minimizing weighted square loss similar to weighted matrix factorization; discussed later). In this way, we introduce more potential unattractive locations (confidently *negative* examples) into matrix factorization.

3.2 Optimization

The learning of these three matrices is achieved by minimizing a similar objective function to weighted matrix factorization for implicit feedback, i.e.,

$$\begin{aligned} \min_{\mathbf{P}, \mathbf{Q}, \mathbf{X}} \quad & \|\mathbf{W} \odot (\mathbf{R} - \mathbf{P}\mathbf{Q}^T - \mathbf{X}\mathbf{Y}^T)\|_F^2 + \gamma(\|\mathbf{P}\|_F^2 + \|\mathbf{Q}\|_F^2) + \lambda \|\mathbf{X}\|_1 \\ \text{subject to } & \mathbf{X} \geq 0 \end{aligned} \quad (7)$$

where $\|\mathbf{X}\|_1$ is ℓ_1 norm of matrix \mathbf{X} . One reason for imposing such a matrix norm is that users are usually constrained around several long-stay locations, such as, home or workplace, while ℓ_1 norm of the matrix is generally exploited to restrict the sparsity of matrices [16]. Another benefit of placing the sparse constraint is that it can also improve the effectiveness and the efficiency of recommendation (shown in experimental section).

This objective function is minimized with an alternative optimization, consisting of one procedure of learning latent factors when fixing \mathbf{X} and another one of non-negative weighted sparse least square with respect to \mathbf{X} when fixing latent factors. Since in each procedure, the objective function is not increasing, the iteration of such an alternative optimization can guarantee the non-increase of the objective function.

When fixing activity area matrix \mathbf{X} , the optimization of this objective function with respect to the users' and POIs' latent factors is similar to alternative least square in weighted matrix factorization discussed previously. More specifically,

the latent factors of user u is updated based on

$$\mathbf{p}_u = (\mathbf{Q}^T \mathbf{W}^u \mathbf{Q} + \gamma \mathbf{I})^{-1} \mathbf{Q}^T \mathbf{W}^u (\mathbf{r}_u - \mathbf{Y} \mathbf{x}_u) \quad (8)$$

where \mathbf{W}^u is an $N \times N$ diagonal matrix, subject to $W_{i,i}^u = w_{u,i}$ and \mathbf{r}_u is a column rating vector of user u . Here, since we have set the same weight, i.e., 1 to the unvisited locations, there is a trick to speed up its calculation [8]. In particular, $\mathbf{Q}^T \mathbf{W}^u \mathbf{Q} = \mathbf{Q}^T (\mathbf{W}^u - \mathbf{I}) \mathbf{Q} + \mathbf{Q}^T \mathbf{Q}$. In this case, the second part is independent of users so that it can be precomputed while the first part only requires $O(\|\mathbf{r}_u\|_0 K^2)$, where ℓ_0 norm of matrix (vector) is the number of non-zero entries in this matrix (here, it is also the number of visited locations of user u). For the inverse of a $K \times K$ matrix, we assume it requires $O(K^3)$ time even though more efficient algorithms exist but probably are less relevant for the typically small values of K . Applying the similar trick to calculate the remaining part $\mathbf{Q}^T \mathbf{W}^u \hat{\mathbf{r}}_u$, where $\hat{\mathbf{r}}_u = (\mathbf{r}_u - \mathbf{Y} \mathbf{x}_u)$, it costs $O(\|\hat{\mathbf{r}}_u\|_0 K)$ to get a $K \times \|\hat{\mathbf{r}}_u\|_0$ matrix. Completing the final matrix multiplication between the inversed matrix and the intermediate matrix requires $O(\|\hat{\mathbf{r}}_u\|_0 K^2)$ time. Therefore, it will cost $O(\|\hat{\mathbf{r}}_u\|_0 K^2 + K^3)$ to update latent factors for user u in total. If we update all users' latent factors in sequence (parallel updating is possible since there is no dependence between users), the total update time is $O(\|\hat{\mathbf{R}}\|_0 K^2 + MK^3)$, where $\hat{\mathbf{R}} = \mathbf{R} - \mathbf{X} \mathbf{Y}^T$.

Similarly, we can update the latent factor of a POI i as:

$$\mathbf{q}_i = (\mathbf{P}^T \mathbf{W}^i \mathbf{P} + \gamma \mathbf{I})^{-1} \mathbf{P}^T \mathbf{W}^i (\mathbf{r}_i - \mathbf{X} \mathbf{y}_i) \quad (9)$$

where \mathbf{W}^i is an $M \times M$ diagonal matrix, subject to $W_{u,u}^i = w_{u,i}$ and \mathbf{r}_i is a column rating vector of the POI i . Applying the similar optimization trick, we can complete the update of POIs' latent factors in $O(\|\hat{\mathbf{R}}\|_0 K^2 + NK^3)$ time.

Since $\|\hat{\mathbf{R}}\|_0 > (M + N) \times K$ is usually satisfied, the total complexity of updating latent factors in one iteration is $O(\|\hat{\mathbf{R}}\|_0 K^2)$. K is set as a small number ($50 \sim 200$) so that it can be considered as proportional to the number of non-zero entries in new matrix $\hat{\mathbf{R}}$, but larger than the original one \mathbf{R} . Therefore, the sparsity constraint of \mathbf{X} is also important to the efficiency of updating these latent factors. Additionally, for the sake of higher efficiency, we assume that the two-dimensional normal distribution for generating influence areas of POIs is truncated so that the sparsity structure of \mathbf{Y} can also be guaranteed. In other words, only those areas within a certain threshold of distance (i.e., d km) from a POI are considered as its influence areas. This is reasonable to some extent since the normal distribution usually decays quickly with the increase of the distance from its center.

According to previous analysis, the updating of latent factors corresponds to approximating the new matrix $\hat{\mathbf{R}} = \mathbf{R} - \mathbf{X} \mathbf{Y}^T$. Therefore, by introducing the geographical information, we have added many negative examples for the latent factor model. These negative examples in this new matrix $\hat{\mathbf{R}}$ have negative ratings due to the non-negativeness of \mathbf{X} and \mathbf{Y} , and their absolute values are in proportion to a user's preference for the locations of POIs. Thus it is more likely that unvisited POIs near frequently visited POIs will become negative examples. This confirms the preceding explanation why incorporating the spatial clustering phenomenon into collaborative filtering can help to cope with the challenge of matrix sparsity.

Now let's turn to learning users' activity area matrix \mathbf{X} . When fixing the latent factors, the objective function with

respect to \mathbf{X} is similar to a non-negative weighted least square problem, which can be further generalized as a bounded-variable least square problem [10]. Such kinds of problems have been solved by several approaches, including active set method [10], sequential coordinate-wise algorithm [4], projected gradient descent method [11] and so on. Among these methods, projected gradient descent is highly efficient and has been extensively studied in non-negative matrix factorization, which can also be cast into two sub-problems related to non-negative least square [11]. The general idea of the projected gradient descent algorithm is to update parameters by gradient descent and then to project the updated ones into feasible regions defined by bound constraints. Nevertheless, the choice of learning rate in the gradient descent needs to guarantee that the projected parameters can sufficiently decrease the objective function. Thus we leverage the methods proposed in [11] to update users' activity area matrix. However, due to the existence of the weighting matrix and latent factors, the gradient of this objective function with respect to \mathbf{X} is a full matrix. It is impractical to update all the parameters at one time. Thus, instead, we update the activity area vectors for each user independently.

Let's rewrite the objective function with respect to the activity area vector of a user u and discard the irrelevant terms.

$$L(\mathbf{x}_u) = \|\mathbf{W}^u (\mathbf{r}_u - \mathbf{Q} \mathbf{p}_u - \mathbf{Y} \mathbf{x}_u)\|_F^2 + \lambda \|\mathbf{x}_u\|_1 \quad (10)$$

subject to $\mathbf{x}_u \geq 0$

The gradient of $L(\mathbf{x}_u)$ with respect to \mathbf{x}_u is

$$\nabla L(\mathbf{x}_u) = \mathbf{Y}^T \mathbf{W}^u (\mathbf{Y} \mathbf{x}_u - (\mathbf{r}_u - \mathbf{Q} \mathbf{p}_u)) + \lambda \quad (11)$$

Based on this gradient, we update \mathbf{x}_u as follows

$$\mathbf{x}_u^{(t+1)} = P_+(\mathbf{x}_u^{(t)} - \alpha \nabla L(\mathbf{x}_u)) \quad (12)$$

where $P_+(\mathbf{x})$ is a function to project a vector $\mathbf{x} \in \mathbb{R}^L$ onto its non-negative orthant \mathbb{R}_+^L . In particular,

$$P_+(x_l) = \begin{cases} x_l & \text{if } x_l > 0 \\ 0 & \text{otherwise} \end{cases}, \quad l \in \{1, \dots, L\} \quad (13)$$

The learning rate, α is chosen so as to ensure the sufficient decrease of the objective function, i.e.,

$$L(\mathbf{x}_u^{(t+1)}) - L(\mathbf{x}_u^{(t)}) \leq \varepsilon \nabla L(\mathbf{x}_u)^T (\mathbf{x}_u^{(t+1)} - \mathbf{x}_u^{(t)}) \quad (14)$$

where ε is a parameter of this condition and commonly set as 0.01. Since our objective is a quadratic function w.r.t \mathbf{x}_u , this condition can be quickly evaluated via the gradient and Hessian matrix ($\nabla^2 L(\mathbf{x}_u) = \mathbf{Y}^T \mathbf{W}^u \mathbf{Y}$), i.e.,

$$(1 - \varepsilon) \nabla L(\mathbf{x}_u)^T \Delta \mathbf{x}_u + \frac{1}{2} \Delta \mathbf{x}_u^T \nabla^2 L(\mathbf{x}_u) \Delta \mathbf{x}_u \leq 0 \quad (15)$$

where $\Delta \mathbf{x}_u = \mathbf{x}_u^{(t+1)} - \mathbf{x}_u^{(t)}$ is the change of \mathbf{x}_u . In this case, in each step, although the objective function has decreased sufficiently, it requires repeatedly searching the learning rate (based on some heuristic rules). And with the change in learning rate, the gradient increases by $\nabla^2 L(\mathbf{x}_u) \Delta \mathbf{x}_u$. Since there are approximately an equal number of influence areas (denoted as $n_{\hat{i}}$) for different POIs, the increase of the gradient along with evaluating the sufficient decrease condition costs at most $O(\|\mathbf{x}_u\|_0 n_{\hat{i}})$. The part that is irrelevant to \mathbf{x}_u in the gradient can be computed in a way that is similar to updating the latent factors and its complexity is

denominated by the calculation of the Hessian matrix, that is, $O(\|\mathbf{r}_u\|_0 n_i^2)$. If we perform an updating operation for each user in sequence (it can be done in parallel), the overall complexity is $O(\#iter \times t \|\mathbf{X}\|_0 n_i + \|\mathbf{R}\|_0 n_i^2)$, where $\#iter$ is the number of iterations to be updated and t is the average number of trials for searching the learning rate. Here, we once again observe the importance of the sparse structure in the efficiency of optimization.

Someone may be curious about what a user's activity areas will be based on such an updating strategy. Thus we will try to conduct an analysis from this perspective. For simplicity, assume that the latent factors are not taken into account. The users' activity area matrix \mathbf{X} is initialized to zero. After the first iteration, $\mathbf{x}_u = \alpha P_+(\mathbf{Y}^T \mathbf{W}^u \mathbf{r}_u - \lambda)$, thus including the regions that can be directly reached from a user's visiting POIs by means of \mathbf{Y} . The possibility of showing up in a grid area depends on the visiting frequency via the weighting matrix \mathbf{W}^u . Thus, the update in this first iteration is similar to kernel density estimation except that it is subject to the decrease of the objective function. In the subsequent iterations, a user's activity areas are expanded by $\mathbf{Y}^T \mathbf{W}^u \mathbf{Y}$, which encodes the personalized spatial correlation between POIs, under the condition of decreasing the objective function.

4. EXPERIMENT

4.1 Dataset and Experimental setup

We evaluate the proposed algorithm on a large-scale location-based social network dataset. This dataset was crawled from Jiebang, a Chinese location-based social network, similar to Foursquare and spanned almost two years from March, 2011 to March, 2013. Although check-ins of Jiebang users are not publicly available on Jiebang itself, they are synchronized to other social networks, i.e., Weibo, so that they can be obtained by the open APIs of these social networks. In this way, we crawled 36,143,085 Jiebang check-ins at 1,000,457 POIs from 454,375 users. If this data is represented as a user-POI matrix, its density is 7.95×10^{-5} . In this dataset, each user made 80 check-ins on average and these check-ins are dispersed at 47 POIs on average. If we distribute these check-ins by their date, we find that each user only make 1.5 check-ins each day on average. If we distribute these POIs into 3 km^2 regions, each region contains 13 POIs on average and up to 13,068 POIs in the maximal case. Then we select POIs that are visited by at least two users and users who have been to at least 10 distinct locations. Finally, 276,450 users and 574,095 POIs are then reserved and the density of these users on these POIs is 1.27×10^{-4} .

For each user, we randomly select 30% of her visiting locations as ground truth for testing. The remaining portions from each user constitute a training dataset for learning the parameters of the proposed model. The learned model scores each unvisited location for each user (i.e., not in her training portion) and then ranks them according to the scores. The learned model is then assessed by its capacity of finding the ground truth locations for each user among the ranked locations. Such a capacity is usually measured by two widely-used metrics, i.e., Recall@k and Precision@k, in the top k POI recommendation [24, 12, 5]. The former metric w.r.t each user indicates what percentage of her visiting locations can emerge in the top k recommended POIs while the latter metric w.r.t each user indicates what percentage of locations

among the top k recommended POIs has been visited by her. Formally, if we define $\mathbb{S}_u(k)$ as the top k recommended POIs and \mathbb{V}_u as the visited locations of user u ,

$$\begin{aligned} \text{Recall@}k &= \frac{1}{M} \sum_{u=1}^M \frac{|\mathbb{S}_u(k) \cap \mathbb{V}_u|}{|\mathbb{V}_u|} \\ \text{Precision@}k &= \frac{1}{M} \sum_{u=1}^M \frac{|\mathbb{S}_u(k) \cap \mathbb{V}_u|}{k}. \end{aligned} \quad (16)$$

The above procedure corresponds to one trial. The final performance of the recommendation algorithms is obtained by averaging on the metrics of 5 times independent trials.

Next we set up the parameters controlling the complexity and being not learned via optimization. All these parameters are set by cross validation. The dimension of the latent space is set as 50 and the weighted matrix is set by $\alpha(c_{u,i}) = 1 + \log(1 + c_{u,i} \times 10^\epsilon)$ where $\epsilon = 10$. However, the recommendation performance is not sensitive to the regularization coefficient γ for shrinking latent factors since all negative examples (unvisited locations) are taken into account and thus we set it as $\gamma = 0.01$. For the parameters related to modeling locations, including the ℓ_1 regularization coefficient λ for the sparsity constraint and the distance d for truncated normal distribution for setting the influence area matrix, we examine them in the result part.

In the following sections, we will evaluate the proposed algorithm according to three aspects. First, we compare the different approaches to using matrix factorization for POI recommendation. Then we study the modeling of spatial clustering phenomenon, denoted as *GeoWLS* (which just ignores the latent factors in *GeoMF*), under the different parameter settings and compare it with kernel density estimation. Finally, we compare the proposed model (*GeoMF*) with weighted matrix factorization (*WMF*) and *GeoWLS* in order to understand the benefit of geographical information.

4.2 Study of Matrix Factorization

For the sake of understanding the effectiveness of WMF for POI recommendation, we compare it with user-based collaborative filtering (*UCF*) [24] and the following factorization algorithms.

- **UCF**. In this *UCF*, the similarity between users is related to the number of their common visited locations and the weight of a user w.r.t a location is 0/1 indicating whether the user has visited the location. In other words, *UCF* is built based on the user-POI 0/1 rating matrix \mathbf{R} .
- **MF-01**. A non weighted version of WMF, performs factorization on user-POI 0/1 rating matrix \mathbf{R} , based on the following optimization $\min_{\mathbf{P}, \mathbf{Q}} \|\mathbf{R} - \mathbf{P}\mathbf{Q}^T\|_F^2 + \gamma(\|\mathbf{P}\|_F^2 + \|\mathbf{Q}\|_F^2)$, where γ is set by cross validation.
- **MF-Freq**. A factorization algorithm is directly conducted on user-POI frequency matrix \mathbf{C} , denoted as *MF-Freq*, based on a similar optimization procedure to *MF-01* except for substituting \mathbf{R} with \mathbf{C} . Such a factorization algorithm has been exploited in [17, 5] for POI recommendation
- **B-NMF**. B-NMF is a Bayesian non-negative matrix factorization algorithm that places Gamma prior on

its latent factors. It tries to optimize the following objective function [15],

$$\min_{\mathbf{P}, \mathbf{Q}} D(\mathbf{C} \| \mathbf{P}\mathbf{Q}^T) - \sum_{u,k} \ln \Gamma(p_{u,k} | \alpha_k, \beta_k) - \sum_{i,k} \ln \Gamma(q_{u,k} | \alpha_k, \beta_k)$$

where $D(\mathbf{A} \| \mathbf{B}) = \sum_{i,j} A_{i,j} \log \frac{A_{i,j}}{B_{i,j}} - A_{i,j} + B_{i,j}$ is the divergence from matrix \mathbf{A} to \mathbf{B} [20] and $\Gamma(x | \alpha, \beta)$ is a Gamma distribution, parametrized by a share parameter α and a rate parameter β . It should be noted that we can not optimize it over all entries in the matrix for the sake of efficiency. Thus we randomly sample the same number of negative POIs as positive ones. This algorithm has been exploited in [2, 12] for POI recommendation.

- **WMF-B.** This algorithm introduces the popularity of POIs into weighted matrix factorization for POI recommendation since it is common to retain items' biases in the factorization models [9]. This algorithm minimize $\|\mathbf{W} \odot (\mathbf{R} - \mathbf{P}\mathbf{Q}^T - \mathbf{1}\mathbf{b}^T)\|_F^2 + \gamma(\|\mathbf{P}\|_F^2 + \|\mathbf{Q}\|_F^2) + \eta\|\mathbf{b}\|_2^2$ with respect to users' and items' latent factors \mathbf{P} and \mathbf{Q} as well as the items' biases $\mathbf{b} \in \mathbb{R}^N$. Here, $\mathbf{1}$ is a column vector of length M with all entries one.

The result of the comparison is shown in Figures 3(a) and 3(d). The **first** observation is that factorization on frequency matrix (*MF-Freq*) based on square loss function performs worst among all the factorization models (even if we apply logarithm to the frequency). However, if we make use of divergence instead of square loss, the performance (i.e., *B-NMF*) improves compared to *MF-Freq* since it can model the skewness of visit frequency [2, 12]. This constitutes our **second** observation. Unfortunately, according to the **third** observation, it doesn't perform as well as the factorization on 0/1 rating matrix (*MF-01*) and user-based collaborative filtering (*UCF*). This indicates that approximating frequency, in spite of terms of Poisson likelihood in *B-NMF*, will attenuate the capacity of factorization models for recommendation. Additionally, in the **next** observation, *MF-01* is superior to both *B-NMF* and *UCF*. This may arise from the sparsity of individual visited locations because *B-NMF* only optimizes over a small subset of all the entries in the matrix for the sake of efficiency and *UCF* is modeled on the observed individual visited locations, while *MF-01* considers all unvisited locations as negative examples efficiently. The superiority of *MF-01* to *UCF* also confirms the advantage of matrix factorization models in recommendation systems compared to memory-based collaborative filtering [1]. When using the visiting frequency as the confidence of a user's preference for locations, we can further improve the recommendation performance, based on the **fifth** observation, that *WMF* outperforms *MF-01*. In summary, weighted matrix factorization works better for collaborative filtering on implicit feedback than the other forms of matrix factorization and user-based collaborative filtering. The **final** observation, which we make by comparing the results of *WMF* with *WMF-B*, is that we can't observe a significantly large improvement due to the introduction of the POIs' biases. One reason for this is that weighted matrix factorization may already imply such bias information. Also, the common reason to introduce the bias in the traditional matrix factorization algorithm is to center user-item rating matrix, but in weighted matrix factorization, the user-item rating matrix

is filled with 0/1 entries, indicating that users has a certain action on items, so that there is no special requirement for centering. In other words, it is not essential to include the items' biases when applying weighted matrix factorization for collaborative filtering with implicit feedback.

4.3 Study of Modeling Spatial Clustering Phenomenon

By ignoring the latent factors in the GeoMF model, the reduced model (GeoWLS) tries to capture the spatial clustering phenomenon. In this model, the sparsity structure of the activity area matrix of users and the influence area matrix of POIs are important to the effectiveness and efficiency of GeoMF. These two parts are captured by two separated parameters, i.e., the regularization coefficient λ of ℓ_1 matrix norm and d km of truncated normal distribution for generating POIs' influence areas. Thus, in this section, we examine the effect of these two parameters and show the results in Figure 3(b) and 3(e). We can make the following observations:

First, we compare the recommendation performance under different distance thresholds for truncated normal distribution. In particular, we take grid regions within 0.5km and 1km of POIs to generate the influence area matrix of POIs while setting λ to zero, i.e., without any sparsity constraints, and then train two GeoWLS models based on such settings. According to the recommendation performance of these two models, we only observe a small improvement after taking more grid regions into account. One major reason lies in the high density of POIs. For such a sake, we analyze the density of POIs in our check-in dataset, which we measure via the number of POIs reached by expanding visited POIs to their nearby locations. In particular, if we expand visited POIs to nearby locations within d km distance, we would like to know how many unvisited locations can be added to the candidates for recommendation. Thus we draw a scatter plot between them, shown in Figure 4(a), where the x axis indicates the expanded distance while the y axis represents the number of candidate POIs (Since this value is different from person to person, we plot the (25%,50%,75%) quantile of its distribution over the population). We can see that when we expand visited POIs to nearby locations within 0.5 km, the number of candidate locations will be around 8,000 among half users and when expanded to 1 km, this number increases to 11,000. Therefore, expanding to 0.5 km may already include sufficient candidates for top-k recommendation ($k \ll 8,000$). Moreover, with the help of this analysis, we can see that it is really difficult to recommend POIs based only on location information. Therefore, the recommendation performance that only uses locations is lower than collaborative filtering approaches.

Second, by fixing the distance to 0.5 km but varying λ , we explore the effect of sparsity constraints of users' activity area matrix on recommendation performance. According to these two figures, increasing from 0 to 10, the performance of GeoWLS improves consistently. In other words, the sparsity structure of users' activity area matrix can help improve the recommendation performance because this constraint reduces some of the noisy activity areas. Moreover, the sparsity constraint reduces the number of non-zero entries in the matrix and increases the efficiency of our GeoMF model. However, when continuing to increase its value from 10 to 20, the performance only improves in the top positions

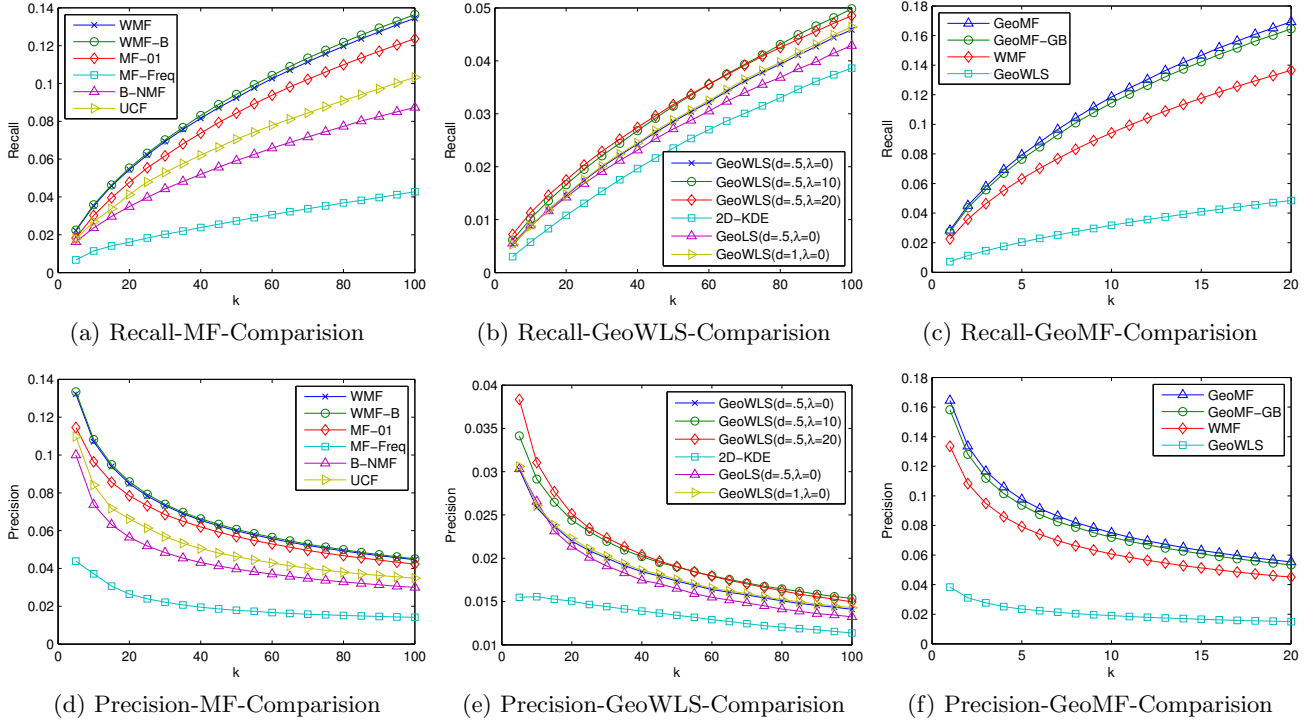


Figure 3: Figures (a) and (d) compare the performance of different kinds of matrix factorization for POI recommendation. Figures (b) and (e) compare the performances of GeoWLS with different settings of the coefficients of sparse constraints and with two-dimensional kernel density estimation. Figures (c) and (f) compare GeoMF with two components (WMF and GeoWLS).

while those at the bottom start to decrease. The underlying reason is that stronger sparsity constraints also remove some significant activity areas at the same time of reducing noise.

It is possible to suspend the necessity of weighted square loss instead of square loss, since this procedure may be independent of matrix factorization. To this end, we compare GeoWLS with a non-weighted version *GeoLS* under the same setting of parameters in the **third** step. From the results, we can see that GeoWLS outperforms GeoLS. Thus the goal of geographical modeling is consistent with weighted matrix factorization w.r.t POI recommendation so that geographical modeling is seamlessly incorporated into matrix factorization for recommending POIs.

Although GeoWLS encompasses two-dimensional kernel density estimation (*2D-KDE*), it is subject to the minimization of weighted square loss and is different from 2D-KDE. Hence, **finally**, we compare GeoWLS with 2D-KDE. Based on the results, we observe that GeoWLS is much better than 2D-KDE and that the formalization in terms of weighted non-negative least square is effective for POI recommendation. Moreover, we also study the reasonableness of GeoWLS by plotting the activity areas of a sample user in Figure 4(b). The results show that it is similar to 2D-KDE, that is, in the areas with larger number of visited locations, users have a higher possibility of showing up. However, the difference is that in GeoWLS there will be a larger possibility of showing up in less popular areas given two areas with the same visit frequency (these two regions are indistinguishable via 2D-KDE). This is reasonable, considering that a visit to less popular areas implies a larger preference for it compared to popular ones.

4.4 Comparison with Baselines

In this section, we will compare GeoMF with its two components, i.e., WMF and GeoWLS in order to determine the benefits of introducing geographical information into WMF. Additionally, we use Gradient Boosting [6] to integrate these two components, denoted as *GeoMF-GB*, since our GeoMF model can be considered an additive model of WMF and GeoWLS. It is similar to one round of GeoMF optimization (it alternates between GeoWLS and WMF three times in our experiments) starting with GeoWLS. However, we can set different weighting strategies for GeoWLS and WMF since these two models are independent of each other in the Gradient Boosting framework. We omit the implementation here due to space limitation. We simply show the results in Figure 3(c) and 3(f). We **first** observed that GeoMF greatly outperforms WMF, indicating that GeoMF benefits from the knowledge of geographical information. In other words, latent vectors in the WMF may not capture the geographical information so that its explicit modeling in GeoMF can remedy this weakness. **Second**, GeoMF is slightly better than GeoMF-GB. This, on the one hand, indicates GeoMF can benefit from multiple alternation between GeoWLS and WMF; on the other hand, it implies the information obtained by the negative examples from GeoWLS is limited with respect to WMF. One potential reason is that all the unvisited locations are already considered as negative examples in WMF. Another possible reason is that these two models are relatively independent of each other so that multiple alternation can not boost the recommendation performance a great deal.

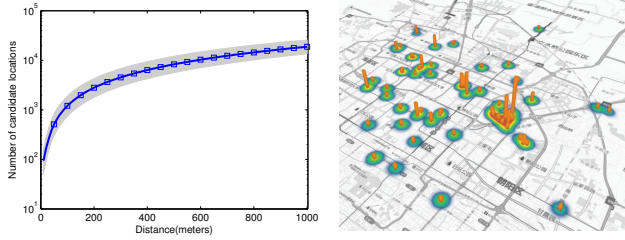


Figure 4: (a) Number of candidate locations vs. the expansion distance (b) The density plot of activity areas of one user

4.5 Discussion and Future Work

Our proposed GeoMF model is a general framework for using the content information of POIs in matrix factorization. POIs also include category attributes and these categories are organized by two-level hierarchy. The top level only contains several coarse categories while the bottom ones include finer categories. Therefore we also try to incorporate finer categories into the factorization models. However, our results show no significant improvement to the factorization models. One reason is that POIs’ latent factors may already encode such information. Another one lies in the incompleteness of POIs’ categories. According to a statistics to the categories of POIs, we find that over 40% of POIs don’t have finer categories. Nevertheless, one interesting thing is that the visit time of users’ to locations may help us to deduce the category information according to [23]. Thus, one future task is to exploit the time information in matrix factorization for POI recommendation.

Another assumption for the GeoMF model is that normal distribution would be truncated with the same influence distance of all POIs. However, in practice, different POIs may have different influence distances. In other words, some locations (i.e., restaurants) are only visited by nearby users while others (i.e., hotels, attractions) may be visited by many non-natives. Thus, it is interesting to learn the influence distance of POIs and to study the correlation between it and POIs’ categories.

Besides, social network information is not taken into consideration in our model. This is because our model focuses on incorporating location into matrix factorization. However, our model can be easily extended with social interaction, e.g., by imposing a Graph Laplacian regularization. Due to space limitation, we will discuss it in the future work.

5. RELATED WORK

Location recommendation has been an important topic in location-based services. For example, some research has focused on recommending some specific types of locations. Park et al. [19] designed a system based on Bayesian learning with both users’ preferences and location contexts to recommend restaurants. Similarly, Horozov et al. [7] developed a user-based collaborative filtering system to recommend restaurants to a user, by finding which restaurants similar users have visited before. Zheng et al. [28] designed a random walk style model to do tourism hot spot recommendation by taking into account both users’ travel experiences and location attractiveness. In addition to single-type location recommendation mentioned above, there is also some other work considering multiple-activity-type lo-

cation recommendation. For example, Zheng et al. considered location recommendation and activity recommendation together, so that they can provide location recommendation w.r.t. different types of activities [27]. The proposed model formulates a location-activity matrix for collaborative filtering and uses some additional information such as location features to help recommendation.

With the growing popularity of location-based social networks, location recommendation is drawing plenty of attention once again, since it is possible to obtain large-scale location visit records and several new challenges, including an extremely sparse user-location matrix and the presence of social networks, have arisen from this data. To address these challenges, several methods have been proposed. For example, Ye et al. [24] discovered the spatial clustering phenomenon of individual visited locations and characterized it by a power law distributed distance of any pair of visited locations [24]. In addition, they also exploited the similarity between users based on location history and social relationships on social networks for collaborative filtering. To better incorporate social relationships from social networks into collaborative filtering, Noulas et al. [17] conducted random walk with a restart on user-POI bipartite graph and social graph. With regard to modeling the spatial clustering phenomenon, instead of making the power law distribution assumption, Zhang et al. [26] suggested using kernel density estimation to estimate the distribution of distance between pairs of locations. Concentrating on modeling the distance distribution may ignore the multi-center characteristics of individual visiting locations according to [2]. Thus, the authors tried to apply clustering techniques on individual visited locations for encoding the spatial clustering phenomenon. They also exploited Bayesian non-negative matrix factorization for location recommendation, placing a Gamma prior on non-negative latent factors since this model can capture the skewness of the visit frequency to locations. These two models are then multiplied together since both of them are modeled in a probabilistic way. To improve the ad hoc integration between them, Liu et al. [12] proposed a geographical probabilistic factor analysis framework to takes geo-clustering and Bayesian non-negative matrix factorization into consideration by defining a user’s preference for locations as a multiplication of her interest in the locations, the locations’ popularity and the distance between her and POIs.

In addition to studying the effect of social network information and of spatial clustering phenomenon, there has also been research into studying the impact of context information, e.g., time, and the textual content of locations on POI recommendation. For example, in [13, 25, 22], they tried to leverage content information of locations via topic modeling to assist POI recommendation; In [5], Gao et al. proposed distinguishing a user’s latent factors at different times and exploiting several strategies to aggregate a user’s time-dependent latent factors; In [3, 14], the authors leveraged the information from previous POIs, including the POIs themselves, categories and so on, for next POI recommendation.

Comparing our work with these existing ones, there are many differences. First, we leverage weighted matrix factorization for POI recommendation since according to our experimental results it may be more appropriate than other methods for collaborative filtering from implicit feedback. Second, the modeling of the spatial clustering phenomenon

is similar to two-dimensional kernel density estimation and thus doesn't make any assumption about the distribution of visited locations. Finally, geographical modeling is seamlessly incorporated into weighted matrix factorization and this incorporation explains why modeling the spatial clustering phenomenon helps to deal with the challenge of matrix sparsity. However, we don't take the content information into consideration in our proposed model since we don't have any other information except the categories of POIs in our dataset. This information can easily be incorporated into the current framework. Actually, we have tried to incorporate the categories of POIs, but haven't found much benefit from doing so. We have elaborated on the related issues in the discussion of Experimental section.

6. CONCLUSIONS

In this paper, we first studied POI recommendation from the viewpoint of weighted matrix factorization. We then augmented the latent space obtained from weighted matrix factorization with geographical information. Particularly, users' latent factors were augmented with users' activity areas while POIs' latent factors were augmented with POIs' influence areas. At the same time, sparse and non-negative constraints were imposed on both of them. Based on this augmented model, we did not only modeled the spatial clustering phenomenon from the novel perspective of two-dimensional kernel density estimation, but we also explained why the modeling of the spatial clustering phenomenon can help to deal with the challenge of matrix sparsity. Then we evaluated the proposed model on a large-scale check-in dataset. The experimental results showed that weighted matrix factorization is superior to other kinds of factorization models for POI recommendation and that incorporating the modeling of the spatial clustering phenomenon into matrix factorization improved the recommendation performance. Moreover, the imposed sparsity constraint also boosted the recommendation performance at the same time of promoting efficiency.

7. REFERENCES

- [1] G. Adomavicius and A. Tuzhilin. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Trans. Know. Data. Eng.*, 17(6):734–749, 2005.
- [2] C. Cheng, H. Yang, I. King, and M. Lyu. Fused matrix factorization with geographical and social influence in location-based social networks. In *Proceedings of AAAI'12*, 2012.
- [3] C. Cheng, H. Yang, M. R. Lyu, and I. King. Where you like to go next: successive point-of-interest recommendation. In *Proceedings of IJCAI'13*, pages 2605–2611. AAAI Press, 2013.
- [4] V. Franc, V. Hlaváč, and M. Navara. Sequential coordinate-wise algorithm for the non-negative least squares problem. In *Computer Analysis of Images and Patterns*, pages 407–414. Springer, 2005.
- [5] H. Gao, J. Tang, X. Hu, and H. Liu. Exploring temporal effects for location recommendation on location-based social networks. In *Proceedings of RecSys'13*, pages 93–100. ACM, 2013.
- [6] T. Hastie, R. Tibshirani, and J. Friedman. The elements of statistical learning: data mining, inference, and prediction, 2001.
- [7] T. Horozov, N. Narasimhan, and V. Vasudevan. Using location for personalized poi recommendations in mobile environments. In *Proceedings of SAINT'06*. IEEE Computer Society, 2006.
- [8] Y. Hu, Y. Koren, and C. Volinsky. Collaborative filtering for implicit feedback datasets. In *Proceedings of ICDM'08*, pages 263–272. IEEE, 2008.
- [9] Y. Koren, R. Bell, and C. Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37, 2009.
- [10] C. L. Lawson and R. J. Hanson. *Solving Least Squares Problems*, volume 15. SIAM, 1995.
- [11] C.-J. Lin. Projected gradient methods for nonnegative matrix factorization. *Neural computation*, 19(10):2756–2779, 2007.
- [12] B. Liu, Y. Fu, Z. Yao, and H. Xiong. Learning geographical preferences for point-of-interest recommendation. In *Proceedings of KDD'13*, pages 1043–1051. ACM, 2013.
- [13] B. Liu and H. Xiong. Point-of-interest recommendation in location based social networks with topic and location awareness. In *Proceedings of SDM'13*, pages 396–404. SIAM, 2013.
- [14] X. Liu, Y. Liu, K. Aberer, and C. Miao. Personalized point-of-interest recommendation by mining users' preference transition. In *Proceedings of CIKM'13*, pages 733–738. ACM, 2013.
- [15] H. Ma, C. Liu, I. King, and M. R. Lyu. Probabilistic factor models for web site recommendation. In *Proceedings of SIGIR'11*, pages 265–274. ACM, 2011.
- [16] K. P. Murphy. *Machine learning: a probabilistic perspective*. The MIT Press, 2012.
- [17] A. Noulas, S. Scellato, N. Lathia, and C. Mascolo. A random walk around the city: New venue recommendation in location-based social networks. In *Proceedings of SocialCom'12*, pages 144–153. IEEE, 2012.
- [18] R. Pan, Y. Zhou, B. Cao, N. Liu, R. Lukose, M. Scholz, and Q. Yang. One-class collaborative filtering. In *Proceedings of ICDM'08*, pages 502–511. IEEE, 2008.
- [19] M. Park, J. Hong, and S. Cho. Location-based recommendation system using bayesian user's preference model in mobile devices. *Ubiquitous Intelligence and Computing*, pages 1130–1139, 2007.
- [20] D. Seung and L. Lee. Algorithms for non-negative matrix factorization. *Advances in neural information processing systems*, 13:556–562, 2001.
- [21] W. Tobler. A computer movie simulating urban growth in the detroit region. *Economic geography*, 46:234–240, 1970.
- [22] D. Yang, D. Zhang, Z. Yu, and Z. Wang. A sentiment-enhanced personalized location recommendation system. In *Proceedings of the 24th ACM Conference on Hypertext and Social Media (HT'13)*, pages 119–128. ACM, 2013.
- [23] M. Ye, D. Shou, W. Lee, P. Yin, and K. Janowicz. On the semantic annotation of places in location-based social networks. In *Proceedings of KDD'11*, pages 520–528. ACM, 2011.
- [24] M. Ye, P. Yin, W.-C. Lee, and D.-L. Lee. Exploiting geographical influence for collaborative point-of-interest recommendation. In *Proceedings of SIGIR'11*, pages 325–334. ACM, 2011.
- [25] H. Yin, Y. Sun, B. Cui, Z. Hu, and L. Chen. Lcars: a location-content-aware recommender system. In *Proceedings of KDD'13*, pages 221–229. ACM, 2013.
- [26] J.-D. Zhang and C.-Y. Chow. igslr: Personalized geo-social location recommendation a kernel density estimation approach. In *Proceedings of GIS'13*.
- [27] V. Zheng, Y. Zheng, X. Xie, and Q. Yang. Collaborative location and activity recommendations with gps history data. In *Proceedings of WWW'10*, pages 1029–1038. ACM, 2010.
- [28] Y. Zheng, L. Zhang, Z. Ma, X. Xie, and W. Ma. Recommending friends and locations based on individual location history. *ACM Trans. Web*, 5(1):5, 2011.