

## Research Article

# Network Embedding-Aware Point-of-Interest Recommendation in Location-Based Social Networks

Lei Guo <sup>1,2</sup>, Haoran Jiang <sup>3</sup>, Xiyu Liu,<sup>2</sup> and Changming Xing<sup>4</sup>

<sup>1</sup>Postdoctoral Research Station of Management Science and Engineering, Shandong Normal University, Jinan, China

<sup>2</sup>Shandong Normal University, Jinan, China

<sup>3</sup>Shandong Post Company, Jinan, China

<sup>4</sup>Shandong University of Finance and Economics, Jinan, China

Correspondence should be addressed to Lei Guo; [leiguoc.s@gmail.com](mailto:leiguoc.s@gmail.com)

Received 9 February 2019; Revised 13 September 2019; Accepted 3 October 2019; Published 4 November 2019

Academic Editor: Xiaoping Liu

Copyright © 2019 Lei Guo et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

As one of the important techniques to explore unknown places for users, the methods that are proposed for point-of-interest (POI) recommendation have been widely studied in recent years. Compared with traditional recommendation problems, POI recommendations are suffering from more challenges, such as the cold-start and one-class collaborative filtering problems. Many existing studies have focused on how to overcome these challenges by exploiting different types of contexts (e.g., social and geographical information). However, most of these methods only model these contexts as regularization terms, and the deep information hidden in the network structure has not been fully exploited. On the other hand, neural network-based embedding methods have shown its power in many recommendation tasks with its ability to extract high-level representations from raw data. According to the above observations, to well utilize the network information, a neural network-based embedding method (node2vec) is first exploited to learn the user and POI representations from a social network and a predefined location network, respectively. To deal with the implicit feedback, a pair-wise ranking-based method is then introduced. Finally, by regarding the pretrained network representations as the priors of the latent feature factors, an embedding-based POI recommendation method is proposed. As this method consists of an embedding model and a collaborative filtering model, when the training data are absent, the predictions will mainly be generated by the extracted embeddings. In other cases, this method will learn the user and POI factors from these two components. Experiments on two real-world datasets demonstrate the importance of the network embeddings and the effectiveness of our proposed method.

## 1. Introduction

With the popularity of mobile devices and the development of global positioning system, location-based social networks (LBSNs) have emerged in recent years (e.g., Gowalla (<https://en.wikipedia.org/wiki/Gowalla>) and Foursquare (<https://foursquare.com/>)), which enable people to share their experiences with friends and check in at interesting places (e.g., restaurants, tourist spots, and stores). POI recommendation as one of the important techniques to explore interesting unknown places has been well studied.

Compared with traditional recommendation tasks, POI recommendations are challenging for three reasons. First,

POI recommendation methods often suffer from the data scarcity and user/item cold-start problems, which means only a small number of users have sufficient check-in data, and most of the users have very little or even no check-in data (as shown in Figure 1). In our experiments, the data scarcity is more than 99%. Second, in LBSNs, the check-in data are not explicit but implicit, which makes POI recommendation more difficult. In check-in data, only positive behaviors can be observed, from which we can only get the POIs that the users have visited, but cannot obtain the POIs that the users do not prefer. The missed check-ins are mixed of actually negative samples and undiscovered but potentially positive POIs. We cannot learn the user interests

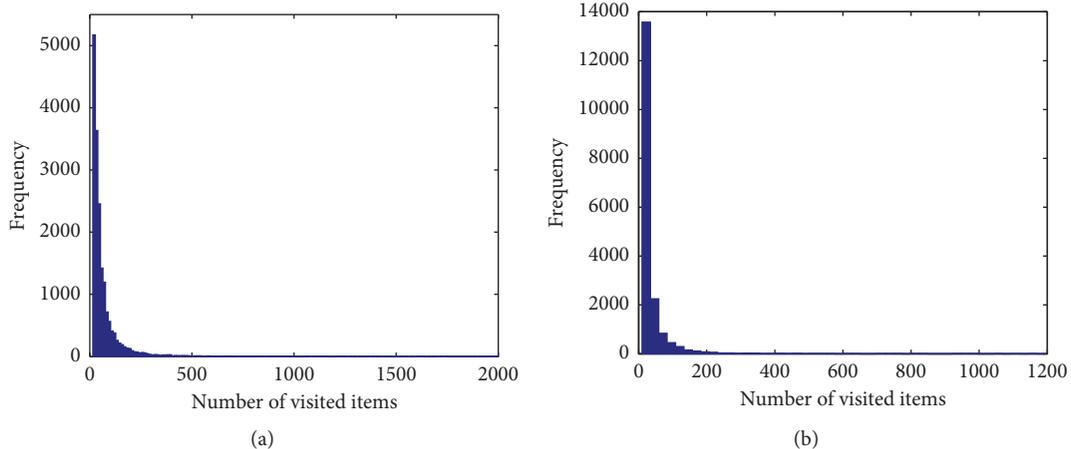


FIGURE 1: The histogram of users' visited POIs: (a) Gowalla and (b) Yelp.

directly from this mixed data by utilizing a common approach, since they are unable to be differentiated from these two levels anymore. Third, users and POIs have different types of context information, e.g., social relationships of users and geographical information of POIs. It is important to exploit context information to improve recommendation accuracy. For example, in real life, we usually turn to our friends for POI recommendation. The social relationships among users help them locate the POIs that they are potentially interested in. In our real life, geographical information is another important type of context, which is not available in conventional recommendation task. As users are more likely to visit nearby POIs, the geographical information can help them locate the POIs that have neighborhood relationship with their preferred places.

How to overcome these challenges to further improve the recommendation performance has become one of the hot spots in recent studies. For example, Ye et al. [1] exploited the user preference and social and geographical influence for POI recommendation. In their work, a power law distribution is utilized to model the discovered spatial clustering phenomenon. Cheng et al. [2] captured the geographical influence by a multicenter Gaussian model and then fused it with matrix factorization (MF) and social influence for POI recommendation. Different from the work in [1, 2], a kernel density estimation approach was proposed by Zhang and Chow [3] to model the users' visiting behaviors. To deal with the implicit check-in data, Lian et al. [4] first introduced the weighted matrix factorization (WMF) method as their basic framework, which fits the positive user activities by utilizing larger weights and unobserved user behaviors by utilizing smaller weights. Then, an augmented factor model was proposed to incorporate the geographical influence. Instead of directly fitting zero check-ins, Li et al. [5] treated POI recommendation as an item ranking problem and learned the latent factors by optimizing an ordered weighted pair-wise classification criterion, which further considers the ranking task as a set of classification problems. Ying et al. [6] exploited the recommendation performance in another famous ranking criterion, i.e., the

Bayesian personalized ranking (BPR) method [7], and proposed a pair-wise-based ranking method.

Although the above methods can work well in many recommendation scenarios, most of them only model the context information as regularization terms. The effectiveness of the deep information hidden in the network structure has not been fully exploited. On the other hand, deep learning-based embedding technique has shown its power in many recommendation tasks with its capability of extracting hierarchical representations from raw data [8]. For example, Wang et al. [9] applied the embedding model learned from the sequential behavior into next basket recommendation problem and proposed a two-layer hierarchical representation model. Zhao et al. [10] utilized the recurrent neural networks to learn the feature presentations of both users and products from an e-commerce website and proposed a feature-based matrix factorization method to leverage the learned features. However, most of these existing works mainly learn the embedding models of users and items from sequential behavior and content information, and little work has been done to leverage the representations deduced from social networks and location neighborhood relationships to deal with the cold-start POI recommendation. For example, Grover and Leskovec [11] proposed a novel method node2vec for learning continuous feature representations of nodes in networks. But their work did not investigate the recommendation problem, and the learned embeddings cannot be directly utilized to collaborative filtering method.

Based on the above observations, we propose a network embedding-based POI recommendation method, which consists of an embedding model and a collaborative filtering model. In the embedding model, a neural network, i.e., the node2vec model [11], is pretrained to learn the user and POI representations from the corresponding social network and a predefined location network, respectively. In the collaborative filtering model, to deal with the implicit check-in data, the ranking-based criterion is first introduced, which learns the user and POI factors by ranking the items correctly. Then, the outputs of the node2vec are treated as the high-level representations of social and location context, and

they are further used as the priors of latent feature factors in the collaborative filtering model. In experiments, we evaluate our method in two real-world datasets [12], and the results indicate that our method is more effective than other related state-of-the-art recommendation methods.

The main contributions of this paper are as follows:

- (i) We propose a network embedding-based POI recommendation method, which formulates the POI recommendation task as an item ranking problem, and considers the influence of social networks and geographical information by leveraging the capability of neural networks in extracting high-representative features from pre-defined networks
- (ii) We incorporate the learned embeddings as the priors of the latent feature factors in collaborative filtering model to enable our method can consider the network information and collaborative filtering information simultaneously
- (iii) We conduct extensive experiments on two real-world datasets to demonstrate the superiority of our proposed method

We organize the remainder of this paper as follows: Section 2 reviews and discusses the related works to our study. Section 3 describes the POI recommendation problem and proposes a ranking-based recommendation framework with pretrained network embeddings. Section 4 shows the experimental results on two real-world datasets. Section 5 concludes this work and points out the future work.

## 2. Related Work

In this section, some works related to our studies are investigated, which include the context-based recommendation, implicit feedback-based recommendation, and network embedding-based recommendation.

*2.1. POI Recommendation with Social and Geographical Context.* In LBSNs, social and geographical context plays an important role in helping users explore the unknown locations [12–17]. Motivated by the intuitions that users are more inclined to visit the places that their friends have suggested, and nearby locations tend to be visited by similar users, the social and geographical influence has been utilized to improve the POI recommendation. Cho et al. [18] studied the relation between human geographic movement, its temporal dynamics, and ties of the social network. They found that people generally move periodically within a bound region but occasionally travel long distance due to the impact of the social relationships. Ye et al. [1] first modeled the spatial clustering phenomenon by a power law distribution. Then, a unified POI recommendation framework was proposed, which fuses user preference to a POI with geographical influence and social influence. Cheng et al. [2] first captured the geographical influence by utilizing the multicenter Gaussian model and then fused it with the social influence in a generalized matrix factorization framework.

Unlike previous proposed methods, which model the user check-in activities universally, Zhang et al. [19] proposed a personalized geographical location recommendation framework, where each user is modeled as a personal distribution. Li et al. [20] improved the POI recommendation by leveraging the information of friends in a two-step framework, where a set of potential locations that each user’s friends have checked in were first learned, and then three types of check-ins were incorporated into the matrix factorization model. Liu et al. [21] proposed a general geographical probabilistic factor model to jointly learn both geographical preference and user interest preference, where a Poisson-based probabilistic generative process is utilized for the recommendation model. Wang et al. [22] constructed a graph to model users and their social relations, which models the user preference and social influence by utilizing different types of edges. To capture the asymmetry of geographical influence and the high variation of geographical influence across POIs, Wang et al. [23] exploited POI-specific geographical influence to improve POI recommendation and modeled the geographical influence between two POIs by using three factors: the geo-influence of POI, the geo-susceptibility of POI, and their physical distance. To incorporate the geographical context information, Ma et al. [24] proposed a novel autoencoder-based model to learn the complex user-POI relations. Guo et al. [25] explored the influence of geographical information from a location perspective and proposed a constrained ranking function by using a location regularization term. However, they did not consider the deep structure of social networks and POI relationships, and the influence of them is still largely unexplored.

Except the social and geographical information, other types of contexts have also been explored [8,26–29], such as the temporal influence [30], the sequential influence, and the content influence [31, 32]. For example, based on the temporal properties of user movement observed from a real-world LBSN dataset, Gao et al. [33] proposed a temporal aware recommendation framework. Qian et al. [34] focused on the joint effect of time and location for POI recommendation and proposed a spatiotemporal context-aware and translation-based recommender framework (STA) to model the third-order relationship among users, POIs, and spatiotemporal contexts for large-scale POI recommendation. In literature [28], they further studied three types of content information that is related to the user’s check-in actions and then modeled them in a unified POI recommendation framework. In addition to social and geographical influence, Zhang et al. [35] also considered the sequential influence. In this work, they first mined sequential patterns from location sequences and represented the sequential patterns as a dynamic graph. Then, they predicted the probability of a user visiting a location by additive Markov chain with this graph. Zhang and Chow [36] modeled three types of context, that is, geographical, social, and categorical correlations, for recommendation, and utilized the two-dimensional KDE for geographical modeling and the power-law distribution for social and categorical influence. Lian et al. [37] proposed a scalable implicit

feedback-based content-aware collaborative filtering framework to incorporate semantic content and to steer clear of negative sampling. However, their work is developed for semantic information modeling, while ours is focused on exploring network embedding information. Yang et al. [38] proposed a neural network to jointly model the social networks and mobile trajectories, which considers four influential factors, namely, user visit preference, influence of friends, and short-term and long-term sequential contexts. But their method is developed for friend recommendation and next location recommendation tasks. Next location recommendation is different from the general POI/location recommendation problem, since it focuses on modeling the location's sequential information rather than user's historical preferences. From the above studies, we notice that these types of contexts are also important factors for POI recommendations and are often fused together with social and geographical context to make more accurate predictions.

*2.2. Implicit Feedback-Based Recommendation.* In many recommendation scenarios, we can only get the user's positive behaviors through what they have purchased, visited, or clicked, but never know what they do not like. The feedback usually is not explicit, but implicit, which makes the recommendation task more challenging (the implicit feedback also is well known as the one-class collaborative filtering (OCCF) problem) [39, 40]. Note that, although the OCCF problem we studied in this work has similar problem definition with positive unlabeled learning (PU learning) [41–44], most of the solutions proposed for PU learning cannot be directly applied to OCCF problem. The main reason is that most of them are focused on dealing with documents or reviews, where a lot of document features can be exploited to decide the true-negative samples initially. But in OCCF problem, we usually can only obtain the positive interactions among users and items, and it is hard to identify the true-negative items without any additional information. Most of the recommendation-oriented methods are trying to model the implicit feedback in a ranking or weighted way. For example, Pan et al. [45] modeled the implicit feedback by using the weighted low rank approximations; that is, they gave larger weights to the positive samples and smaller weights to the negative samples. But their work needs to directly model all the missed values as negative samples. Rendle et al. [7] proposed a pair-wise ranking-based criterion to directly optimize the loss error for ranking, and a generic ranking framework derived from a Bayesian analysis of the problem was presented. Lian et al. [4] extended the weighted matrix factorization method proposed in [45] for POI recommendation, where the latent feature factors of users and POIs are augmented by the activity area vector and the influence area vector, respectively. For the sake of improving efficiency and flexibility, Lian et al. [46] further extended the GeoMF [4] and proposed GeoMF++, which map the location influential areas into the same latent space as that formed by weighted regularized matrix factorization. Different from [4], user activity area is not used in GeoMF++ anymore, but can be recovered from user latent factors using

negative sparse coding. Liu et al. [47] exploited the geographical characteristics from a location perspective, where the neighborhood is modeled at two levels, i.e., the instance-level neighborhood that is defined by its nearest neighbors and the region-level neighborhood that is defined by the geographical region where the location exists. Then, these two levels of geographical characteristics are incorporated to the weighted matrix factorization to learn the latent feature factors of users and POIs. Li et al. [5] investigated the POI recommendation by treating it as the ordered weighted pair-wise classification problem and proposed a ranking-based geographical factorization model. Zhao et al. [48] first proposed the cogeographical influence to filter geographically noisy POIs and then incorporated it into a personalized pair-wise preference ranking matrix factorization model. Although ranking-based POI recommendations have been well studied, most of them do not consider incorporating these contexts by utilizing the embedding techniques and cannot be directly applied to the cold-start situations.

*2.3. Embedding-Based Recommendation.* With the development of distributed representation techniques [49–51], embedding-based models have been proved in many recommendation and other related tasks [10, 52, 53]. For example, Liang et al. [54] proposed a content-aware collaborative music recommendation by first training a neural network on semantic tagging information and then used it as the prior in a collaborative filtering model. He et al. [55] proposed a scalable factorization model to incorporate visual signals into predictors of people's opinions, where the visual signals are learned from product images by using the pretrained neural networks. Wang et al. [56] further studied the problem of enhancing POI recommendation with visual contents, where they used the convolutional neural network to extract features from images and used it to guide the learning process of latent user and POI features. Deng et al. [57] employed the deep autoencoder technique to improve the quality of the initial vectors of MF for trust-aware social recommendations and to differentiate the community effect in users' trusted friendships. Except the above models that integrate deep learning with traditional recommendation models loosely, there are also tightly coupled models, that is, learn the parameters from these two components simultaneously. In this way, the learning process of latent factor model and feature extracting model mutually influence each other. For instance, Chen et al. [58] introduced an attention-based neural network to collaborative filtering to address the implicit feedback problem in multimedia recommendation. Guo et al. [59] combined the factorization machine and deep learning for feature learning and proposed a novel neural network architecture. To deal with the inherent limitation of the bag-of-words model, Kim et al. [60] integrated the convolutional network into probabilistic matrix factorization model to capture the contextual information of documents and enhance the rating prediction accuracy. Guo et al. [61] incorporated a pretrained network embedding into the weighted matrix factorization method. However, their

method was proposed for social recommendation and the network embedding was integrated by a linear latent factor model, and ours is regarded as the prior knowledge of the latent feature factor.

**2.4. Differences.** Our method has significant differences with these existing methods. First, although Yang et al. [38] adopted a network embedding method for the construction of social networks, it is proposed for friend recommendation and next location recommendation. Moreover, Yang et al. [38] did not consider the influence of geographical information. But in real world, nearby POIs are more likely to be visited by similar users, and the geographical relationship between POIs is also an important factor to the POI recommendation task. Hence, we further incorporate the POI-POI graph to enhance the representation of POI embeddings. Second, although there are existing works that formulate the POI recommendation as an item ranking task and try to optimize a BPR loss function, they did not consider the performance of leveraging network embeddings as user’s and item’s prior knowledge. The advantage of our method is that it can enhance POI recommendation by exploiting high-level embeddings from external network information and can make cold-start predictions for the users/items that have no training data in history by only using the learned distributional representations.

### 3. Our Approach

In this section, we first describe the POI recommendation task in LBSNs. Then, the users’ high-level representations and social-aware recommendation method are proposed. To exploit the influence of neighborhood relationships among locations, we then learn the POI representations from a predefined location network. Finally, by fusing the extracted embeddings of users and POIs with collaborative filtering model, the location-aware social POI recommendation method is achieved.

**3.1. Task Definition.** Let us consider the POI recommendation problem in LBSNs, where users can express their preferences by checking in at different locations, and the closer the two locations are, the more likely they are to be visited by similar users. But different from traditional social networks, in LBSNs, only positive behaviors can be observed, that is, we can only know the places that the user has checked in, but never know the places that the user does not like. In other words, the check-in data are not explicit but implicit. In addition to these check-in activities, users can also express their preferences by making different friends, and the closer the relationship between two friends is, the more likely they are to be affected. This recommendation scenario is shown in Figure 2, which is associated with three central elements: the user check-in data (also known as user-POI check-in matrix), social network, and location network (i.e., POI-POI relation network).

Figure 2 shows a toy example, where users express their visiting preferences by checking in at different places. But

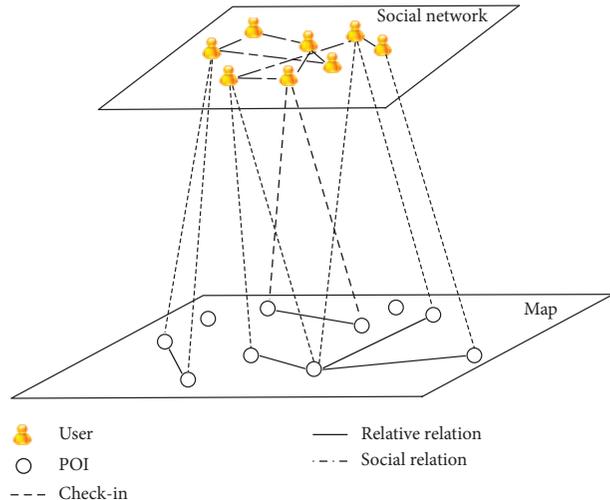


FIGURE 2: POI recommendation scenario in LBSNs, which consists of three kinds of information: the user-POI check-in matrix, social network, and POI-POI network.

the check-in data are very sparse (some users are totally new to the system), only a small number of users’ activities can be observed, and the remaining are unknown data (denoted as “?”). Besides the check-in matrix, users have also made some friends that have common interests with them to further explore the unknown places. The social relationship helps us make more accurate recommendations by leveraging the locations that their friends have visited. Moreover, in POI recommendation, neighboring locations tend to share common users, which means users are more inclined to visit nearby places. This kind of context is also beneficial for our recommender system. Hence, how to make effective recommendations by utilizing the above contexts from the implicit check-in data is the main task of this work.

**3.2. The Pair-Wise Item Ranking Criterion.** To deal with the implicit check-in data, we treat POI recommendation problem as the pair-wise item ranking task and introduce the Bayesian personalized ranking (BPR) criterion [7] as our basic recommendation framework. Instead of optimizing individual scores for a single item, BPR infers the latent feature factors by optimizing the partial ordering relationships between items, which is more suitable to the implicit feedback.

Suppose we have a check-in matrix  $R \subseteq \mathcal{U} \times \mathcal{F}$  (as shown in Figure 3), where  $\mathcal{U}$  denotes the user set and  $\mathcal{F}$  denotes the POI set. The entry  $r_{ui} = 1$  in this matrix represents that user  $u$  has visited POI  $i$  in history, and  $r_{ui} = ?$  represents that the status of  $u$  to  $i$  is unknown, which means  $i$  is unattractive or undiscovered to user  $u$ . We cannot directly treat all these unknown entries as negative samples, as they may be potential attractive POIs to users. To provide a personalized total ranking  $\succ_u \subset \mathcal{F}^2$  of all POIs to the user, BPR method proposes a pair-wise data policy to reconstruct the user-POI check-in matrix, which assumes that compared with unvisited POIs, users will prefer the POIs that he/she has

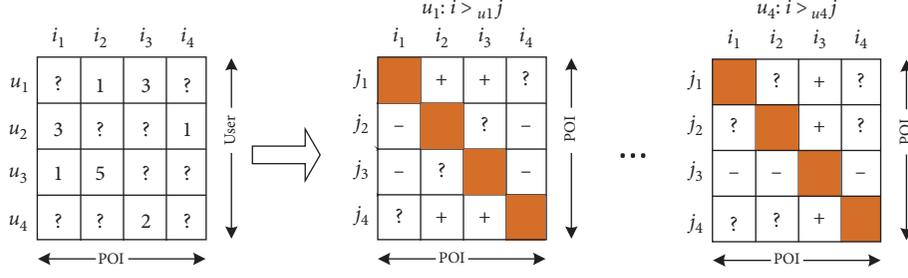


FIGURE 3: The reconstructing data policy from user-POI check-in matrix to pair-wise POI dataset.

visited. From Figure 3, we can find some examples of creating pair-wise preferences  $i >_u j$  for user  $u$  from the check-in matrix, where the sign “+” represents that  $u$  prefers  $i$  over  $j$  and the sign “-” represents that user  $u$  prefers  $j$  over  $i$  (see the left part of Figure 3). For example, as user  $u_1$  has checked in POI  $i_2$  and unvisited POI  $i_4$ , then according to our data policy the partial order between  $i_2$  and  $i_4$  can be created as  $i_2 >_u i_4$  ( $u$  prefer  $i_2$  over  $i_4$ ). But for the POIs that both have been checked in by  $u$ , we cannot deduce any preference for him/her (e.g., POI  $i_2$  and  $i_3$  for user  $u_1$ ). The same is true for the POIs that both have been unvisited by the user (e.g., POI  $i_2$  and  $i_3$  for user  $u_2$ ). Then, we can formalize the created training data  $\mathcal{D}_{\mathcal{R}}: \mathcal{U} \times \mathcal{I} \times \mathcal{I}$  as

$$\mathcal{D}_{\mathcal{R}} := \{(u, i, j) \mid i \in \mathcal{I}_u^+ \wedge j \in \mathcal{I} \setminus \mathcal{I}_u^+\}, \quad (1)$$

where  $\mathcal{I}_u^+$  represents the POI set that has been checked in by user  $u$ ,  $\mathcal{I} \setminus \mathcal{I}_u^+$  denotes the unvisited POI set by user  $u$ , and  $(u, i, j) \in \mathcal{D}_{\mathcal{R}}$  means user  $u$  prefers POI  $i$  over  $j$ . As the ranking with respect to  $u$  ( $>_u$ ) is antisymmetric, the negative samples are regarded implicitly.

To rank all the POIs correctly for every user, BPR maximizes the posterior probability of the model parameter from a Bayesian analysis of the problem. By using  $p(i >_u j \mid \Theta)$  as the likelihood function and  $p(\Theta)$  as the prior function, we can arrive at the objective function of BPR:

$$\text{BPR} = - \sum_{(u, i, j) \in \mathcal{D}_{\mathcal{R}}} \ln \sigma(\hat{r}_{uij}(\Theta)) + \lambda_{\Theta} \|\Theta\|_F^2, \quad (2)$$

where  $\Theta$  is the model parameter,  $\sigma(x)$  is the logistic sigmoid function, and  $\lambda_{\Theta}$  is the regularization parameter.  $\hat{r}_{uij}(\Theta)$  captures the relationship among  $u$ ,  $i$ , and  $j$ , which can be calculated by any real-valued score function of  $\Theta$ . By decomposing the estimator  $\hat{r}_{uij}$  as  $\hat{r}_{uij} = w_{uij}(\hat{r}_{ui} - \hat{r}_{uj})$ , and delegating the task of modeling  $\hat{r}_{uij}$  to the matrix factorization (MF) method, the MF-based BPR method can be arrived at

$$\text{BPRMF} = - \sum_{(u, i, j) \in \mathcal{D}_{\mathcal{R}}} \ln \sigma(\hat{r}_{ui} - \hat{r}_{uj}) + \frac{\lambda_U}{2} \|U\|_F^2 + \frac{\lambda_V}{2} \|V\|_F^2, \quad (3)$$

where  $V \in \mathbb{R}^{d \times n}$  and  $U \in \mathbb{R}^{d \times m}$  denote the latent factors of POIs and users, respectively.  $\lambda_U$  and  $\lambda_V$  denote the regularization parameters.  $\hat{r}_{uv} = U_u^T V_v$  is the score function computed by the dot product of  $U$  and  $V$ . As in MF, we also place the zero-mean spherical Gaussian priors on the latent factors  $U$  and  $V$ :

$$p(U \mid \sigma_U^2) = \prod_{u=1}^m \mathcal{N}(U_u \mid 0, \sigma_U^2 \mathbf{I}), \quad (4)$$

$$p(V \mid \sigma_V^2) = \prod_{v=1}^n \mathcal{N}(V_v \mid 0, \sigma_V^2 \mathbf{I}), \quad (5)$$

where  $\mathcal{N}(x \mid \mu, \sigma^2)$  denotes the Gaussian distribution with variance-covariance matrix  $\sigma^2$  and mean  $\mu$ .

**3.3. Pretrained Social Network Embedding.** Inspired by the success of neural network-based embedding model in link prediction and node classification, in this work we pretrain a network embedding model node2vec [11] (based on the skip-gram architecture) in an semisupervised task and use the learned embeddings as the representations of social networks (the project of node2vec is available at <https://snap.stanford.edu/node2vec/>). The training data we used come from the Gowalla dataset [12], which consists of 1.2 million check-ins (the details can be seen in Section 4.1). The social network we used is the friend relationships among 18,738 users, which can be represented as an unweighted and undirected graph. Let  $\mathcal{G} = (\mathcal{U}, \mathcal{E})$  be the given social network and  $f: \mathcal{U} \rightarrow \mathbb{R}^D$  be the mapping function from nodes to the  $D_s$  dimension feature representations, where  $\mathcal{U}$  denotes the user set and  $\mathcal{E}$  denotes the friend relationship set. For every source node  $u \in \mathcal{U}$ , we define  $N_S(u) \in \mathcal{U}$  as a network neighborhood of node  $u$  generated through a neighborhood sampling strategy  $S$ , that is, a biased random walk method that can explore neighborhoods in a breadth-first sampling as well as depth-first sampling fashion.

To learn the high-level representations of every node, node2vec tries to maximize the log-probability of observing a network neighborhood  $N_S(u)$  for a node  $u$  conditioned on its feature representation, given by  $f$ :

$$\max_f \sum_{u \in \mathcal{U}} \log P(N_S(u) \mid f(u)). \quad (6)$$

To make the above optimization problem tractable, two standard assumptions are made in node2vec: (1) conditional independence, which assumes the observation likelihood of a neighborhood node is independent of the likelihood of observing any other neighborhood node given the feature representation of the source, and (2) symmetry in feature space, which means a source node and neighborhood node have a symmetric effect over each other in feature space.

With these two assumptions, the objective function of node2vec is arrived at

$$\max_f \sum_{u \in \mathcal{U}} \left[ -\log Z_u + \sum_{n_i \in \mathcal{N}_s(u)} f(n_i) \cdot f(u) \right], \quad (7)$$

where  $Z_u = \sum_{v \in \mathcal{U}} \exp(f(u) \cdot f(v))$  denotes the per-node partition function. As it is expensive to compute for large networks, the negative sampling method [62] is used to approximate it. In their work, stochastic gradient ascent method is used to learn the model parameters.

In our work, the parameters of node2vec are set as follows: the dimension of the learned feature vectors is set as  $D_s = 128$ , the length of walk per source is set as  $l = 80$ , the number of walks per source is set as  $r = 10$ , the context size is set as  $k = 10$ , and the optimization runs only a single epoch. The best in-out and return parameters are achieved by using 10-fold cross-validation on tune dataset and are set as  $p = 1$  and  $q = 2$ .

**3.4. Social-Aware Recommendation.** The output of node2vec can be interpreted as the network representations of users. As users always express their preferences by making different friends, these representations denote how well a user is influenced by his friends. By introducing the network feature as the prior of the latent user factor in BPRMF model, the distribution of  $U$  can be modified as

$$p(U | X, S, \sigma_U^2) = \prod_{u=1}^m \mathcal{N}(U_u | XS_u, \sigma_U^2 \mathbf{I}), \quad (8)$$

where  $S_u \in \mathbb{R}^{D_s}$  denotes the learned network embedding of user  $u$  ( $D_s$  is the dimension of  $S_u$ ) and  $X \in \mathbb{R}^{d \times D_s}$  is the weight matrix that transforms the learned social network representations from the neural networks into the collaborative filtering latent space (via  $XS_u$ ).

To complete the inference, we also assume that the weight matrix  $X$  follows the zero-mean spherical Gaussian distribution:

$$p(X | \sigma_X^2) = \prod_{x=1}^d \prod_{s=1}^{D_s} \mathcal{N}(X_{xs} | 0, \sigma_X^2). \quad (9)$$

Then, the posterior probability over the user, item, and weight latent factor vectors can be rewritten as

$$\begin{aligned} & p(U, V, X | \mathcal{D}_{\mathcal{R}}, S, \sigma_X^2, \sigma_U^2, \sigma_V^2) \\ & \propto p(\mathcal{D}_{\mathcal{R}} | U, V) p(U | X, S, \sigma_U^2) p(V | \sigma_V^2) p(X | \sigma_X^2) \\ & = \prod_{(u,i,j) \in \mathcal{D}_{\mathcal{R}}} \sigma(\hat{r}_{uij}) \times \prod_{u=1}^m \mathcal{N}(U_u | XS_u, \sigma_U^2 \mathbf{I}) \\ & \times \prod_{v=1}^n \mathcal{N}(V_v | 0, \sigma_V^2 \mathbf{I}) \times \prod_{x=1}^d \prod_{s=1}^{D_s} \mathcal{N}(X_{xs} | 0, \sigma_X^2). \end{aligned} \quad (10)$$

The log-posterior probability of  $U, V$ , and  $X$  is derived as

$$\begin{aligned} \ln p(U, V, X | \mathcal{D}_{\mathcal{R}}, S, \sigma_X^2, \sigma_U^2, \sigma_V^2) & = \sum_{(u,i,j) \in \mathcal{D}_{\mathcal{R}}} \ln \sigma(\hat{r}_{uij}) \\ & - \frac{\lambda_U}{2} \sum_{u=1}^m (U_u - XS_u)^T \\ & \cdot (U_u - XS_u) \\ & - \frac{1}{2\sigma_V^2} \sum_{k=1}^n V_k^T V_k \\ & - \frac{1}{2\sigma_X^2} \sum_{x=1}^d X_x^T X_x \\ & - \frac{1}{2} ((m \times d) \ln \sigma_U^2 + (n \times d) \\ & \cdot \ln \sigma_V^2 + (d \times D_s) \ln \sigma_X^2) + C. \end{aligned} \quad (11)$$

Keeping the parameters fixed, maximizing the log-posterior over the latent feature vectors is equivalent to minimize the following objective function, that is, the objective function of our social-aware recommendation method:

$$\begin{aligned} \text{SRank}(\mathcal{D}_{\mathcal{R}}, X, U, V) & = - \sum_{u=1}^m \sum_{i \in \mathcal{F}_u^+} \sum_{j \in \mathcal{F}_u^-} \ln \sigma(\hat{r}_{uij}) \\ & + \frac{\lambda_U}{2} \sum_{u=1}^m (U_u - XS_u)^T (U_u - XS_u) \\ & + \frac{\lambda_X}{2} \|X\|_F^2 + \frac{\lambda_V}{2} \|V\|_F^2, \end{aligned} \quad (12)$$

where  $\|\cdot\|_F^2$  is the Frobenius norm, and  $\lambda_U = 1/\sigma_U^2$ ,  $\lambda_V = 1/\sigma_V^2$ , and  $\lambda_X = 1/\sigma_X^2$  are the regularization parameters of  $U, V$ , and  $X$ , respectively. Gradient descent-based algorithm is an obvious option to help find the local minimum of this objective function. And to alleviate the influence of huge training pairs ( $O(|\mathcal{U}||\mathcal{I}|)$ ), the stochastic gradient descent method is utilized in our algorithm. The gradients of the corresponding latent factors are shown as follows:

$$\frac{\partial \text{SRank}}{\partial X} = -\lambda_U U_u S_u^T + \lambda_U X S_u S_u^T + \lambda_X X, \quad (13)$$

$$\frac{\partial \text{SRank}}{\partial U_u} = -\frac{e^{-\hat{r}_{uij}}}{1 + e^{-\hat{r}_{uij}}} (V_i - V_j) + \lambda_U U_u - \lambda_U X S_u, \quad (14)$$

$$\frac{\partial \text{SRank}}{\partial V_i} = -\frac{e^{-\hat{r}_{uij}}}{1 + e^{-\hat{r}_{uij}}} U_u + \lambda_V V_i, \quad (15)$$

$$\frac{\partial \text{SRank}}{\partial V_j} = \frac{e^{-\hat{r}_{uij}}}{1 + e^{-\hat{r}_{uij}}} U_u + \lambda_V V_j. \quad (16)$$

**3.5. Pretrained Location Network Embeddings.** To utilize the context of geographical information, we first deduce a POI-POI graph based on the intuition that nearby locations are more likely to be visited by similar users and then pretrain the node2vec model for learning the representations of POIs. The definition of the POI-POI graph is shown as follows.

**Definition 1 (POI-POI graph).** The POI-POI graph is denoted as  $\mathcal{G}_V = (\mathcal{V}, \mathcal{E}_V)$ , which captures the neighborhood relationships of POIs, where  $\mathcal{V}$  is the POI set and  $\mathcal{E}_V$  is the edge set of POIs. Given a distance threshold  $\Delta D$  (set as 0.2 km), for any two POIs  $v_i \in \mathcal{V}$  and  $v_j \in \mathcal{V}$ , if the distance (here, the distance is computed by the Haversine formula)  $d_{ij}$  between them is smaller than  $\Delta D$  ( $d_{ij} < \Delta D$ ), there will be an edge  $e_{ij}$  from  $v_i$  to  $v_j$ , and otherwise none.

According to the definition of POI-POI graph, the location network among POIs is thereafter formed. Location network captures the geographical characteristics of users' check-in activities. Intuitively, if two POIs  $v_i$  and  $v_j$  are geographically close, they will have a strong correlation in location network. When we know a user has checked in  $v_i$ , we can conclude this user will also visit  $v_j$  next with high confidence. In experiments, we obtain the location network of Gowalla with 25,517 nodes and 210,325 edges. The parameter settings of our pretrained node2vec model are as follows:  $D_t = 128, l = 80, r = 10, k = 10, p = 1$ , and  $q = 0.5$ , where  $D_t$  is the dimension of the learned network feature vector.

**3.6. Location-Aware Social Recommendation.** The pretrained location network embeddings can be seen as the representations of POIs from geographical relations. Based on the assumption that nearby locations tend to be visited by similar users, that is, neighboring locations tend to have similar representations, we treat the learned embeddings as the prior of the latent POI factor in collaborative filtering model and modify the distribution of  $V$  as

$$p(V | Y, T, \sigma_V^2) = \prod_{v=1}^n \mathcal{N}(V_v | YT_v, \sigma_V^2 \mathbf{I}), \quad (17)$$

where  $T_v \in \mathbb{R}^{D_t}$  denotes the pretrained embedding of POI  $v$  from geographical relationships ( $D_t$  is the dimension of  $T_v$ ) and  $Y \in \mathbb{R}^{d \times D_t}$  is the weight matrix that transforms the learned POI representations from location network into the collaborative filtering latent space via  $YT_v$ .

As in SRank, the zero-mean spherical Gaussian distribution is placed on the weight matrix  $Y$ :

$$p(Y | \sigma_Y^2) = \prod_{y=1}^d \prod_{t=1}^{D_t} \mathcal{N}(Y_{yt} | 0, \sigma_Y^2), \quad (18)$$

Then, the posterior probability of SRank can be further modified as

$$\begin{aligned} & p(U, V, X, Y | \mathcal{D}_{\mathcal{R}}, S, T, \sigma_X^2, \sigma_Y^2, \sigma_U^2, \sigma_V^2) \\ & \propto p(\mathcal{D}_{\mathcal{R}} | U, V) p(U | X, S, \sigma_U^2) \cdot p(V | Y, T, \sigma_V^2) p(X | \sigma_X^2) p(Y | \sigma_Y^2) \\ & = \prod_{(u,i,j) \in \mathcal{D}_{\mathcal{R}}} \sigma(\hat{r}_{uij}) \times \prod_{u=1}^m \mathcal{N}(U_u | XS_u, \sigma_U^2 \mathbf{I}) \\ & \quad \times \prod_{v=1}^n \mathcal{N}(V_v | YT_v, \sigma_V^2 \mathbf{I}) \\ & \quad \times \prod_{x=1}^d \prod_{s=1}^{D_s} \mathcal{N}(X_{xs} | 0, \sigma_X^2) \times \prod_{y=1}^d \prod_{t=1}^{D_t} \mathcal{N}(Y_{yt} | 0, \sigma_Y^2). \end{aligned} \quad (19)$$

By computing the log form of equation (19) and keeping the hyperparameters fixed, the objective function of our final recommendation method Geo-SRank is achieved:

$$\begin{aligned} \text{Geo-SRank}(\mathcal{D}_{\mathcal{R}}, X, Y, U, V) = & - \sum_{u=1}^m \sum_{i \in \mathcal{I}_u^+} \sum_{j \in \mathcal{J} \setminus \mathcal{I}_u^+} \ln \sigma(\hat{r}_{uij}) \\ & + \frac{\lambda_U}{2} \sum_{u=1}^m (U_u - XS_u)^T \\ & \quad \cdot (U_u - XS_u) \\ & + \frac{\lambda_V}{2} \sum_{v=1}^n (V_v - YT_v)^T \\ & \quad \cdot (V_v - YT_v) \\ & + \frac{\lambda_X}{2} \|X\|_F^2 + \frac{\lambda_Y}{2} \|Y\|_F^2, \end{aligned} \quad (20)$$

where  $\lambda_Y = 1/\sigma_Y^2$  is the regularization parameter of  $Y$ . To find a local minimum of this objective function, gradient descent-based algorithm is utilized, and the latent factors can be updated by the following gradients:

$$\frac{\partial \text{Geo-SRank}}{\partial X} = -\lambda_U U U^T S^T + \lambda_U X S U^T S^T + \lambda_X X, \quad (21)$$

$$\frac{\partial \text{Geo-SRank}}{\partial Y} = -\lambda_V V V^T T^T + \lambda_V Y T V^T T^T + \lambda_Y Y, \quad (22)$$

$$\frac{\partial \text{Geo-SRank}}{\partial U_u} = -\frac{e^{-\hat{r}_{uij}}}{1 + e^{-\hat{r}_{uij}}} (V_i - V_j) + \lambda_U U_u - \lambda_U X S_u, \quad (23)$$

$$\frac{\partial \text{Geo-SRank}}{\partial V_i} = -\frac{e^{-\hat{r}_{uij}}}{1 + e^{-\hat{r}_{uij}}} U_u + \lambda_V V_i - \lambda_V Y T_i, \quad (24)$$

$$\frac{\partial \text{Geo-SRank}}{\partial V_j} = \frac{e^{-\hat{r}_{uij}}}{1 + e^{-\hat{r}_{uij}}} U_u + \lambda_V V_j - \lambda_V Y T_j. \quad (25)$$

Algorithm 1 presents the training process of our Geo-SRank method.

Require: the user-POI check-in matrix  $R$ , learning rate  $\eta$ , regularization parameters  $\lambda_U, \lambda_V, \lambda_X, \lambda_Y$ , and  $\lambda_r$ , the pretrained network embeddings  $S$  and  $T$ ;  
 Ensure: the latent factors  $U, V, X$ , and  $Y$ ;  
 (1) The initialization of  $U, V, X$ , and  $Y$ ;  
 (2) **while**  $(L(t) - L(t-1)) > \text{tolerate error}$  **do**  
 (3) Randomly select one training example  $(u, i, j)$  from  $\mathcal{D}_{\mathcal{R}}$ ;  
 (4)  $\hat{r}_{uij} \leftarrow \hat{r}_{ui} - \hat{r}_{uj}$ ;  
 (5) Calculate  $U_u$  by utilizing equation (23);  
 (6) Calculate  $X$  by utilizing equation (21);  
 (7) Calculate  $V_i$  by utilizing equation (24);  
 (8) Calculate  $V_j$  by utilizing equation (25);  
 (9) Calculate  $Y$  by utilizing equation (22);  
 (10) Calculate  $L(t)$  (the loss error in  $t$  step) by utilizing equation (20);  
 (11) **end while**  
 (12) **return**  $U, V, X$ , and  $Y$ ;

ALGORITHM 1: The training process of Geo-SRank.

**3.7. Computational Complexity.** The complexity of training the Geo-SRank method is mainly caused by computing the ranking loss (equation (20)) and the gradients of latent feature factors  $U, V, X$ , and  $Y$ . Assuming the average number of sampled triples is  $\bar{N}$ , the average dimension of pretrained social network embedding is  $\bar{s}$ , and the average dimension of pretrained location network embedding is  $\bar{t}$ , the complexity of computing equation (20) is  $O(\bar{N}d + m\bar{s}d + n\bar{t}d)$ . Since in our experiments the dimensions of the feature factors ( $d, \bar{s}$ , and  $\bar{t}$ ) are set as very small values, the computing complexity is mainly determined by the sampled training triples  $\bar{N}$ . The computation of the objective function of Geo-SRank is very fast and linear with respect to the sampled triple number. The complexities of computing the gradients  $\partial L/\partial U$  and  $\partial L/\partial X$  are  $O(\bar{N}d\bar{s})$ . The complexities of  $\partial L/\partial V$  and  $\partial L/\partial Y$  are  $O(\bar{N}d\bar{t})$ . Both of these two complexities are also linear with respect to the sampled triple number.

**3.8. Applications in Cold-Start Situations.** The advantage of our feature-based recommendation method Geo-SRank is that it can be easily extended to the cold-start situations, i.e., there are no training data for the users/items in history. In the user cold-start situation, we can make predictions for users that no one has check-in records by using the social network representations  $\hat{r}_{uv} = (XS_u)^T V_v$ , where pure collaborative filtering models have trouble in learning the latent user factors. Here, the weight matrix  $X$  can be learned by known users and  $S_u$  is the pretrained network embeddings. In the item cold-start situation, we can make predictions for cold-start items by using the location network representations  $\hat{r}_{uv} = U_u^T (YT_v)$ , where the weight matrix  $Y$  is learned by known items and  $T_v$  is the pretrained embeddings of items.

## 4. Experiments

In this section, several experiments on two real-world datasets are conducted to evaluate the recommendation performance of our feature-based method. To test the scalability of our method to the cold-start problem, we also conduct experiments on the situations with cold-start settings.

**4.1. Datasets and Evaluation Metrics.** In our experiments, two real-world datasets Gowalla and Yelp [12] are exploited to evaluate our proposed method (Table 1). Gowalla was launched in 2007 and closed in 2012, which provides the location and social services that enable users to check in at places and make friends based on where they have visited. The Gowalla dataset (<https://snap.stanford.edu/data/loc-gowalla.html>) we used was crawled by its API from February 2009 to October 2010. By filtering out those users with fewer than 15 check-in POIs and those POIs with fewer than 10 visitors, we can reach our final dataset, which contains 25,379 users, 32,623 POIs, 1,395,856 check-ins, and 118,717 social ties. Yelp is a local-search service powered by crowdsourced review forum, which publishes crowdsourced reviews about local businesses, as well as the online reservation service. Yelp was founded in 2004 and grew very quickly. By the end of 2017, Yelp has 148 million reviews. The dataset we used was collected by Liu et al. [12], which contains 30,887 users, 18,942 businesses, 561,746 check-ins, and 265,533 social ties.

To measure the performance of our recommendation approach, we exploit three popular measures Precision@ $k$ , Recall@ $k$ , and mean average precision (MAP) [12] as the evaluation metrics. In this Top- $k$  item recommendation task, Precision@ $k$  measures how many relevant POIs can be predicted in the recommendation list. Here, the relevant POIs refer to the ones that have been labeled as positive in the dataset. Recall@ $k$  denotes how many relevant POIs can be predicted among the total relevant POIs in the dataset. The formal definitions of these two metrics are as follows:

$$\text{Precision@}k = \frac{1}{M} \sum_{u=1}^M \frac{|L^k(u) \cap L^T(u)|}{k}, \quad (26)$$

$$\text{Recall@}k = \frac{1}{M} \sum_{u=1}^M \frac{|L^k(u) \cap L^T(u)|}{|L^T(u)|}, \quad (27)$$

where  $M$  is the number of being tested users and  $L^T(u)$  is the POI set that have been checked in by  $u$  in the test data.  $L^k(u)$  represents the recommended Top- $k$  POI list for user  $u$  ( $k$  is

TABLE 1: Statistics of the training data of Gowalla and Yelp.

Statistics	Yelp	Gowalla
Check-in sparsity	99.903	99.906
No. of users	30,887	18,737
No. of POIs	18,942	32,495
No. of check-ins	561,746	566,791
Avg. No. of users per POI	0.6	0.57
Avg. No. of POIs per user	1.6	1.73

the size of this list). In this work, the list size  $k$  is set as 5. MAP measures the average precision (AP) over all users, which is computed by

$$\begin{aligned} \text{MAP} &= \frac{1}{M} \sum_{u=1}^m \text{AP} \\ &= \frac{1}{M} \sum_{u=1}^m \left( \frac{1}{k} \sum_{i=1}^k \text{Precision}@i \right), \end{aligned} \quad (28)$$

where  $\text{Precision}@i$  refers to the precision for Top- $i$  recommendations.

**4.2. Data Analysis.** To better understand the users' visiting behaviors in location-based social networks, we further investigate the social influence and geographical characteristics in Gowalla and Yelp and try to investigate the following questions: (1) Are users with friend relationships more likely to have similar check-in activities? (2) Are POIs with neighboring relationships more likely to be visited by the same set of users? To answer these two questions, we need to first define a method to measure the similarity between two users.

Let  $A(u)$  denote the POI set that has been visited by user  $u$  in history. The similarity between user  $u$  and  $v$  can be computed as

$$x_{uv} = \frac{\sum_{i \in A(u) \cap A(v)} (1/\log(1 + |A(i)|))}{\sqrt{|A(u)||A(v)|}}, \quad (29)$$

where  $|A(x)|$  represents the size of the POI set  $A(x)$  and  $1/\log(1 + |A(u)|)$  is the item that punishes the influence of popular POIs.

For each user  $u$ , two similarities are computed, that is, (1) the average similarity between user  $u$  and his/her friends ( $x_f(u)$ ) and (2) the average similarity between user  $u$  and randomly selected users. For fairness, we choose the same number of users with their friend relationships. Figure 4 plots the kernel-smoothing density estimations based on the vectors  $x_f$  and  $x_r$ , from which we can observe that  $x_r$  has smaller similarity values than  $x_f$ . This result positively answers our first question, that is, users with friend relationships share more common interests than those without.

To answer the second question, the measurement of the similarity between two POIs needs to be predefined. By denoting  $B(i)$  as the user set that has checked in location  $i$ , the similarity between POI  $i$  and  $j$  is arrived at

$$y_{ij} = \frac{\sum_{u \in B(i) \cap B(j)} (1/\log(1 + |B(u)|))}{\sqrt{|B(i)||B(j)|}}. \quad (30)$$

Based on this similarity, Figure 5 shows how the similarity of randomly chosen POI pairs varies with the increase of geographical distance, where we can find that, with the increase of the distance, the mean value of the POI similarity decreases in both Yelp and Gowalla data. This result demonstrates the positive relationship between the POI similarity and geographical distance and provides us a positive answer to the second question: POIs with neighboring relationships are inclined to be visited by similar users.

**4.3. Performance Comparison.** To evaluate the recommendation performance of our method, we conduct comparisons with several related recommendation methods. As the matrix factorization- (MF-) based method has been proved to have state-of-the-art performance, we mainly compare against the MF-based collaborative filtering models and the related ranking-based method (most of these methods are implemented in MyMediaLite [63]):

- (i) MostPopular: this is a basic recommendation method, which ranks the POIs according to how many users have visited them.
- (ii) WRMF [45]: this method is proposed for the one-class collaborative filtering problem and provides a solution by giving small weights to negative samples and large weights to positive samples.
- (iii) GeoMF [4]: this is a state-of-the-art POI recommendation method, which extends the WRMF method by augmenting the user and POI latent factors.
- (iv) BPRMF [7]: this item recommendation method is proposed for implicit feedback, which learns the user interests from their pair-wise item preferences. This is the state-of-the-art POI recommendation method, which is only based on check-in matrix, without utilizing any context information.
- (v) Rank-GeoFM [5]: this is a state-of-the-art POI recommendation method that develops a ranking-based matrix factorization model to learn the user's preferences and include the geographical influence simultaneously.
- (vi) IReMF [47]: this is another state-of-the-art POI recommendation method that develops a weighted matrix factorization to model the geographical influence on POIs.
- (vii) SRank: this is our proposed social-aware POI ranking method, which incorporates the social network embeddings into the pair-wise ranking framework. Compared with Geo-SRank, the only difference is that it does not consider the location network information.

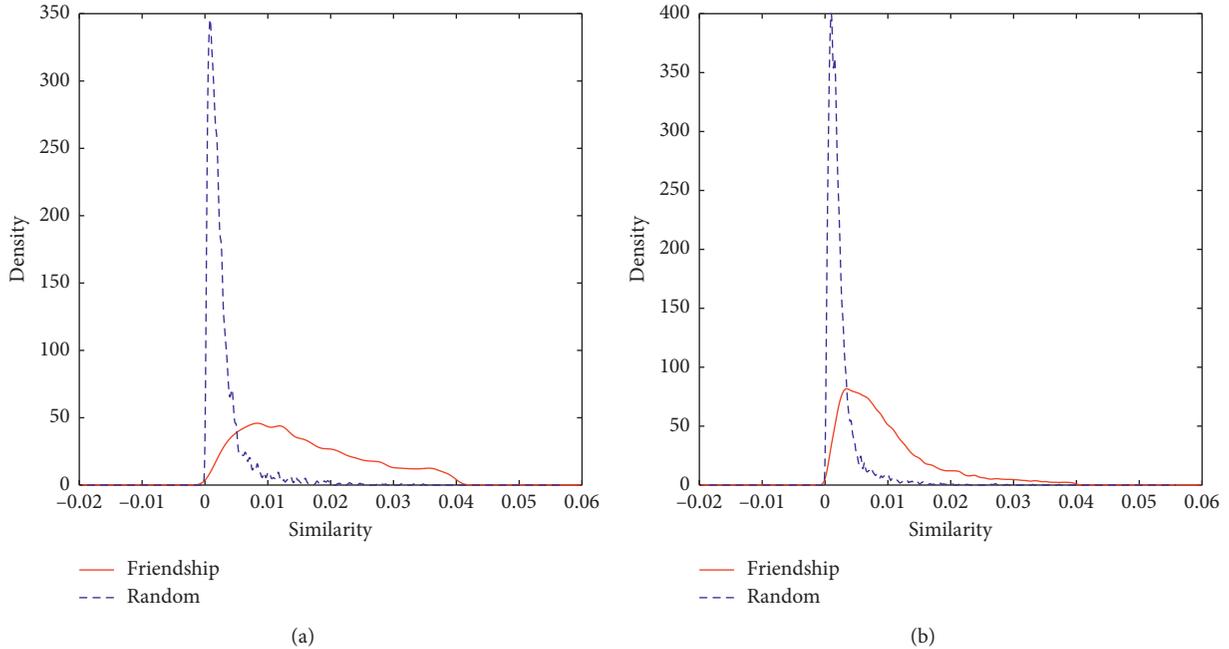


FIGURE 4: Density estimation of users' similarity on (a) Gowalla and (b) Yelp.

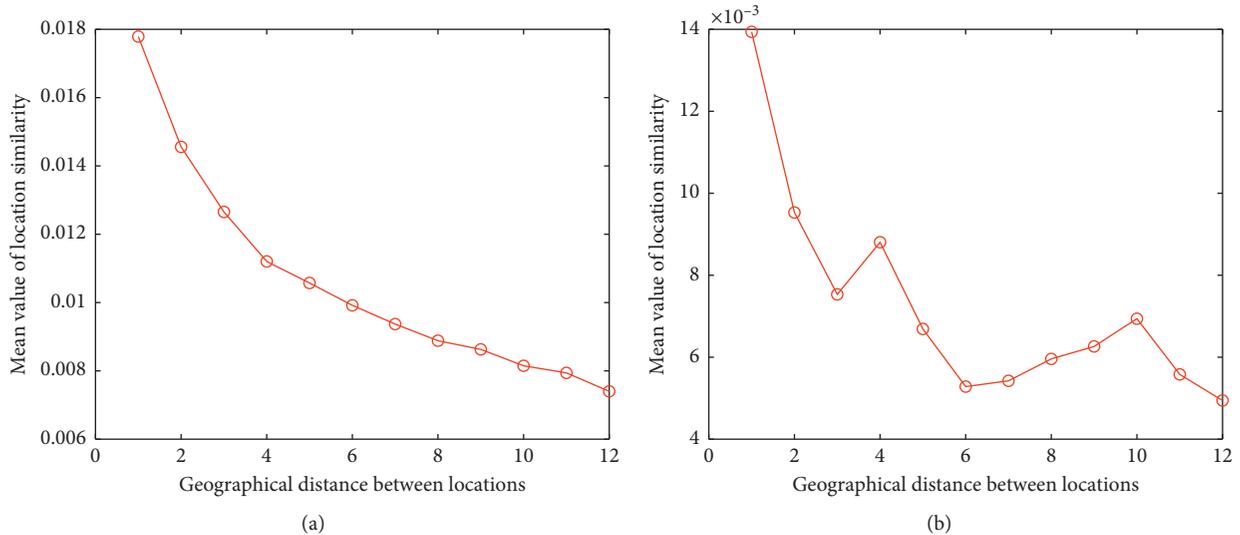


FIGURE 5: The similarity between POIs varies with geographical distance: (a) Yelp and (b) Gowalla.

(viii) Geo-SRank: this is our final recommendation method, which further extends the SRank method by utilizing the representations learned from a predefined location network.

In experiments, we evaluate our method in both the non-cold-start and cold-start situations, respectively. In the non-cold-start situation, we randomly select 70% user-POI actions for training, 10% user-POI pairs for tuning, and 20% for testing. In the cold-start situation, we use the same dataset for testing and randomly select part of the training data for training (from 10% to 80%). To ensure the validity of these data, the data selection process is conducted five times

independently. In these two situations, the regularization parameters of  $U$ ,  $V$ ,  $X$ , and  $Y$  are set as  $\lambda_U = \lambda_{V_i} = 0.0025$  and  $\lambda_{V_j} = \lambda_X = \lambda_Y = 0.00025$ . The dimensions of the latent factors  $U$  and  $V$  are set as  $d = 10$ .

The experimental results evaluated on Gowalla and Yelp with non-cold-start settings have been shown in Tables 2 and 3, from which we have the following observations: (1) Our Geo-SRank method achieves the best performance over all the baseline methods on both datasets. This demonstrates the superiority of the Geo-SRank method, that is, the effectiveness of our learned network embeddings and the usefulness of the social influence and the geographical

TABLE 2: Top- $k$  recommendation performance on Gowalla ( $\Delta$  denotes the improvement of Geo-SRank over other methods).

Overall performance comparison (Gowalla)						
$d = 10, k = 5$						
	Precision@ $k$	$\Delta\%$	Recall@ $k$	$\Delta\%$	MAP	$\Delta\%$
MostPopular	0.01796	188	0.01159	217	0.00597	253
WRMF	0.04908	5.7	0.0334	10	0.02004	5.2
GeoMF	0.05104	1.64	0.03505	4.9	0.02085	1.1
BPRMF	0.0472	9.91	0.03254	12.9	0.01904	10.8
Rank-GeoFM	0.05139	0.9	0.03596	2.2	0.02097	0.6
IRenMF	0.05083	2.06	0.03485	5.5	0.02061	2.3
SRank	0.0496	4.59	0.03315	10.9	0.01983	6.4
Geo-SRank	0.05188	—	0.03677	—	0.0211	—

TABLE 3: Top- $k$  recommendation performance on Yelp ( $\Delta$  denotes the improvement of Geo-SRank over other methods).

Overall performance comparison (Yelp)						
$d = 10, k = 5$						
	Precision@ $k$	$\Delta\%$	Recall@ $k$	$\Delta\%$	MAP	$\Delta\%$
MostPopular	0.00852	201	0.0095	172	0.00444	180
WRMF	0.02384	7.9	0.02354	9.77	0.01205	3.3
GeoMF	0.02464	4.4	0.02446	5.6	0.01229	1.3
BPRMF	0.01892	35.9	0.01855	39.2	0.0085	46.4
Rank-GeoFM	0.02521	2	0.02536	1.8	0.01234	0.8
IRenMF	0.02440	5.4	0.02430	6.3	0.01216	2.3
SRank	0.02448	5.1	0.02515	2.7	0.01221	1.9
Geo-SRank	0.02573	—	0.02584	—	0.01245	—

information. (2) Compared with basic recommendation methods (BPRMF and WRMF), the methods (i.e., GeoMF, Rank-GeoFM, IRenMF, and Geo-SRank) that consider the geographical information can significantly improve the POI recommendation performance. This result demonstrates the usefulness of the geographical information in helping users find the right POIs. (3) The method (i.e., SRank and Geo-SRank) that considers the social influence can perform better than BPRMF and WRMF, demonstrating the importance of the social influence. Geo-SRank can achieve the best performance over all the baseline methods, again, demonstrating the usefulness of both geographical information and social influence, and we can get a better result by considering them simultaneously. (4) MF-based methods can achieve a reasonable result with low-dimensional latent feature vectors, indicating the effectiveness of MF-based methods in the POI recommendation task. In experiments, we set the dimension of all the latent factors as 10. (5) From the results, we can also observe that the accuracy (i.e., Precision, Recall, and MAP) of all the POI recommendation methods is relatively low (less than 7%). One of the important reasons might be that the check-in data in LBSNs is very sparse (the density of check-in matrix is usually less than 0.5%), and thus, the recommendation methods suffer from the data scarcity problem. Second, compared with traditional movie rating or review data, the check-in data may not fully reflect the user’s interests. For example, a user might check-in at a location where he/she is waiting for a friend or just passing by, which is not what he/she really likes.

4.4. *Evaluation on Cold-Start Settings.* To evaluate the recommendation performance of our method in cold-start situations, we further conduct experiments with cold-start settings. In this experiment, we randomly select part of the training data for training. The training ratio is changed from 10% to 80%. In experiments, we compare our method with the baseline method BPRMF and the state-of-the-art POI recommendation method GeoMF. As BPRMF does not consider any other context information, it will use randomly generated feature values for new uses and items. But for GeoMF, as it not only exploits the basic user-item interactions to make predictions but also considers the geographical context, it will employ the location information to alleviate the cold-start problem. The experimental results with different training ratios are shown in Figure 6, from which we can find that, with the increase of the training data, the performance of these three methods both increases, which indicates that, with more training data, we can make our model more accurate. From this result, we can also find that our feature-based method can work well in these two datasets, which demonstrates that treating the learned network embeddings as prior knowledge of the latent factors is effective and can further improve the recommendations for those cold-start users. A larger margin between BPRMF and Geo-SRank can be found in Yelp than Gowalla, which is consistent with the results experimented on the non-cold-start data.

4.5. *The Influence of the Recommendation List.* In POI recommendation, the system usually allows users to specify the POI numbers that shall be returned to her/him to avoid being overwhelmed by irrelevant POIs. Appropriate POI number helps to improve the user experience in recommender systems. Therefore, we further investigate the performance of our method with varying the number of the recommended POIs. The Top- $k$  recommendation result is shown in Figure 7, where only the Top- $k$  most relevant POIs are considered. From this result, we can observe that, with the increase of the recommendation list, the hits of the relevant POIs are increasing, hence resulting in the increase of Recall@ $k$  and the decrease of Precision@ $k$ . The increase of Recall@ $k$  means the predictions can cover more relevant POIs, that is, the capability of the recommendation method to retrieve relevant POIs is increasing. The decrease of Precision@ $k$  represents the predictions can only cover less relevant POIs, i.e., the capability of the recommendation method to retrieve correct POIs is decreasing.

4.6. *The Influence of Latent Factor Dimensions.* The dimension of the latent factors controls how much information can be used to represent users and items (In this experiment, we set  $U$  and  $V$  as the same dimension.). If the dimension is too small, the learned features will not be sufficient to represent the user and POI information. If the dimension is too large, not only will the learned features appear to be duplicated, but also the complexity of the algorithm will increase. Appropriate feature dimension plays a crucial role in our algorithm.

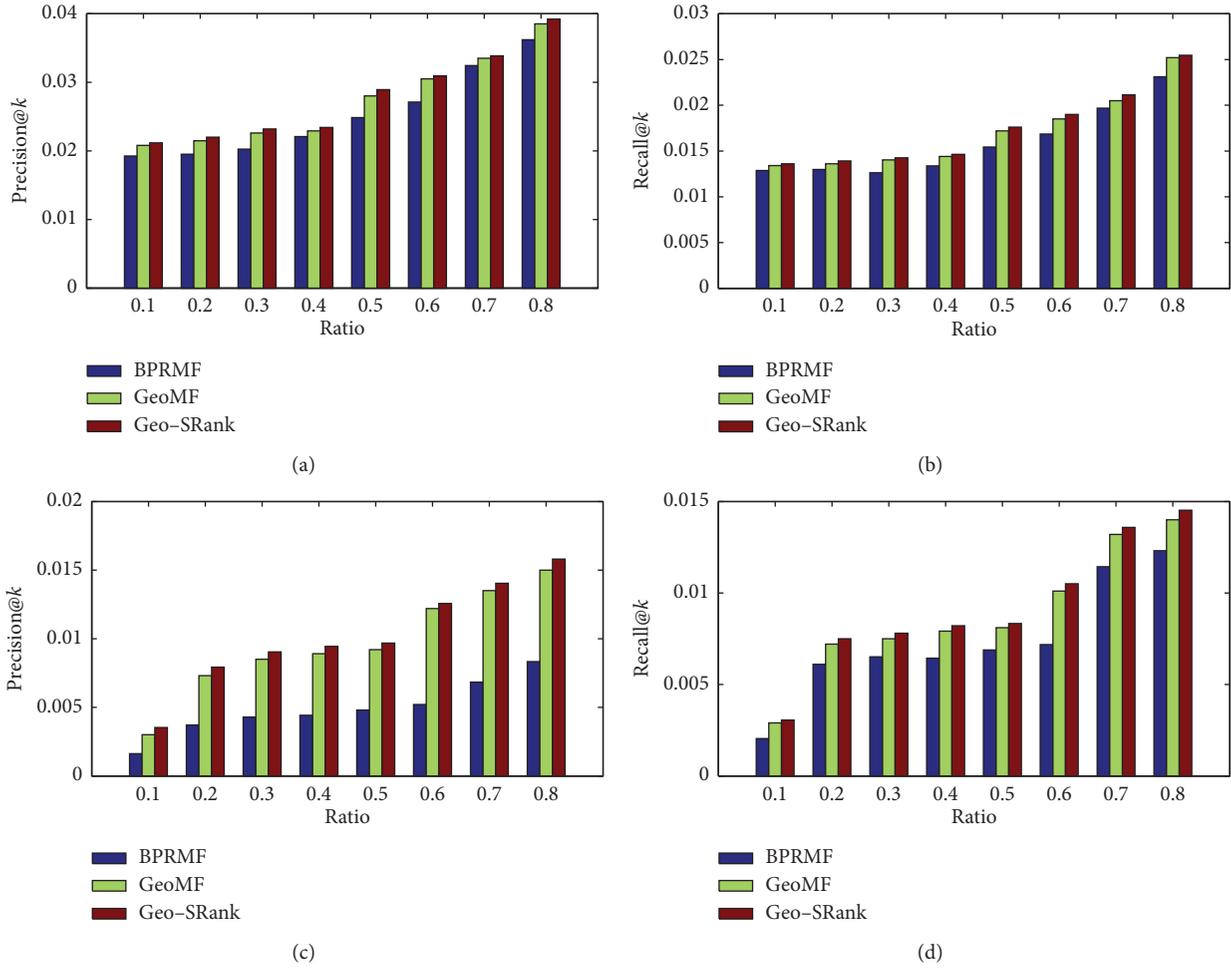


FIGURE 6: Experimental results with different training ratios ( $k=5$ ). (a, b) Gowalla and (c, d) Yelp.

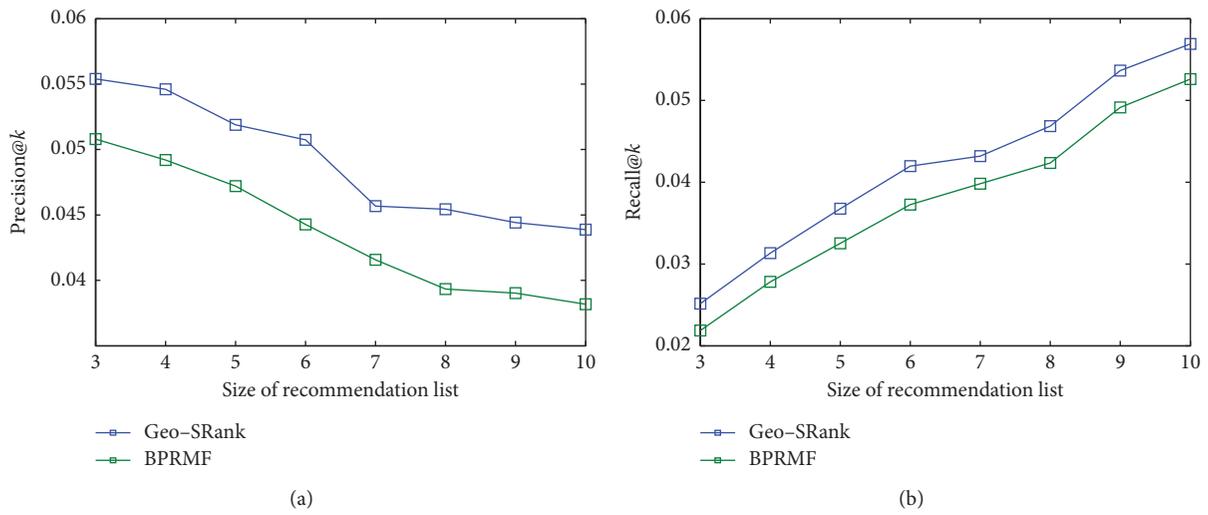


FIGURE 7: Continued.

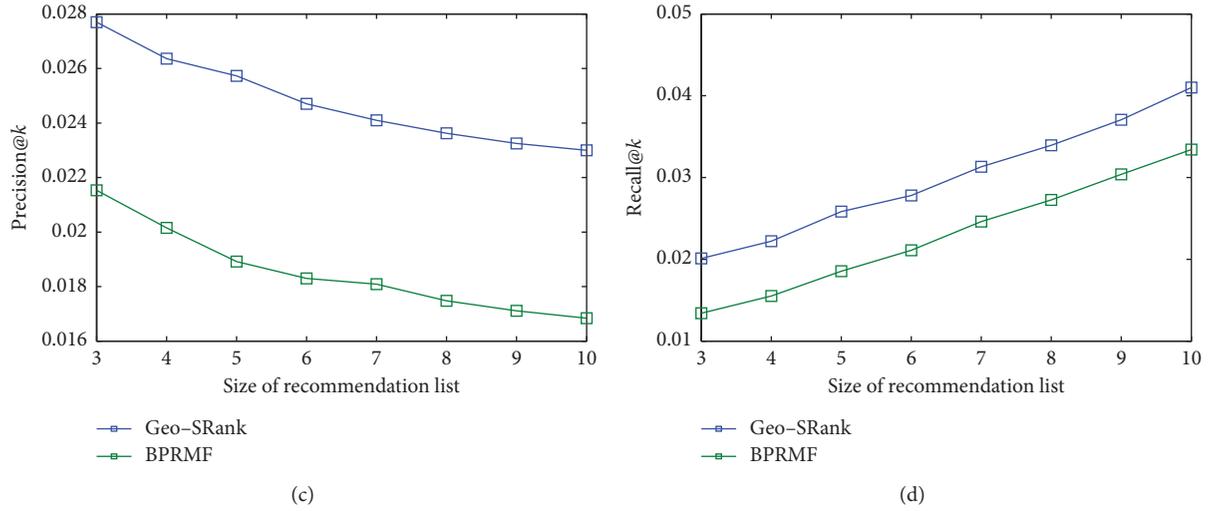


FIGURE 7: Experimental results with different recommendation numbers. (a, b) Gowalla and (c, d) Yelp.

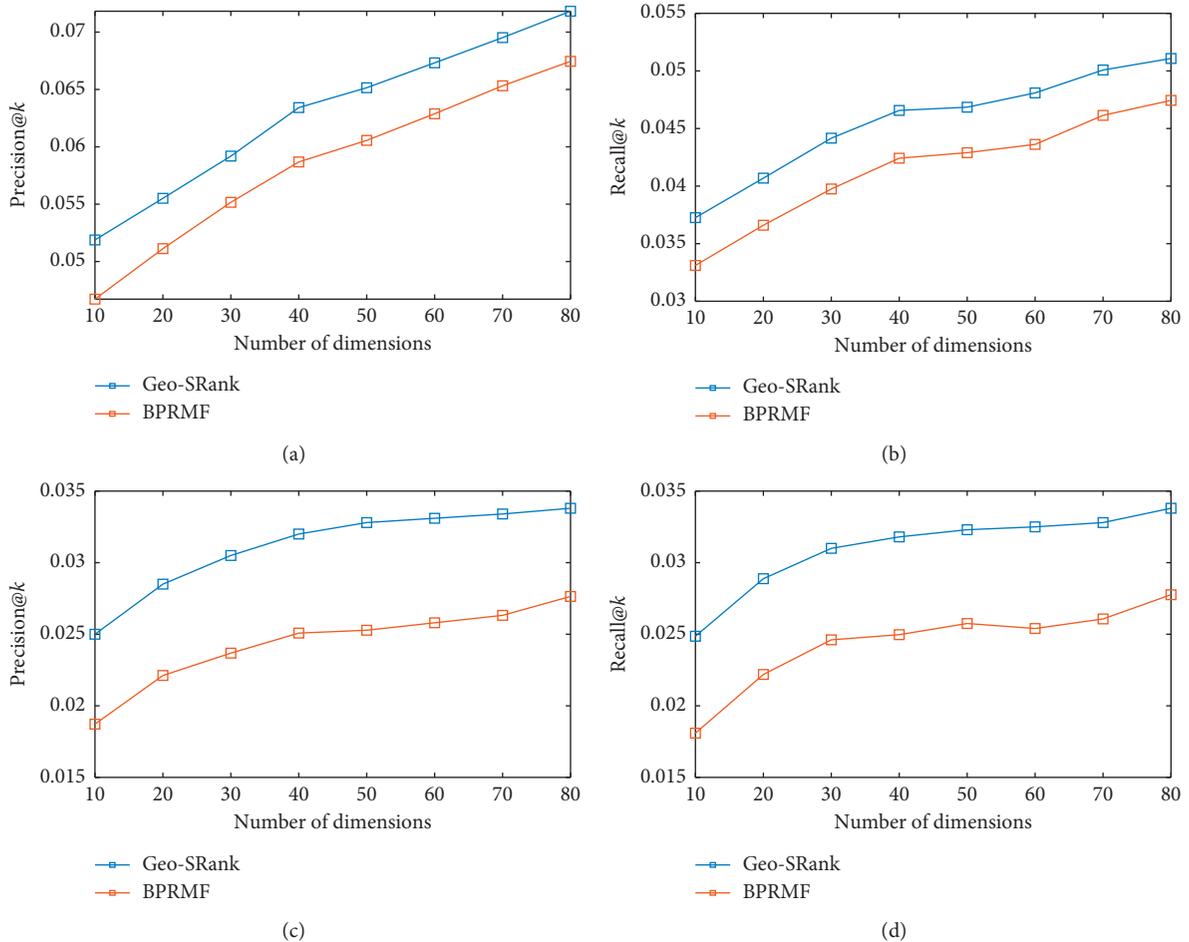


FIGURE 8: The experimental results with different latent factor dimensions. (a, b) Gowalla and (c, d) Yelp.

Figure 8 shows the ranking performance of our method with the feature dimension changes from 10 to 80, from which we can observe that, with the feature dimension increases, the ranking performance of BPRMF and Geo-

SRank also increases gradually. But when its value surpasses a certain threshold (such as 50), the increase begins to slow down, which indicates that when the dimension is big enough, there will be enough information for extracted

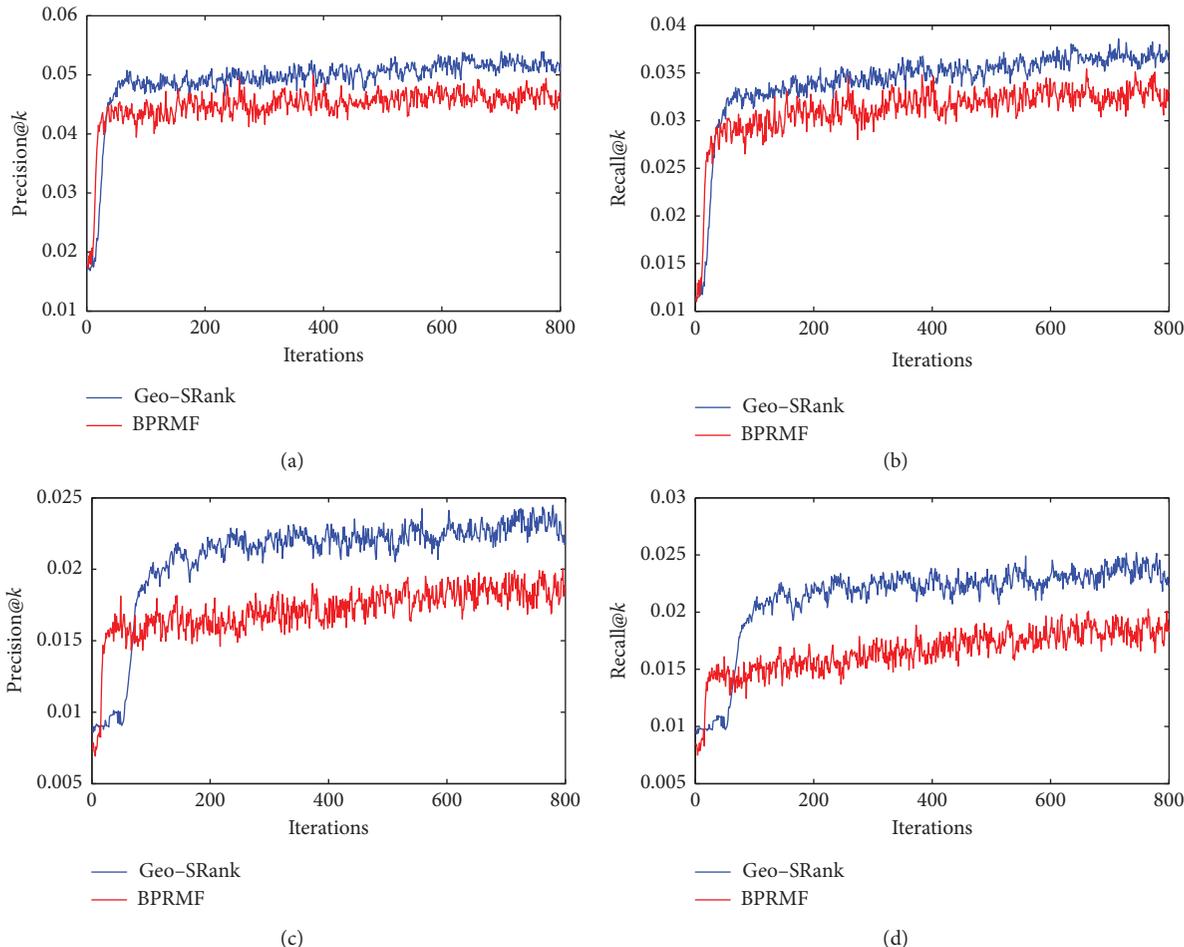


FIGURE 9: The experimental results with training iterations on the cold-start data. (a) Precision@ $k$  on Gowalla, (b) Recall@ $k$  on Gowalla, (c) Precision@ $k$  on Yelp, and (d) Recall@ $k$  on Yelp.

features to represent the social and location networks. In our experiments, we find that, with very low dimension of these latent factors, the recommendation can achieve comparable results. To reduce the computational complexity of our method, in experiments both user and POI dimensions are set as  $d = 10$ .

**4.7. Convergence Analysis.** To demonstrate the training efficiency of our feature-based recommendation method, we show the evaluation results with increasing training iterations in Figure 9. In this experiment, the learning rates of BPRMF and Geo-SRank are both set as 0.05. From this result, we can observe that, with the increase of the iterations, the Geo-SRank method can converge (due to the stochastic gradient descent algorithm we used, after convergence these two methods still fluctuate around a certain value) very fast in the Gowalla data, which almost converges within the same iterations with BPRMF (after about 20 iterations). But it converges a little slow in Yelp data (about 100 iterations), and BPRMF can also converge within 20 iterations. The reason of this result is that we introduce two external feature vectors in our method, and the increased random variables slow down the converge rate. From this

result, we can also find our method can perform better than BPRMF throughout the whole learning process, which proved the effectiveness of these extracted network embeddings.

## 5. Conclusions and Future Work

In this work, we focused on how to make accurate POI recommendations in LBSNs and proposed a location-aware social recommendation method with pretrained network embeddings. To utilize the social relation and geographical information, we pretrained the network embedding method node2vec to learn the representations of users and POIs from the users' social network and the predefined location network, respectively. To incorporate the learned features into collaborative filtering model, we extended user and POI latent feature factors in a pair-wise ranking framework, where the learned embeddings are treated as their prior knowledge. Experimental results on two real-world datasets demonstrate the advantages of our method in the both non-cold-start and cold-start settings. Although we only tested our method on Gowalla and Yelp, Geo-SRank can also be applied to other real-world settings. For example, when a

user planning a trip on Mafengwo (<https://www.mafengwo.cn/>) (a famous communication platform for travellers in China), he/she may want to ask his/her friends first and then pick out the POIs that have been visited by the users that have similar interests with him. Similarly, we can also help users to find out the restaurants that they want to have dinner with friends. In the above cases, both the social influence and geographical information help us make the right choices.

This work mainly investigates how to utilize the pre-trained representations from social relation and location networks, but the check-in sequence is not considered. In real life, the users' check-in sequence is also an important aspect of their interest, for example, users always like to go to the playground and then go to the supermarket for shopping. In future, we plan to leverage the check-in sequence information to further enhance the POI recommendation performance.

## Data Availability

The data supporting this study are from previously reported studies and datasets, which have been cited. The processed data are available from the corresponding author upon request.

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## Acknowledgments

This work was supported by the National Natural Science Foundation of China (nos. 61602282, 61772321, and 71301086), China Postdoctoral Science Foundation (no. 2016M602181), and the Innovation Foundation of Science and Technology Development Center of Ministry of Education and New H3C Group (no. 2017A15047).

## References

- [1] M. Ye, P. Yin, W.-C. Lee, and D.-L. Lee, "Exploiting geographical influence for collaborative point-of-interest recommendation," in *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 325–334, ACM, New York, USA, July 2011.
- [2] C. Cheng, H. Yang, I. King, and M. R. Lyu, "Fused matrix factorization with geographical and social influence in location-based social networks," in *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence*, vol. 12, pp. 17–23, Toronto, Canada, July 2012.
- [3] J.-D. Zhang and C.-Y. Chow, "iGSLR: personalized geo-social location recommendation: a kernel density estimation approach," in *Proceedings of the 21st ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pp. 334–343, ACM, Orlando, FL, USA, November 2013.
- [4] D. Lian, C. Zhao, X. Xie, G. Sun, E. Chen, and Y. Rui, "GeoMF: joint geographical modeling and matrix factorization for point-of-interest recommendation," in *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 831–840, ACM, New York, NY, USA, August 2014.
- [5] X. Li, G. Cong, X.-L. Li, T.-A. N. Pham, and S. Krishnaswamy, "Rank-GeoFM: a ranking based geographical factorization method for point of interest recommendation," in *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 433–442, ACM, Santiago, Chile, August 2015.
- [6] H. Ying, L. Chen, Y. Xiong, and J. Wu, "PGRank: personalized geographical ranking for point-of-interest recommendation," in *Proceedings of the 25th International Conference Companion on World Wide Web*, pp. 137–138, International World Wide Web Conferences Steering Committee, Montreal, Canada, April 2016.
- [7] S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme, "BPR: bayesian personalized ranking from implicit feedback," in *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, pp. 452–461, AUAI Press, Montreal, QC, Canada, June 2009.
- [8] M. Hang, I. Pytlarz, and J. Neville, "Exploring student check-in behavior for improved point-of-interest prediction," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 321–330, ACM, London, UK, August 2018.
- [9] P. Wang, J. Guo, Y. Lan, J. Xu, S. Wan, and X. Cheng, "Learning hierarchical representation model for nextbasket recommendation," in *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 403–412, ACM, Santiago, Chile, August 2015.
- [10] W. X. Zhao, S. Li, Y. He, E. Y. Chang, J.-R. Wen, and X. Li, "Connecting social media to e-commerce: cold-start product recommendation using microblogging information," *IEEE Transactions on Knowledge and Data Engineering*, vol. 28, no. 5, pp. 1147–1159, 2016.
- [11] A. Grover and J. Leskovec, "Node2vec: scalable feature learning for networks," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 855–864, ACM, San Francisco, CA, USA, August 2016.
- [12] Y. Liu, T.-A. N. Pham, G. Cong, and Q. Yuan, "An experimental evaluation of point-of-interest recommendation in location-based social networks," *Proceedings of the VLDB Endowment*, vol. 10, no. 10, pp. 1010–1021, 2017.
- [13] M. Hong and J. J. Jung, "Multi-sided recommendation based on social tensor factorization," *Information Sciences*, vol. 447, pp. 140–156, 2018.
- [14] Z. Zhang, Y. Liu, Z. Zhang, and B. Shen, "Fused matrix factorization with multi-tag, Social and geographical influences for POI recommendation," *World Wide Web*, vol. 22, no. 3, pp. 1135–1150, 2019.
- [15] Y. Yu and X. Chen, "A survey of point-of-interest recommendation in location-based social networks," in *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence Workshops*, vol. 130, Austin, TX, USA, January 2015.
- [16] H. Yin, B. Cui, L. Chen, Z. Hu, and C. Zhang, "Modeling location-based user rating profiles for personalized recommendation," *ACM Transactions on Knowledge Discovery from Data (TKDD)*, vol. 9, no. 3, pp. 1–41, 2015.
- [17] S.-B. Cho, "Exploiting machine learning techniques for location recognition and prediction with smartphone logs," *Neurocomputing*, vol. 176, pp. 98–106, 2016.

- [18] E. Cho, S. A. Myers, and J. Leskovec, "Friendship and mobility: user movement in location-based social networks," in *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1082–1090, ACM, San Diego, CA, USA, August 2011.
- [19] J.-D. Zhang, C.-Y. Chow, and Y. Li, "iGeoRec: a personalized and efficient geographical location recommendation framework," *IEEE Transactions on Services Computing*, vol. 8, no. 5, pp. 701–714, 2015.
- [20] H. Li, Y. Ge, R. Hong, and H. Zhu, "Point-of-interest recommendations: learning potential check-ins from friends," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 975–984, San Diego, CA, USA, August 2016.
- [21] B. Liu, H. Xiong, S. Papadimitriou, Y. Fu, and Z. Yao, "A general geographical probabilistic factor model for point of interest recommendation," *IEEE Transactions on Knowledge and Data Engineering*, vol. 27, no. 5, pp. 1167–1179, 2015.
- [22] H. Wang, M. Terrovitis, and N. Mamoulis, "Location recommendation in location-based social networks using user check-in data," in *Proceedings of the 21st ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pp. 374–383, ACM, Orlando, FL, USA, November 2013.
- [23] H. Wang, H. Shen, W. Ouyang, and X. Cheng, "Exploiting poi-specific geographical influence for point-of-interest recommendation," in *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, pp. 3877–3883, Stockholm, Sweden, July 2018.
- [24] C. Ma, Y. Zhang, Q. Wang, and X. Liu, "Point-of-interest recommendation: exploiting self-attentive autoencoders with neighbor-aware influence," in *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, pp. 697–706, ACM, Turin, Italy, October 2018.
- [25] L. Guo, H. Jiang, and X. Wang, "Location regularization-based poi recommendation in location-based social networks," *Information*, vol. 9, no. 4, p. 85, 2018.
- [26] B. Chang, Y. Park, D. Park, S. Kim, and J. Kang, "Content-aware hierarchical point-of-interest embedding model for successive poi recommendation," in *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, pp. 3301–3307, Stockholm, Sweden, July 2018.
- [27] B. Zimba, S. Chibuta, D. Chisanga, F. Banda, and J. Phiri, "Point of interest recommendation methods in location based social networks: traveling to a new geographical region," 2017, <https://arxiv.org/abs/1711.09471>.
- [28] H. Gao, J. Tang, X. Hu, and H. Liu, "Content-aware point of interest recommendation on location-based social networks," in *Proceedings of the 29th AAAI Conference on Artificial Intelligence*, pp. 1721–1727, Austin, TX, USA, January 2015.
- [29] J.-B. Griesner, T. Abdessalem, and H. Naacke, "Poi recommendation: towards fused matrix factorization with geographical and temporal influences," in *Proceedings of the 9th ACM Conference on Recommender Systems*, pp. 301–304, ACM, Vienna, Austria, September 2015.
- [30] J. Zeng, Y. Li, F. Li, X. He, and J. Wen, "Time-slot-based point of interest recommendation on location-based social network," *International Journal of Internet Manufacturing and Services*, vol. 5, no. 2-3, pp. 157–168, 2018.
- [31] B. Liu and H. Xiong, "Point-of-interest recommendation in location based social networks with topic and location awareness," in *Proceedings of the 2013 SIAM International Conference on Data Mining*, pp. 396–404, Austin, TX, USA, May 2013.
- [32] D. Lian, Y. Ge, F. Zhang et al., "Content-aware collaborative filtering for location recommendation based on human mobility data," in *Proceedings of the 2015 IEEE International Conference on Data Mining (ICDM)*, pp. 261–270, IEEE, Atlantic City, NJ, USA, November 2015.
- [33] H. Gao, J. Tang, X. Hu, and H. Liu, "Exploring temporal effects for location recommendation on location-based social networks," in *Proceedings of the 7th ACM Conference on Recommender Systems*, pp. 93–100, ACM, Hong Kong, China, October 2013.
- [34] T. Qian, B. Liu, Q. V. H. Nguyen, and H. Yin, "Spatiotemporal representation learning for translation-based poi recommendation," *ACM Transactions on Information Systems*, vol. 37, no. 2, pp. 1–24, 2019.
- [35] J.-D. Zhang, C.-Y. Chow, and Y. Li, "Lore: exploiting sequential influence for location recommendations," in *Proceedings of the 22nd ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pp. 103–112, ACM, Dallas, TX, USA, November 2014.
- [36] J.-D. Zhang and C.-Y. Chow, "GeoSoCa: exploiting geographical, social and categorical correlations for point-of-interest recommendations," in *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 443–452, ACM, Santiago, Chile, August 2015.
- [37] D. Lian, Y. Ge, F. Zhang et al., "Scalable content-aware collaborative filtering for location recommendation," *IEEE Transactions on Knowledge and Data Engineering*, vol. 30, no. 6, pp. 1122–1135, 2018.
- [38] C. Yang, M. Sun, W. X. Zhao, Z. Liu, and E. Y. Chang, "A neural network approach to jointly modeling social networks and mobile trajectories," *ACM Transactions on Information Systems (TOIS)*, vol. 35, no. 4, pp. 1–28, 2017.
- [39] Y. Hu, Y. Koren, and C. Volinsky, "Collaborative filtering for implicit feedback datasets," in *Proceedings of the 2008 8th IEEE International Conference on Data Mining*, pp. 263–272, IEEE, Pisa, Italy, December 2008.
- [40] L. Guo, J. Ma, H.-R. Jiang, Z.-M. Chen, and C.-M. Xing, "Social trust aware item recommendation for implicit feedback," *Journal of Computer Science and Technology*, vol. 30, no. 5, pp. 1039–1053, 2015.
- [41] Y. Ren, D. Ji, and H. Zhang, "Positive unlabeled learning for deceptive reviews detection," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 488–498, Doha, Qatar, October 2014.
- [42] H. Li, B. Liu, A. Mukherjee, and J. Shao, "Spotting fake reviews using positive-unlabeled learning," *Computación Y Sistemas*, vol. 18, no. 3, pp. 467–475, 2014.
- [43] C. Elkan and K. Noto, "Learning classifiers from only positive and unlabeled data," in *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 213–220, ACM, Las Vegas, NV, USA, August 2008.
- [44] X.-L. Li, P. S. Yu, B. Liu, and S.-K. Ng, "Positive unlabeled learning for data stream classification," in *Proceedings of the 2009 SIAM International Conference on Data Mining*, pp. 259–270, Sparks, NV, USA, April 2009.
- [45] R. Pan, Y. Zhou, B. Cao et al., "One-class collaborative filtering," in *Proceedings of the 2008 Eighth IEEE International Conference on Data Mining*, pp. 502–511, IEEE, Pisa, Italy, December 2008.
- [46] D. Lian, K. Zheng, Y. Ge, L. Cao, E. Chen, and X. Xie, "GeoMF++: scalable location recommendation via joint geographical modeling and matrix factorization," *ACM*

- Transactions on Information Systems (TOIS)*, vol. 36, no. 3, pp. 1–29, 2018.
- [47] Y. Liu, W. Wei, A. Sun, and C. Miao, “Exploiting geographical neighborhood characteristics for location recommendation,” in *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*, pp. 739–748, ACM, Shanghai, China, November 2014.
- [48] S. Zhao, I. King, and M. R. Lyu, “Geo-pairwise ranking matrix factorization model for point-of-interest recommendation,” in *Proceedings of International Conference on Neural Information Processing*, pp. 368–377, Springer, Guangzhou, China, November 2017.
- [49] T. Mikolov, “Using neural networks for modeling and representing natural languages,” in *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Tutorial Abstracts*, pp. 3–4, Dublin, Ireland, August 2014.
- [50] D. Clarke, “A context-theoretic framework for compositionality in distributional semantics,” *Computational Linguistics*, vol. 38, no. 1, pp. 41–71, 2012.
- [51] C. Yang, Z. Liu, D. Zhao, M. Sun, and E. Y. Chang, “Network representation learning with rich text information,” in *Proceedings of the 24th International Joint Conference on Artificial Intelligence (IJCAI 2015)*, pp. 2111–2117, Buenos Aires, Argentina, July 2015.
- [52] N. Zhou, W. X. Zhao, X. Zhang, J.-R. Wen, and S. Wang, “A general multi-context embedding model for mining human trajectory data,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 28, no. 8, pp. 1945–1958, 2016.
- [53] W.-T. Chu and Y.-L. Tsai, “A hybrid recommendation system considering visual information for predicting favorite restaurants,” *World Wide Web*, vol. 20, no. 6, pp. 1313–1331, 2017.
- [54] D. Liang, M. Zhan, and D. P. Ellis, “Content-aware collaborative music recommendation using pre-trained neural networks,” in *Proceedings of the 16th International Society for Music Information Retrieval Conference*, pp. 295–301, Malaga, Spain, October 2015.
- [55] R. He and J. McAuley, “VBPR: visual bayesian personalized ranking from implicit feedback,” in *Proceedings of the 30th AAAI Conference on Artificial Intelligence (AAAI-16)*, pp. 144–150, Phoenix, AZ, USA, February 2016.
- [56] S. Wang, Y. Wang, J. Tang, K. Shu, S. Ranganath, and H. Liu, “What your images reveal: exploiting visual contents for point-of-interest recommendation,” in *Proceedings of the 26th International Conference on World Wide Web, International World Wide Web Conferences Steering Committee*, pp. 391–400, Perth, Australia, April 2017.
- [57] S. Deng, L. Huang, G. Xu, X. Wu, and Z. Wu, “On deep learning for trust-aware recommendations in social networks,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 5, pp. 1164–1177, 2017.
- [58] J. Chen, H. Zhang, X. He, L. Nie, W. Liu, and T.-S. Chua, “Attentive collaborative filtering: multimedia recommendation with item-and component-level attention,” in *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 335–344, ACM, Shinjuku, Tokyo, Japan, August 2017.
- [59] H. Guo, R. Tang, Y. Ye, Z. Li, and X. He, “DeepFM: a factorization-machine based neural network for ctr prediction,” in *Proceedings of the 26th International Joint Conference on Artificial Intelligence (IJCAI-17)*, pp. 2782–2788, Melbourne, Australia, August 2017.
- [60] D. Kim, C. Park, J. Oh, S. Lee, and H. Yu, “Convolutional matrix factorization for document context-aware recommendation,” in *Proceedings of the 10th ACM Conference on Recommender Systems*, pp. 233–240, ACM, Boston, MA, USA, September 2016.
- [61] L. Guo, Y.-F. Wen, and X.-H. Wang, “Exploiting pre-trained network embeddings for recommendations in social networks,” *Journal of Computer Science and Technology*, vol. 33, no. 4, pp. 682–696, 2018.
- [62] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, “Distributed representations of words and phrases and their compositionality,” in *Proceedings of the 2013 Advances in Neural Information Processing Systems*, pp. 3111–3119, Lake Tahoe, NV, USA, December 2013.
- [63] Z. Gantner, S. Rendle, C. Freudenthaler, and L. Schmidt-Thieme, “Mymedialite: a free recommender system library,” in *Proceedings of the 5th ACM Conference on Recommender Systems*, pp. 305–308, ACM, Chicago, IL, USA, October 2011.

