A Deep Neural Network for Crossing-City POI Recommendations

Dichao Li and Zhiguo Gong, *Senior Member, IEEE*

Abstract—With the popularity of location-aware devices (e.g., smart phones), large amounts of location-based social media data such as check-ins are generated. This stimulates plenty of studies for POI recommendations by applying machine learning techniques. However, most of the existing studies focus on POI recommendations in the same city or region, and fail to recommend POIs for users when they travel to a new city. In this paper, we propose a novel deep neural network, named as ST-TransRec, for crossing-city POI recommendations. It integrates the deep neural network, transfer learning technique, and density-based resampling method into a unified framework. In this model, the deep neural network is used to capture users' preferences for POIs and learn the embeddings of POIs. Besides, the transfer learning technique is employed to bridge the gap between cities that results from the city-dependent features. As the distributions over POIs are imbalanced, we design a density-based spatial resampling model which enables POIs to be well matched across cities. We conduct extensive experiments on two real-world datasets. The experimental results show the advantages of ST-TransRec over the state-of-the-art methods for crossing-city POI recommendations.

Index Terms—Crossing-city, POI recommendation, deep learning, transfer learning, density-based resampling

1 Introduction

The popularity of location-aware devices such as smart phones makes users freely share their activities through various location-based social networks (LBSNs), such as Foursquare[1] and Yelp.[2] A large amount of user-contributed data enable to develop effective point-of-interest (POI) recommender systems. It not only guides users to explore more interesting attractions, but also helps the location service providers deliver targeted advertising. Now most of existing studies focus on recommending POIs in the same city or region, named as traditional POI recommender systems. However, they fail to deal with the increasingly popular case: users travel to new cities to explore more attractions. This raises the problem that how we shall recommend POIs in a target city to a new visitor based on her/his check-in records in source cities. We refer to this problem as crossing-city POI recommendations. Compared with traditional POI recommender systems, crossing-city POI recommender systems are more important since users are more eager to receive recommendations of POIs in a target city where he/she has not visited before. Despite its great importance, this problem is confronted with the following challenges:

Sparse Check-ins of Crossing-City Users in a Target City. The number of check-ins generated in a target city by crossing-city users who have visited both source and target cities is limited [1]. Specifically, the check-in records generated in Los Angeles by users from New York only accounted for 0.47 percent of their total check-ins in New York and 0.75 percent of the check-ins generated by local users in Las Angeles. Therefore, it is ineffective to depend on similar crossing-city users who visited the same POIs for the crossing-city POI recommendations.

Behaviors Drifting of Crossing-City Users. A promising way for the crossing-city POI recommendations is to integrate contents of POIs. However, the city-dependent features for each city are quite different, which significantly makes users' behaviors drift across cities. For example, users traveling to Las Vegas tend to visit casinos while they prefer to visit colleges once they travel to Boston (though it also has casinos). Therefore, it is unsafe to directly transfer users' preferences inferred from their check-in records in source cities to a target city.

Imbalanced Distributions over POIs. Users, especially travelers, prefer to visit the easily accessible regions such as transportation convenient regions instead of the marginal ones. As a result, POIs in more accessible regions will be more attractive, which leads the distributions over POIs to be strongly related to the locations of POIs. Furthermore, users' preferences for POIs in sparse regions will be underweighted while those in dense regions will be overweighted.

To address the above challenges, some Collaborative Filtering (CF) [2] based methods such as [3], [4], and [5] were proposed. However, they failed to work properly for the crossing-city POI recommendations. For example, Ference *et al.* [4] proposed to compute users' similarities based on both their profiles and those of their social friends. However, as pointed out in [3] about 90 percent POIs that users had visited before were not visited by any friends when they traveled more than 100 km. An alternative is to incorporate the textual descriptions of POIs into

1. https://foursquare.com/
2. https://www.yelp.com/

• *The authors are with the State Key Laboratory of Internet of Things for Smart City, Department of Computer and Information Science, University of Macau, Taipa, Macau 999078, China. E-mail: {yb57417, fstzgg}@um.edu.mo.*

Manuscript received 18 Jan. 2020; revised 9 Oct. 2020; accepted 21 Oct. 2020. Date of publication 26 Oct. 2020; date of current version 7 July 2022.
(Corresponding author: Zhiguo Gong.)
Recommended for acceptance by M. Wang.
Digital Object Identifier no. 10.1109/TKDE.2020.3033841

recommender systems. Bao *et al.* [1] and Zhao *et al.* [6] proposed to match users from a source city with those in a target city via the textual descriptions of their visited POIs. Further, some topic model based models such as [7], [8], [9], and [10] were proposed to solve the sparsity problem of crossing-city users in the target city. However, these methods can only conduct users' preference transfer via shallow features of POIs, which are not adaptive due to various city-dependent features.

Recently, deep neural networks have achieved great success on recommender systems. For instance, SH-CDL [11] applied deep belief network to learning unified representations of POIs. Although this method addressed the data sparsity problem of crossing-city users to some extend, it failed to eliminate the impact of city-dependent features when capturing POI representations and users' preferences. Yang *et al.* [12] extended the deep neural collaborative filtering framework [13] by jointly predicting the context of POIs and modeling user-POI interactions. However, this model still suffered from the data sparsity and behaviors drift of crossing-city users as it just exploited the geographical relations among POIs within a limited distance .

Besides the mentioned limitations of existing studies, none of them noticed and tackled the imbalanced distribution problem caused by the geographical locations of POIs in the city when trying to match POIs (or users) through the textual descriptions. It is intuitive that the spatial features of POIs enable to well define the distributions over POIs and further alleviate the bias of recommender system towards POIs in the dense-distributed regions.

Motivated by the above points, we propose a spatial and textual transfer learning framework for crossing-city POI recommendation, named as ST-TransRec. In this model, we combine the deep neural network and transfer learning technique to address the sparse check-ins and behaviors drift of crossing-city users, and design a resampling algorithm to overcome the imbalanced distribution problem.

Specifically, we first apply the Word2vec technique [14] to learning the embeddings of POIs based on their textual descriptions. However, the learned POI embeddings are not safely transferable due to city-dependent features. There is an incentive to eliminate the impact of city-dependent features. Maximum Mean Discrepancy (MMD) [15] is a widely used transfer learning technique which is applicable to address this problem. MMD enables to enhance the transferability so that POIs across different cities can be well matched. On the top of that, it is feasible to accurately capture users' embeddings through the available user-POI interactions. When users arrive at a target city, the local POIs can be accurately recommended based on their check-ins in source cities.

However, if a source POI and a target POI have significantly different number of check-ins, it will be difficult for a transfer learning technique to match them well even though they have quite similar textual descriptions. We ascribe the cause of imbalanced distributions over POIs to the geographical locations of POIs in a city, i.e., downtown regions often naturally have more travelers compared with the marginal ones. To solve this problem, we segment the whole city into several regions, such that travelers can easily access to any POIs in the same region while relatively few users can travel across regions in one trip. Then, the check-ins in each region are generated with respect to POIs' attraction distribution (not affected by the geographical locations) while the check-ins in different regions may be skewed with respect to the geographical locations of the regions. To prevent a sparse region from being overwhelmed by dense regions, we resort a resampling algorithm to conducting the transfer learning technique. This enables POIs in different regions to be in balanced distributions when extracting city-independent features of POIs.

Our main contributions are summarized as follows.

- We propose a novel deep neural network, referred as ST-TransRec, for crossing-city POI recommendations. This model integrates user-POI interactions and textual descriptions of POIs to overcome data sparsity, and exploits transfer learning technique to enhance the transferability of POI embeddings.
- We explore the resampling method for addressing the imbalanced distributions over POIs. This method enables POIs to be identically and independently distributed across cities, and further improves the effectiveness of transfer learning.
- We conduct extensive experiments on two real-world datasets. The experimental results indicate that ST-TransRec outperforms the state-of-the-art methods for the crossing-city POI recommendations.

The rest of this paper is organized as follows. Section 2 gives a preliminary; Section 3 presents our proposed model in detail; the experimental results are reported in Section 4; Section 5 reviews the related studies and finally we make conclusions in Section 6.

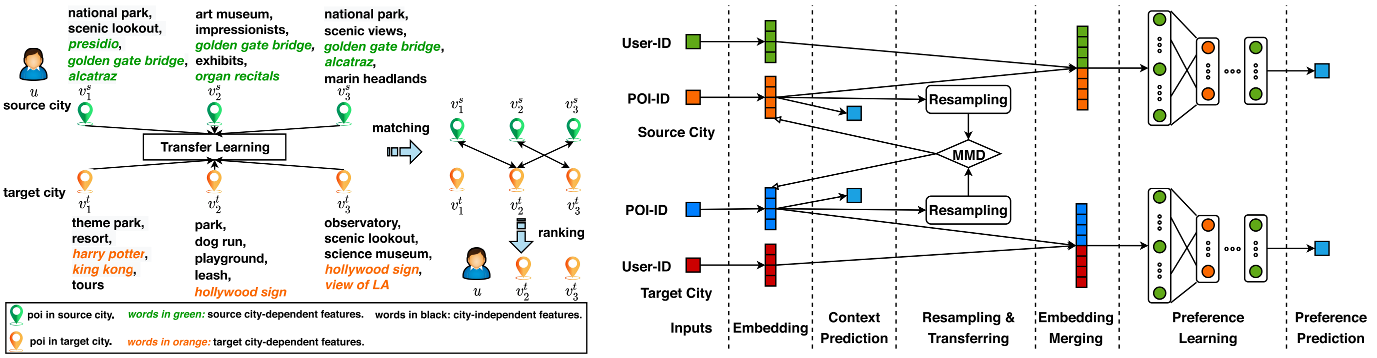
2 Preliminary

Here, we give an overview of Maximum Mean Discrepancy [15] and introduce some definitions used in this paper.

2.1 Maximum Mean Discrepancy (MMD)

Let X denote a random variable on domain Ω with distribution $P(X)$, and x be the instantiations of X. $\mathcal{H}$ denotes the reproducing kernel Hilbert space (RKHS) endowed with a characteristic kernel k satisfying $k(\cdot,\cdot) = \langle \phi(\cdot), \phi(\cdot) \rangle$, where ϕ is a feature map. The mean embedding of distribution P can be represented by an element in RKHS associated with kernel k such that $\mu_X(P) \triangleq \int_\Omega \phi(x)dP(x)$. While the true distribution $P(X)$ is not accessible, its mean embedding can be estimated by a finite set of samples. Given a set of samples $D_X = \{x_1, \ldots, x_n\}$ of size n that are identically and independently drawn from $P(X)$, the empirical estimation of mean embedding for P is defined as $\widehat{\mu}_X = \frac{1}{n}\sum_{i=1}^{n} \phi(x_i)$.

Let $\mathcal{D}_{X^s} = \{x_1^s, \ldots, x_{n_s}^s\}$ and $\mathcal{D}_{X^t} = \{x_1^t, \ldots, x_{n_t}^t\}$ denote the sets of samples from distributions $P(X^s)$ and $Q(X^t)$, respectively. The insight of MMD [15] is that if the distributions are identical, all of their statistics should be the same. The theoretical result indicates that $P = Q$ if and only if $D_{\mathcal{H}}(P,Q) = 0$. In an universal RKHS $\mathcal{H}$, MMD measures the discrepancy $D_{\mathcal{H}}(P,Q)$ between two distributions P and Q, which can be expressed as distance between their mean embeddings of distributions. Formally, it is defined as



(a) Transfer Learning via City-independent Features

(b) The Neural Architecture of ST-TransRec

Fig. 1. Illustration of ST-TransRec.

$$D_{\mathcal{H}}(P,Q) \triangleq ||\mu_X(P) - \mu_X(Q)||^2_{\mathcal{H}}. \tag{1}$$

In practice, an estimate of MMD is defined as

$$\begin{aligned}\hat{D}_{\mathcal{H}}(P,Q) = & \frac{1}{n_s^2}\sum_{i=1}^{n_s}\sum_{j=1}^{n_s} k(x_i^s, x_j^s) \\ & + \frac{1}{n_t^2}\sum_{i=1}^{n_t}\sum_{j=1}^{n_t} k(x_i^t, x_j^t) \\ & - \frac{2}{n_s n_t}\sum_{i=1}^{n_s}\sum_{j=1}^{n_t} k(x_i^s, x_j^t),\end{aligned} \tag{2}$$

where $\hat{D}_{\mathcal{H}}(P,Q)$ is an unbiased estimator of $D_{\mathcal{H}}(P,Q)$.

2.2 Problem Definition

Definition 1 (Check-in Record). *A check-in record is a tuple (u, v, l_v, W_v, c), where u, v, l_v, W_v, and c denote a user, a POI, location of the POI, textual descriptions of the POI, and the city of the POI, respectively. In practice, l_v is usually in the form of latitude and longitude, and W_v usually represents categories and tips describing POIs.*

Definition 2 (Textual Context Graph). *A textual context graph $\mathcal{G}_{vw}$ is defined as $\mathcal{G}_{vw} = \{\mathcal{V}, \mathcal{W}, \mathcal{E}_{vw}\}$, where $\mathcal{V}$ is the set of POIs, $\mathcal{W}$ is the word vocabulary; and $\mathcal{E}_{vw}$ is the set of edges between POIs and words.*

In this work, the textual descriptions such as categories and tips of POIs are used to construct the textual context graph. In the graph, the nodes are POIs and words, and the edges are constructed by connecting a POI to each word in its textual descriptions.

Definition 3 (User Profile). *Each user's profile is formally defined as a set of check-in records, i.e., $D_u = \{(u, v_i, t_i, l_{v_i}, W_{v_i}, c_i)\}_{i=1}^{n_u}$, where n_u is the total number of check-ins generated by user u. The dataset D in our model consists of all user profiles, i.e., $D = \{D_u : u \in U\}$, where U is the set of all users in the collection.*

Problem 1 (Crossing-city POI Recommendations). *Given a target user u associated with her/his profile D_u in source cities and a target city that she/he has never visited, the task of crossing-city POI recommendations is to find the top-k POIs in target city that match u's interests.*

3 Model

In this paper, we propose a spatial and textual transfer learning framework for crossing-city POI recommendations, named as ST-TransRec. In the following, we will elaborately introduce the proposed model, parameter learning, and time complexity analysis.

3.1 ST-TransRec

Here, we will present the principles of ST-TransRec, the overall framework, and each module consisting of it.

3.1.1 *Principles of ST-TransRec*

Our key idea of transfer learning via city-independent features is illustrated in Fig. 1a, where v_i^s and v_i^t denote POIs in source and target cities, respectively. Observed from the figure, each POI is associated with its textual descriptions which are comprised of the city-dependent words and city-independent words. City-dependent words will lead to POIs across cities fail to match each other even though they are semantically similar to each other. Therefore, it is an incentive to eliminate the city-dependent features and enhance the city-independent features. For example, the preferences of user u can be inferred from her/his check-in records $\{v_1^s, v_2^s, v_3^s\}$ in the source city and she/he is likely to be interested in parks and museums. Although v_2^s and v_3^t share the same city-independent feature as museum, city-dependent features such as "golden gate bridge" in the source city and "hollywood sign" in the target city will result in gap between them. As a result, v_2^s fails to correctly match v_3^t. Fortunately, transfer learning will enable to enhance city-independent features and make v_2^s match v_3^t. Similarly, v_2^t will match v_1^s and v_3^s. Therefore, v_2^t and v_3^t will be recommended to user u. Besides, we assume user u has visited a POI v_1^s tagged by "national park" in a sparse region. v_1^s is likely to be ignored and hard to match POIs in the target city due to the small probability of being sampled. Therefore, we are targeting at resampling to improve its probability of being sampled.

3.1.2 *Overall Framework*

Fig. 1b gives an illustration of ST-TransRec. It is comprised of a deep neural network, a transfer learning layer, and a resampling module. Specifically, the inputs are user-ids and POI-ids from source and target cities, respectively. The randomly initialized embeddings of POIs are fed to a fully connected embedding layer to predict the textual context as we assume words as the media to bridge the gap between POIs from different cities. And the loss for context prediction is $\mathcal{L}_{\mathcal{G}_{vw}}$, by optimizing which the textual embedding of POIs can be derived. Then, a resampling module is injected to overcome the problem of imbalanced distributions of check-ins over POIs. After that, an MMD [16] based transfer learning layer is embedded to enhance the transferablity of POI embeddings. We assume that the embeddings of POI in source and target cities follow the distributions P and Q, respectively. Their discrepancy is $D(P,Q)$. Minimizing $D(P,Q)$ enables to capture the city-independent embeddings of POIs and further match POIs across cities. Further, embeddings of users and POIs are merged and then fed to a multi-layer perceptron for modeling the user-POI interactions. At last, the prediction layer yields the prediction score to minimize the prediction loss $\mathcal{L}_I$. To meet all these criteria, the overall loss for ST-TransRec is defined as follows.

$$\mathcal{L} = \mathcal{L}_I^s + \mathcal{L}_{\mathcal{G}_{vw}}^s + \mathcal{L}_I^t + \mathcal{L}_{\mathcal{G}_{vw}}^t + \lambda D(P,Q), \tag{3}$$

where $\mathcal{L}_I$ denotes the loss of user-POI interactions; $\mathcal{L}_{\mathcal{G}_{vw}}$ is the loss for context predictions; $D(P,Q)$ is the discrepancy between distributions over POIs from source and target cities, respectively; λ is a hyperparameter; and the superscripts s and t indicate the source and target city, respectively.

3.1.3 Learning POI Embeddings

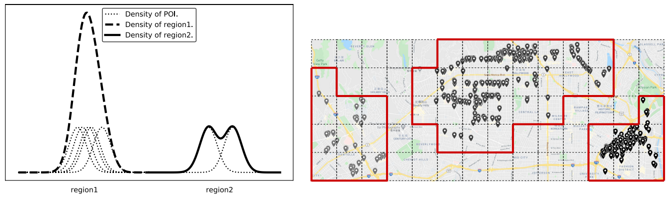
To obtain the POI embeddings, we first construct a textual context graph as each POI is associated with various of textual descriptions. Such kind of information can be easily built ino a textual context graph $\mathcal{G}_{vw} = \{\mathcal{V}, \mathcal{W}, \mathcal{E}_{vw}\}$, where $\mathcal{V}$ is the set of POIs, $\mathcal{W}$ is the word vocabulary, and $\mathcal{E}_{vw}$ is the set of edges connecting words to POIs. Based on this constructed graph, we apply Skipgram [14] to learning POI embeddings by context predictions. Given the POI embedding, the objective is to minimize the binary cross entropy loss by predicting its context, which can be formulated as

$$\begin{aligned}\mathcal{L}_{\mathcal{G}_{vw}} &= -\sum_{(v,w)\in\mathcal{E}_{vw}} \log P(w|v)\\ &\approx -\sum_{(v,w)\in\mathcal{E}_{vw}} \Big[\log\sigma(\boldsymbol{x}_{\boldsymbol{w}}^{\boldsymbol{T}}\boldsymbol{x}_{\boldsymbol{v}}) + \sum_{w'\notin W_v}\log\sigma(-\boldsymbol{x}_{\boldsymbol{w'}}^{\boldsymbol{T}}\boldsymbol{x}_{\boldsymbol{v}})\Big],\end{aligned} \tag{4}$$

where v is a POI, w is the positive context of POI, w' is the negative context of POI, W_v is the set of all positive contexts for POI v, $\boldsymbol{x}_{\boldsymbol{w}}$ is the embedding of word w, and $\boldsymbol{x}_{\boldsymbol{v}}$ is the embedding of POI v. Based on the losses $\log P(w|v)$ and $\log P(w|v')$ for two POIs v and v', two POI embeddings $\boldsymbol{x}_{\boldsymbol{v}}$ and $\boldsymbol{x}_{\boldsymbol{v'}}$ must be similar as they share the same context. In this way, minimizing the loss on all pairs of POIs and words will guarantee that POIs associated with more similar textual context will have more similar embeddings.

3.1.4 Resampling for Imbalanced Distributions

As the textual descriptions of POIs are city-dependent, the POI embeddings learned from the above model are not safely transferable across cities. There is an incentive to exploit transfer learning technique to drive the city-independent embeddings of POIs. As discussed in the preliminary, MMD [16] is a widely used technique for transfer learning. The assumption behind MMD is that data are identically and independently distributed. In this paper Gaussian kernel with fixed bandwidth is used to estimate MMD as defined in Equation (2). However, users tend to explore transportation-convenient regions in a city and subsequently generate more check-ins there, which results in imbalanced distributions over POIs in different regions. Therefore, it is infeasible to directly apply MMD to matching POIs across cities. That may lead the recommender system to have bias towards the POIs in dense regions and ignore POIs in sparse regions even though they are better matched users' tastes. Fig. 2a shows an example of such a situation. We assume region 1 is a dense region with 5 check-ins while region 2 is a sparse one with only 2 check-ins. Given a Gaussian kernel with fixed bandwidth σ, i.e., $k_\sigma(x,y) = \exp(-(x-y)^2/2\sigma^2)$, and a set of i.i.d samples $D_X = \{x_1, \ldots, x_n\}$ of size n from distribution P, the empirical mean embedding of P is estimated by $\mu_X = \sum_{i=1}^n k_\sigma(x,\cdot)/n$. If the kernel bandwidth is optimized based on region 1 (the dense region), the density estimation of region 2 (the sparse region) will be ignored. This is due to the strong co-relationship between the number of samples (check-ins) and the bandwidth of kernel, i.e., small number of check-ins requires a large bandwidth.



(a) Imbalanced Distributions over POIs

(b) Region Segmentations

Fig. 2. An example of imbalanced distributions in Los Angeles.

As theoretically discussed in [17] and [18], the bandwidth σ is a function of the sample size n, and it satisfies $\lim_{n\to\infty}\sigma(n) \to 0$ for the unbiased kernel estimations and $\lim_{n\to\infty} n\sigma(n) \to \infty$ for the uniform consistency. To this end, it is appropriate to reduce the bandwidth in regions with more check-in records and increase it in regions with fewer check-ins, or correspondingly, to use more samples (check-ins) in the sparse regions and fewer samples in dense regions.

Motivated by the above discussion, we first divide a city into regions, such that travelers on each region can easily access to any POIs on it while relatively few travelers can cross the border of two neighboring regions in one trip. In this way, we can assume the POI distribution in each region is generated purely by the POI's attraction instead of the degree of accessibility. Then we employ resampling technique to boost check-in records in the sparse regions.

Region Segmentation. Now, we discuss how to segment a city into regions. We first uniformly divide a city into $n_1 \times n_2$ equal-sized small grids $\{r_{ij}\}$ and each POI v will correspond to a grid based on its location l_v. Then, starting from the dense grids we extensively merge two neighboring grids if they share enough number of common users. Fig. 2b shows an example of the result in Los Angeles, where three regions with bound lines are formulated with different density. In particular, we extensively merge adjacent grids starting at a seed grid until no more adjacent grids can be merged. Formally, we define the distance between two adjacent grids $r_{i_l j_l}$ and $r_{i_{l+1} j_{l+1}}$ as

$$dis(r_{i_l j_l}, r_{i_{l+1} j_{l+1}}) = \frac{|U_{r_{i_l j_l}} \cap U_{r_{i_{l+1} j_{l+1}}}|}{\min\{|U_{r_{i_l j_l}}|, |U_{r_{i_{l+1} j_{l+1}}}|\}}, \tag{5}$$

where $U_{r_{ij}}$ is the set of users who have visited a POI located in a grid r_{ij}. And a grid r_{ij} is accessible from a seed grid $r_{i_0 j_0}$ if there exists a sequence of grids $r_{i_1 j_1}, r_{i_2 j_2}, \ldots, r_{i_L j_L}$ with $r_{i_1 j_1} = r_{i_0 j_0}$ and $r_{i_L j_L} = r_{ij}$, such that every two adjacent grids $r_{i_l j_l}, r_{i_{l+1} j_{l+1}}$ satisfy the requirement $dis(r_{i_l j_l}, r_{i_{l+1} j_{l+1}}) \geq \delta$, where δ is a given threshold. All the accessible grids from a seed grid $r_{i_0 j_0}$ are merged together to formulate a uniformly accessible region $R(r_{i_0 j_0})$. This process is recursively applied to the remaining grids until no grids are left. The details of our proposed region segmentation algorithm are summarized in Algorithm 1.

Algorithm 1. Clustering for Uniformly Accessible Regions

Input: $n_1 \times n_2$ grids of a city, a threshold δ
Output: a set of uniformly accessible regions R

```
U = {r_ij}  unmerged grids
R = {}  uniformly accessible regions
while U is not empty do
    Randomly sample a grid r_{i0 j0} from U as a seed point;
    Choose a grid sequence r_{i1 j1}, r_{i2 j2}, ..., r_{iL jL} with
    r_{i1 j1} = r_{i0 j0} and r_{iL jL} = r_ij
    if dis(r_{il jl}, r_{il+1 jl+1}) >= δ then
        Merge a grid r_{il+1 jl+1} from U into R(r_{i0 j0});
    R = R ∪ R(r_{i0 j0});
    U = U − R(r_{i0 j0});
return R
```

Balancing Sparse Regions by the Resampling Technique. After the segmentation, we can assume that POIs in each region are uniformly accessible. However, the distributions over POIs in different regions are still imbalanced. To address this problem, we apply the resampling technique to making any two regions in a city have similar densities. Formally, the density-based resampling is defined as follows. For each region r, its density can be formulated as $\rho_r = n_r/S_r$, where n_r is the number of check-in records and S_r is the number of grids in region r, respectively. Let r^* be the region with maximum density ρ_{r^*} and r be a region with density ρ_r, respectively. Inspired by [17], the bandwidth σ and the density ρ should meet the criterion, $\rho_{r^*}\sigma_{r^*} \approx \rho_r \sigma_r$. Formally, the raw number of check-in records and the number of resampled check-in records for region r are n_r and n'_r, respectively, and they should satisfy

$$\frac{n_r + n'_r}{S_r} = \frac{n_{r^*}}{S_{r^*}}. \tag{6}$$

Formally, given a set of POIs $\{v_{r,1}, v_{r,2}, \ldots\}$ in a region r, we define the distribution over POIs as

$$P(V = v|r) = \frac{n_{r,v}}{\sum_{v'} n_{r,v'}}, v = 0, 1, \ldots, \tag{7}$$

where $n_{r,v}$ is the number of POI v in region r. And given a city c, the distribution over regions is defined as

$$P(r|c) = \frac{\rho_{r^*}/\rho_r}{\sum_{r'} \rho_{r^*}/\rho_{r'}}, r = 0, 1, \ldots, R, \tag{8}$$

where R is the number of regions in city c. The resampling procedure is motivated by [19] and is defined as follows.

$$\begin{cases} \text{Draw a region } r \text{ based on } P(r|c). \\ \text{Draw a POI } v \text{ based on } P(V = v|r). \end{cases} \tag{9}$$

In this way, the distributions are balanced. In this paper, we propose to lessen the impact of resampled data by using a punishment hyperparameter α in the range of [0,1]. The number of resampling check-in records is set as $\alpha \sum_{r=1}^{R} n'_r$. Obviously, when α is set to 0, there is no resampling data; when α is set to 1, it will make POIs over all regions uniformly distributed; when α is smaller than 1, the number of resampled data will be suppressed. In this paper, we set $\alpha < 1$ as a larger α means more sampled POIs in sparse regions. As a result, recommender system tends to overweight POIs in sparse regions which are not as important as those in dense regions for crossing-city POI recommendations.

3.1.5 Transferring POI Embeddings

After resampling for POIs in sparse regions, two sets of identically and independently distributed samples $D_{X_v^s} = \{\boldsymbol{x}^s_{v_1}, \boldsymbol{x}^s_{v_2}, \ldots, \boldsymbol{x}^s_{v_{n^s}}\}$ and $D_{X_v^t} = \{\boldsymbol{x}^t_{v_1}, \boldsymbol{x}^t_{v_2}, \ldots, \boldsymbol{x}^t_{v_{n^t}}\}$ are drawn from source and target cities, respectively. Their distributions are denoted as $P(X_v^s)$ and $Q(X_v^t)$. Based on the theory of MMD [15], the empirical estimate of the discrepancy between $P(X_v^s)$ and $Q(X_v^t)$ is formulated as

$$\begin{aligned} \widehat{D}(P,Q) = & \frac{1}{n_s^2} \sum_{r=1}^{R_s} \sum_{i=1}^{n_{s,r}} \sum_{r'=1}^{R_s} \sum_{j=1}^{n_{s,r'}} k(\boldsymbol{x}_{\boldsymbol{v}_i^s}, \boldsymbol{x}_{\boldsymbol{v}_j^s}) \\ & + \frac{1}{n_t^2} \sum_{r=1}^{R_t} \sum_{i=1}^{n_{t,r}} \sum_{r'=1}^{R_t} \sum_{j=1}^{n_{t,r'}} k(\boldsymbol{x}_{\boldsymbol{v}_i^t}, \boldsymbol{x}_{\boldsymbol{v}_j^t}) \\ & - \frac{2}{n_s n_t} \sum_{r=1}^{R_s} \sum_{i=1}^{n_{s,r}} \sum_{r'=1}^{R_t} \sum_{j=1}^{n_{t,r'}} k(\boldsymbol{x}_{\boldsymbol{v}_i^s}, \boldsymbol{x}_{\boldsymbol{v}_j^t}), \end{aligned} \tag{10}$$

where $n_s = \sum_{r=1}^{R_s} n_{s,r} (n_t = \sum_{r=1}^{R_t} n_{t,r})$, $R_s(R_t)$, and $n_{s,r}(n_{t,r})$ are the number of samples, regions, and samples of region r in a source(target) city; and $k(\cdot,\cdot)$ is a Gaussian kernel. Intuitively, the smaller MMD, the similar distributions over POIs across cities. Therefore, minimizing $\widehat{D}(P,Q)$ will enhance the city-independent features and eliminate the city-dependent features.

3.1.6 Modeling User-POI Interactions

Given a user's profile D_u, the embeddings $\boldsymbol{x_u}$ and $\boldsymbol{x_v}$ of user u and POI v can be obtained in the embedding layer. They are concatenated as $[\boldsymbol{x_u}, \boldsymbol{x_v}]$ and then fed to a multi-layer perceptron. Formally, the hidden layer is defined as

$$\boldsymbol{e_L} = \sigma_L(\boldsymbol{W_L}\sigma_{L-1}(\cdots\sigma_1(\boldsymbol{W_1}[\boldsymbol{x_u}, \boldsymbol{x_v}] + \boldsymbol{b_1})\cdots) + b_{\boldsymbol{L}}), \tag{11}$$

where $\boldsymbol{W_l}$, $\boldsymbol{b_l}$, σ_l, and $\boldsymbol{e_l}$ $(l = 1, \ldots, L)$ denote the weight matrix, bias vector, activation function, and output vector of the lth hidden layer, respectively. The activation function is defined as $\mathrm{ReLU}(x) = \max(0, x)$. Finally, the prediction score $\hat{y}_{uv}$ is generated in the prediction layer by the following equation.

$$\hat{y}_{uv} = \sigma(\boldsymbol{W}^T \boldsymbol{e}_L), \tag{12}$$

where $\boldsymbol{W}^T$ is the weight vector of the prediction layer and σ is the sigmoid function defined as $\sigma(x) = 1/(1 + \exp(-x))$.

We adopt the binary cross entropy loss to optimize the parameters, which is defined as

$$\mathcal{L}_I = -\sum_{(u,v)} (y_{uv} \log \hat{y}_{uv} + (1 - y_{uv}) \log (1 - \hat{y}_{uv})), \tag{13}$$

where $\mathcal{L}_I$ is the loss for user-POI interactions, and y_{uv} is the label. Only positive labels, i.e., the observed interactions, are available, therefore, we uniformly sample the negative labels from unobserved interactions.

3.2 Parameter Learning and Time Complexity Analysis

To learn the parameters for ST-TransRec, we jointly train our model *w.r.t.* the multiple objectives in Equation (3). In each training epoch, we first randomly generate negative instances based on the interactions of users and POIs, and set the labels of these negatives as 0. In our experiments we set the number of negative samples as 4 similar to [13]. Simultaneously, we generate the textual context of POIs based on the textual context graph $\mathcal{G}_{vw}$. Then we sample a batch of labeled pairs of users and POIs, and labeled pairs of POIs and words from source and target cities, respectively. Finally, we take a gradient step to optimize the overall loss based on the sampled batch. We repeat the above procedures for T iterations until $\mathcal{L}$ converges. To avoid overfitting, we adopt dropout on the embedding layer and each hidden layer, and the dropout rate is ρ.

Based on parameter learning process, we notice that the main costs of our proposed model are predicting textual context of POIs, transferring POI embeddings, and modeling user-POI interactions. Let V and D denote the number of POIs and check-ins. For the module of textual context prediction, the time complexity is $\mathcal{O}(nKD)$, where n is the average degree of POIs in the textual context graph $\mathcal{G}_{vw}$ and K is the number of negative contexts of a POI. A direct implementation of MMD takes time $\mathcal{O}(D^2)$ as we need to compute the pairwise similarities between POIs. To reduce the time complexity, we adopt the technique used in [16] which enables to compute MMD with cost $\mathcal{O}(D)$. The time complexity to model user-POI interactions is $\mathcal{O}(D)$. Besides, there is a module of resampling with cost $\mathcal{O}(V)$, which can be omitted due to $V \ll D$. And the number of resampled data is αD linear to the number of check-ins. Therefore, the total time complexity for each iteration is $\mathcal{O}(nD)$, which is n times of the number of check-ins.

4 Experiments

We conduct extensive experiments to evaluate the effectiveness of ST-TranRec and report the experimental results in this section.

TABLE 1
Statistics of Datasets

		Foursquare	Yelp
Total Data	#Users	3,600	9,805
	#POIs	31,784	6,910
	#Words	3,619	1,648
	#Check-ins	191,515	433,305
Crossing-city Data	#Users	732	983
	#Check-ins	3,520	6,137

4.1 Experimental Settings

Datasets. We conduct experiments on two publicly available datasets, i.e., Foursquare[3] and Yelp.[4] The statistics of them are summarized in Table 1.

Foursquare. This is a check-in dataset and each check-in record is formulated as user-ID, POI-ID, time, contents, location, and city. The contents here are referred to categories and tips describing the POIs. We regard Los Angeles as the target city and the remaining cities as source cities.

Yelp. This dataset is from Yelp Challenge. We extract the users who have post at least ten reviews in the city of Phoenix and Las Vegas. Those reviews are formed as check-in record which is stored as user-ID, POI-ID, POI-location, POI-content.

Dataset Construction. To support the scenario of the crossing-city POI recommendations, we pick out a city as the target city and the remaining cities as source cities. Los Angeles in Foursquare and Las Vegas in Yelp are regarded as target cities, respectively. Given a collection of data $D = \{D_u : u \in U\}$, we pick out the crossing-city users who have visited both the target and source cities based on their profiles D_u. The crossing-city users are regarded as test users and their check-ins in the target city are regarded as ground truth for testing. The check-ins of non-crossing-city users and the remaining check-ins of crossing-city users in the source cities are used for training.

Evaluation Metrics. Following the common practice in evaluating the performance for POI recommendations, we randomly sample 100 POIs in the target city that are not visited by a crossing-city user and rank them with the ground truth. To evaluate the performance of recommendation methods, we first compute the ranking score for the ground truth and the sampled POIs, then form a ranked list based on scores of POIs, and finally get a top-k recommendation list. We adopt 4 widely-used metrics, i.e., recall (Recall@k), precision (Precision@k), normalized discounted cumulative gain (NDCG@k), and mean average precision (MAP@k) to evaluate all the methods, whose definitions are referred as [20]. The values of k are set as 2, 4, 6, 8, and 10. We calculate all metrics for each test user and present the average scores.

Baselines. We compare our proposed method with the following methods:

-*ItemPop*. This method ranked POIs based on their popularity, judged by the number of check-ins.

-*LCE*[21]. This was a matrix factorization framework. This model combined the contents of items and collective

3. https://sites.google.com/site/dbhongzhi/
4. https://www.yelp.com/dataset/challenge

information of users to learn the local collective embeddings for cold-start recommendations.

-*CRCF*[22]. It was a CF-based model for crossing-region POI recommendations. Users' content interests and location preferences over POIs were learned and combined to predict the probability of users' visit to a POI in a new region or a new city.

-*PR-UIDT*[5]. This was a CF-based framework which utilized the crossing-city users to make users' preferences transfer across cities. As there are no crossing-city users in our scenario, this model makes users' preferences learned from the source city directly match POIs in the target city.

-*ST-LDA*[23]. It proposed a probabilistic generative model to learn region-dependent personal interests and crowd's preferences for out-of-town POI recommendations.

-*CTLM*[10]. This was a probabilistic generative model combined with transfer learning technique for crossing-city POI recommendations. To transfer users' interests across cities this model separated common topics from city-specific topics.

-*SH-CDL*[11]. It unified the deep belief network and matrix factorization to learn the deep representation POIs from heterogeneous features and users' spatial-aware personal preferences for out-of-town POI recommendations.

-*PACE*[12]. It proposed a deep neural architecture which jointly learned embeddings of POIs and users by predicting the context of POIs and modeling user-POI interactions .

To study the contribution of each module considered in ST-TransRec, we design the following variants of ST-TransRec.

-*ST-TransRec-1*. This is a simplified version of ST-TransRec, which ignores the impact of city-dependent features. MMD loss in Equation (3) is omitted to derive city-independent embeddings of POIs.

-*ST-TransRec-2*. It simplifies ST-TransRec without incorporating textual context prediction of POIs. While it fails to bridge the gap across cities, it still captures features shared by POIs such as their popularities.

-*ST-TransRec-3*. It fails to apply the density-based resampling in addressing imbalanced distributions over POIs, where the hyperparameter α is set as 0.

Implementation Details. We implement ST-TransRec based on Tensorflow.[5] ST-TransRec is optimized with Adam optimizer by searching the learning rate in $\{1e^{-5}, 5e^{-5}, 1e^{-4}, 5e^{-4}, 1e^{-3}, 5e^{-3}\}$, fixing batch size to 128, and initializing parameters with a Gaussian distribution. The embedding size is fixed to 64 on Foursquare while it is set as 128 on Yelp, respectively. The deep structure (without embedding layer) of ST-TransRec is set as $128 \rightarrow 64 \rightarrow 32 \rightarrow 16 \rightarrow 1$ on Foursqaure and $256 \rightarrow 128 \rightarrow 64 \rightarrow 32 \rightarrow 1$ on Yelp, respectively. For the hyparameters n and δ, we apply grid search and set them as 50 and 0.10 on Foursquare, and 60 and 0.25 on Yelp, respectively. For the deep learning based baseline SH-CDL and PACE, the hyparameters and structure are set the same to those of ST-TransRec. Besides, we use data parallelism to split the training process across multiple GPUs. In practice, we use a server equipped with two GPUs of type GeForce RTX 2080 Ti. We compare the training speed by Multi-GPU-2 and Single-GPU. The average costs of each iteration are summarized in Table 2. We can observe that training speed by Multi-GPU-2 is nearly twice faster than that by Single-GPU.

TABLE 2
Training Time by Using Different Number of GPU

Foursquare		Yelp	
Single-GPU	Multi-GPU-2	Single-GPU	Multi-GPU-2
94.29s	50.74s	275.44s	153.73s

4.2 Experimental Results

In this subsection, we demonstrate the experimental results conducted on the two datasets.

4.2.1 Performance Comparison

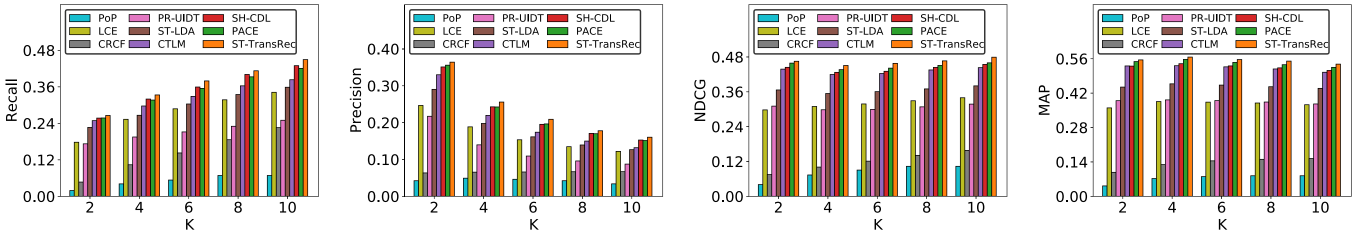
Here we report the top-k $(k = 2, 4, 6, 8, 10)$ performance of ST-TransRec and its competitors in Figs. 3 and 4.

Performance Comparison on Foursquare. Fig. 3 shows the experimental results on Foursquare, where the recall of ST-TransRec is about 0.450 when $k = 10$. Obviously, ST-TransRec achieves the best performance among its competitors. For example, the improvements of ST-TransRec *w.r.t.* Recall@10 are 39.4, 10.8, 22.0, 20.6, 9.87, 6.55, 2.30, and 2.50 percent compared with that of ItemPop, LCE, CRCF, PR-UIDT, ST-LDA, CTLM, SH-CDL, and PACE, respectively.

More specifically, we have the following observations. First, ST-TransRec achieves better performance than the state-of-the-art methods, i.e., CRCF, PR-UIDT, ST-LDA, CTLM, and SH-CDL, which are designed for the crossing-city POI recommendations. This demonstrates the effectiveness of ST-TransRec by combining the deep neural network, the density-based resampling, and transfer learning in a unified model. Further, the deep models, i.e., ST-TransRec, PACE, and SH-CDL, outperform the topic model based methods, i.e., CTLM and ST-LDA. And the CF-based methods, i.e., PR-UIDI, CRCF, and LCE, perform worst. The main reason behind this is that the deep structure contributes more to model the complex user-POI interactions than the shallow methods. For deep models, ST-TransRec performs better than PACE and SH-CDL, which indicates the effectiveness of transfer learning and density-based resampling for crossing-city POI recommendations. For topic model based methods, CTLM outperforms ST-LDA as CTLM utilizes transfer learning technique to match POIs across cities. For the CF-based methods, CRCF fails behind LCE and PR-UIDT as CRCF depends on the location of users in a new city. ItemPop performs worst, which indicates the effectiveness of modeling personality instead of just modeling popularity.

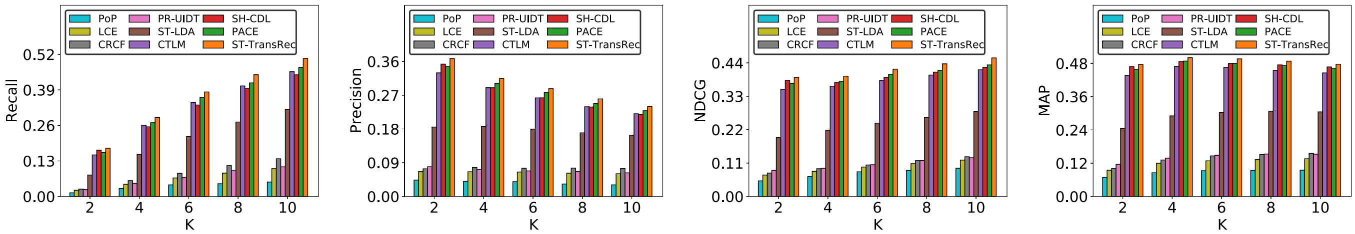
Performance Comparison on Yelp. Fig. 4 reports the performance on Yelp dataset. Similarly, our proposed model outperforms its competitors observed from those figures. Recall@10 of ST-TransRec is 0.505 and improvements are 45.2, 40.3, 36.7, 39.6, 18.6, 4.8, 5.9, and 3.3 percent compared with ItemPop, LCE, CRCF, PR-UIDT, ST-LDA, CTLM, SH-CDL, and PACE. It is worth noting that PACE performs better than SH-CDL although it is designed for crossing-city POI recommendation. The main reason is that PACE exploits a deep neural network to model user-POI interactions while SH-

5. https://www.tensorflow.org



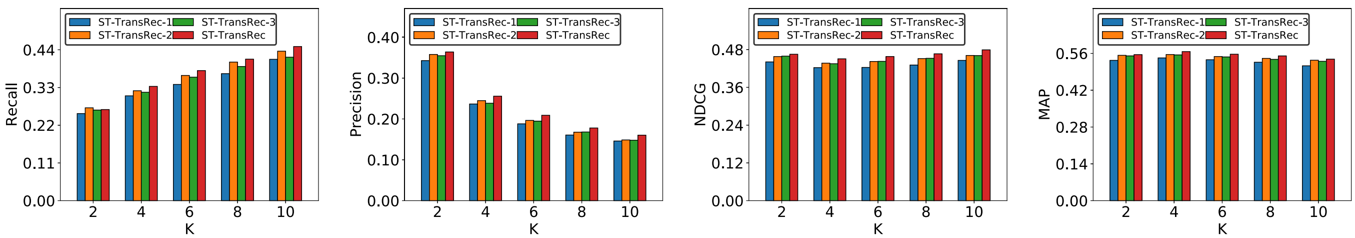
(a) Los Angeles-Recall (b) Los Angeles-Precision (c) Los Angeles-NDCG (d) Los Angeles-MAP

Fig. 3. Evaluation of top-k POI recommendation on foursquare dataset.



(a) Las Vegas-Recall (b) Las Vegas-Precision (c) Las Vegas-NDCG (d) Las Vegas-MAP

Fig. 4. Evaluation of top-k POI recommendation on yelp dataset.



(a) Los Angeles-Recall (b) Los Angeles-Precision (c) Los Angeles-NDCG (d) Los Angeles-MAP

Fig. 5. Evaluation of ST-TransRec variants on foursquare dataset.

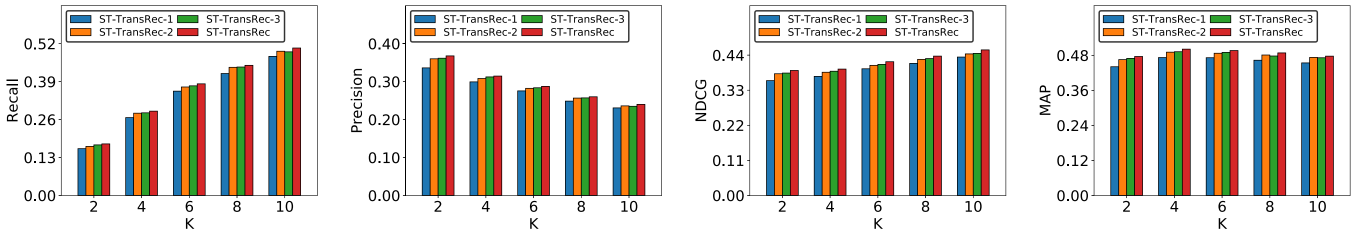
CDL only applies deep model in learning the representations of POIs. Besides, the performances of LCE, PR-UIDT, and ST-LDA are much worse on Yelp compared with on Foursquare. These three methods fail to eliminate the gap caused by city-dependent features and the discrepancy between the source and target cities on Yelp is larger compared with that on Foursquare. Finally, the performance of CRCF and Pop performs even worse than that on Foursquare dataset as users tend to visit popular POIs when they travel to Los Angeles.

4.2.2 Factor Analysis

To validate the effectiveness of factors considered in our proposed model, we compare ST-TransRec with its three variants and the hyparameters are set the same to ST-TransRec. Besides, we give a case study to further validate the contribution of textual features in improving the explainability of recommendation results.

Ablation Study. Figs. 5 and 6 show the results on Foursquare and Yelp, respectively. From the results, we observe that ST-TransRec outperforms its variants for most of the evaluation metrics on Foursquare and Yelp, respectively. For example, NDCG@10 of ST-TransRec is 0.4792, and the improvements are 3.35, 1.78, and 1.82 percent compared with ST-TransRec-1, ST-TransRec-2, and ST-TransRec-3 on Foursquare, respectively. These results validate that all the components considered in our proposed model are effective to improve the performance. The most important factor is transfer learning which follows by density-based resampling and the textual features. It is worthy noting that ST-TransRec-2 is also well-performed without incorporating textual descriptions of POIs. On one hand, the deep structure of ST-TransRec is able to capture features shared by POIs such as popularity. On the other hand, transfer learning technique enables to match POIs across cities based on the features learned from user-POI interactions.

Case Study. As ST-TransRec-2 also achieves good performance without incorporating textual descriptions of POIs, we give a case study to show the effectiveness of textual features. The detailed results are summarized in Table 3. The top-10 words collecting from the check-ins in source city are utilized to present the preferences of user #377. And we can observe that user #377 is interested in scenic views and arts. We evaluate the performance of ST-TransRec and ST-TransRec-2, respectively. The top-5 POIs associated with their top-5 textual descriptions are listed in the table. It is observed that the textual descriptions of the top-5 POIs (the bold ones are the ground truth in the test data) in the rank list of ST-TransRec are well matched user #377. This indicates ST-TransRec enables to accurately transfer the user's preferences from a source city to a target city and match POIs across cities. Besides, POIs such as Los Angeles International Airport and

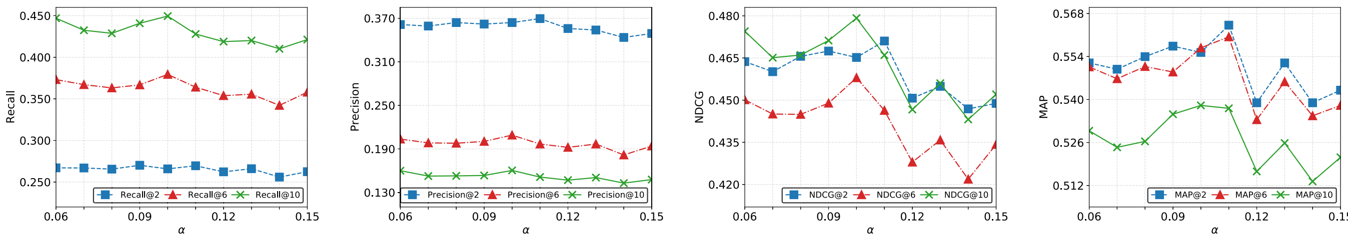


(a) Las Vegas-Recall (b) Las Vegas-Precision (c) Las Vegas-NDCG (d) Las Vegas-MAP

Fig. 6. Evaluation of ST-TransRec variants on yelp dataset.

TABLE 3
Case Study for User #377 of Foursquare

Training Data	Rank List of ST-TransRec		Rank List of ST-TransRec-2	
Top-10 Words	Top-5 POIs	Textual Descriptions	Top-5 POIs	Textual Descriptions
park scenic views	**ArcLight Cinemas**	multiplex, popcorn, cinerama dome, caramel corn, movies	Los Angeles International Airport	fligt, quiet, 24-hour, airports, wifi
tours theme	**Downtown Los Angeles ArtWalk**	art gallery, art museum, bar, historic downtown, hipster	**ArcLight Cinemas**	multiplex, popcorn, cinerama dome, caramel corn, movies
music blues	**EI Rey Theater**	concert hall, rock club, stage, dancing, theaters	Thai Patio	Thai restaurant, thai food, great thai, pad thai, spicy lime
fireworks candlelight	The Hollywood Roosevelt	hotel, historic site, old hollywood, swimming pool, bowling	**Downtown Los Angeles ArtWalk**	art gallery, art museum, bar, historic downtown, hipster
candy pizza	Bottega Louie	Italian restaurant, bakery, pizza place, portobello fires, cocktails	Bottega Louie	Italian restaurant, bakery, pizza place, portobello fires,cocktails



(a) Recall vs. Resample Rate (b) Precision vs. Resample Rate (c) NDCG vs. Resample Rate (d) MAP vs. Resample Rate

Fig. 7. Performance comparison *w.r.t.* resample rate on Los Angeles from foursquare.

Thai Patio in the rank list of ST-TransRec-2 fail to match the user's preferences.

4.2.3 Parameter Analysis

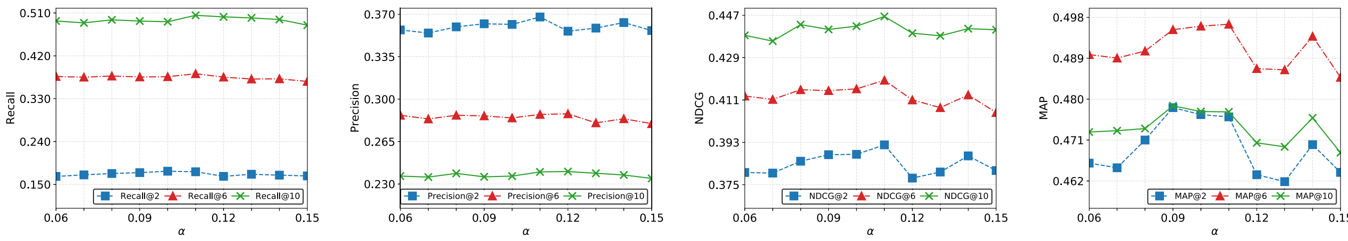
Here, we analyze the impact of hyperparameters used in ST-TransRec.

Resampling Rate. We further explore the effect of resampling ratio α. Figs. 7 and 8 show the performance of ST-TransRec in terms of recall, precision, NDCG, and MAP by setting the values of k as 2, 6, and 10. We range the values of α from 0.06 to 0.15. We avoid to setting the value of α as 1 because the large number of resampling data will lead our proposed model to be dominated by the POIs in marginal areas. As a result, the performance of ST-TransRec will decrease since the POIs in the dense regions are more important compared with those in spars regions. When the values of α on Foursquare and Yelp dataset are set as 0.10 and 0.11, ST-TransRec achieves the best performance.

Dropout Rate. The dropout technique is used to prevent ST-TransRec from overfitting. Fig. 9 demonstrates the performance on Foursquare and Yelp by setting the value of top-k as 10. We vary the dropout rate from 0 to 0.5 as a large value of dropout will prevent our model from reaching stable. We can observe that dropout is an effective way to improve the performance of our proposed model. When the dropout rate exceeds the optimal value, the evaluation metrics significantly decrease as a large dropout rate will result in under-fitting. The optimal dropout rate of ST-TransRec for Foursquare and Yelp are 0.1 and 0.2, respectively.

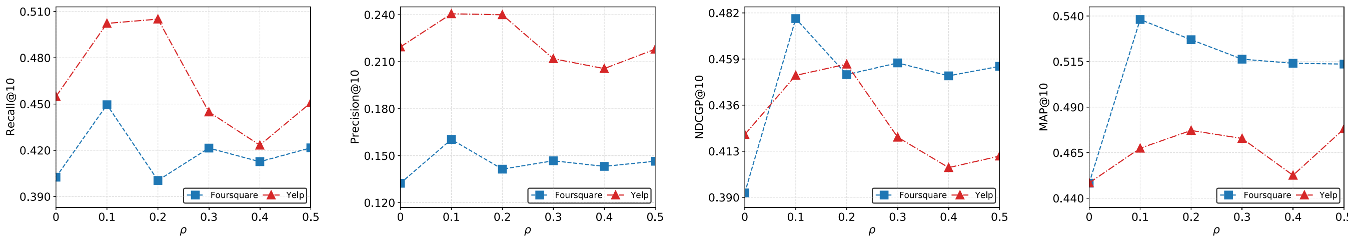
Embedding Size. We further investigate ST-TransRec by varying the embedding size as {16, 32, 64, 128}. Table 4 summarizes the results on Foursquare and Yelp dataset, respectively. Clearly, the optimal embedding sizes are 64 and 128 for Foursquare and Yelp dataset, respectively. It can be observed from the experimental results, the performance of our model can be improved by increasing the value of embedding size. Besides, the experimental results on Foursquare indicate that when the embedding size exceeds the optimal size our model will suffer from overfitting.

Depth of Hidden Layers. We vary the depth of ST-TransRec to investigate the impact of multiple hidden layers. The layer number is ranged of {1, 2, 3, 4} and the experimental results



(a) Recall vs. Resample Rate (b) Precision vs. Resample Rate (c) NDCG vs. Resample Rate (d) MAP vs. Resample Rate

Fig. 8. Performance comparison *w.r.t.* resample rate on Las Vegas from yelp.



(a) Recall vs. Dropout Rate (b) Precision vs. Dropout Rate (c) NDCG vs. Dropout Rate (d) MAP vs. Dropout Rate

Fig. 9. Performance comparison *w.r.t.* dropout rate.

TABLE 4
Recommendation Performance of ST-TransRec with Different Embedding Size

Embedding Size	Foursquare								Yelp							
	Recall		Precision		NDCG		MAP		Recall		Precision		NDCG		MAP	
	2	4	2	4	2	4	2	4	2	4	2	4	2	4	2	4
16	.2640	.3076	.3586	.2336	.4627	.4358	.5444	.5525	.1662	.2674	.3443	.2988	.3670	.3740	.4496	.4800
32	.2612	.3180	.3525	.2404	.4536	.4330	.5464	.5496	.1714	.2714	.3474	.2978	.3782	.3803	.4715	.4932
64	**.2658**	**.3333**	**.3641**	**.2558**	**.4652**	**.4507**	**.5553**	**.5668**	.1710	.2708	.3494	.3041	.3794	.3909	.4654	.4882
128	.2567	.3143	.3429	.2367	.4420	.4221	.5376	.5487	**.1765**	**.2888**	**.3677**	**.3145**	**.3919**	**.3962**	**.4760**	**.5012**

TABLE 5
Recommendation Performance of ST-TransRec With Different Number of Layers

Layers Number	Foursquare								Yelp							
	Recall		Precision		NDCG		MAP		Recall		Precision		NDCG		MAP	
	2	4	2	4	2	4	2	4	2	4	2	4	2	4	2	4
1	.2536	.3111	.3477	.2377	.4493	.4290	.5362	.5453	.1737	**.2888**	.3596	.3123	.3816	.3900	.4659	.4916
2	.2554	.3054	.3320	.2254	.4410	.4217	.5362	.5391	.1677	.2712	.3606	.3044	.3817	.3806	.4613	.4847
3	.2627	.3091	.3429	.2261	.4443	.4207	.5376	.5423	.1698	.2788	.3580	.3112	.3823	.3889	.4654	.4930
4	**.2658**	**.3333**	**.3641**	**.2558**	**.4652**	**.4507**	**.5553**	**.5668**	**.1765**	**.2888**	**.3677**	**.3145**	**.3919**	**.3962**	**.4760**	**.5012**

are summarized in Table 5. It is clear that ST-TransRec achieves the best performance when the numbers of layers are set as 4 on both datasets. This indicates that stacking more hidden layers will contribute to modeling the complex users and POIs interactions and further improving the recommendation performance.

5 Related Work

Based on techniques used for POI recommendations, the studies can roughly be divided into the pure collaborative filtering [2] based approaches, the content-assisted ones, and the deep neural networks based methods.

Pure CF-Based Models. The insight of the pure CF approaches is that similar users tend to have similar behaviors. For example, GM-FCF [24] and USG [25] were proposed to fuse user preference to a POI with social influence and geographical influence. Further, temporal and spatial influence were introduced into CF-based frameworks, such as [26] and [27], to support temporal and spatial aware POI recommendations. Instead of capturing the similarities between users or POIs based on check-in records directly, matrix factorization [28] was widely used for POI recommendations. Further it was extended in [29], [30], [31], [32], [33], [34], and [35] by incorporating different types of features such as spatial features of POIs and social relations among users to address the matrix sparsity problem. Later, the CF-based frameworks were extended for crossing-city POI recommendations. In [4] and [36] the check-in records generated by social friends were combined with those generated by similar users. Recently, Ding *et al.* [5] proposed to fuse matrix factorization and transfer learning for crossing-city POI recommendations, which enabled to jointly capture the drift and transfer of users' interests.

Content-Assisted Models. A promising way of improving the recommendation accuracy is to integrate contents of POIs. For example, some studies [21], [24], [37], [38], and [39] introduced contents into CF-based approaches. The models proposed in [1] and [6] mapped the users in their home city to a group of users in a target city by the content similarities of their visited POIs. Zhang *et al.* [22] exploited matrix factorization to model users' content interests and location preferences. Gao *et al.* [40] proposed to fuse user-POI matrix, user-word matrix, and POI-word matrix into a unified matrix factorization framework. Topic model is a widely used technique to model contents of POIs as well. Some spatial based topic models [41], [42], [43], [44], and [45] were proposed to capture users' mobility patterns and preferences for POIs by leveraging contents and geographical locations of POI. Further, some studies such as [7], [9], [23], [46], and [47] extended topic models for the out-of-town POI recommendations. For example, Yin *et al.* [7] proposed to provide out-of-town POI recommendations by topic model which captured both personal interests and local preferences based on the contents of POIs. Li *et al.* [10] proposed to fuse topic model and transfer learning technique into a unified framework, where the city-specific topics and the common ones are sperated based on the contents of POIs to support crossing-city POI recommendations.

Deep Neural Networks Based Models. The deep learning techniques have yielded immense success on POI recommendations. They were further extened by incorporating various contexts such as geographical and textual features of POIs, and social relations of users. For example, embedding learning techniques were applied by [48] and [49] to integrating multiple context into unified embedding models for the distributed representations of users and POIs. The collaborative deep learning model (CDL) [50] was further proposed to model contents of items. Inspired by CDL [50], Yin *et al.* [11] extended a deep belief network to fuse heterogeneous features of POIs into a unified latent vector and learned spatial-aware preferences of users for POI recommendations. Ma *et al.* [51] proposed an encoder-decoder framework to make users reach the similar and nearby neighbors of checked-in POIs. Further, collaborative filtering [2] was extended to deep neural collaborative filtering (NCF) [13]. Yang *et al.* [12] stood on the advantage of NCF [13] and proposed a deep neural network by jointly incorporating the geographical features of POIs and social relations among users to support POI recommendations. To model the sequential denpendency of check-in records, RNN-based model were proposed in [52] and [53]. For example, the model proposed in [53] not only modeled the denpendency of POIs but also captured the time-varing representation of users. Recently, Zhou *et al.* [54] proposed to combine deep learning technique with reinforcment learning, which enabled an adversarial mechanism to learn the discriminative preferences of users.

Existing studies such as [1], [4], [6], [7], [9], [10], [11], [22], [23], [47], and [5] are applicable to crossing-city POI recommendations while there are some key points distinguishing ST-TransRec from them. First, ST-TransRec exploits the textual descriptions instead of the crossing-city users used in pure CF-based methods [4] and [5]. Therefore, compare with the pure CF-based methods our proposed model is more effective to deal with the sparse check-ins of crossing-city users in a target city. Second, we employ transfer learning technique to eliminate the impact of city-dependent features and enhance the city-independent ones. However, most of models incorporating contents of POIs such as [22], [23], and [11] directly applied users' preferences learned from source cities to a target city and failed to consider the impact of city-dependent features. Third, we design a resampling technique to address the imbalanced distributions over POIs that all of the above studies failed to consider.

6 Conclusion

In this paper, we propose a novel deep neural network, named as ST-TransRec, to support crossing-city POI recommendations. ST-TransRec integrates the deep neural network, transfer learning technique, and density-based resampling into a unified model. The deep neural network is adopted to model the complex interactions of users and POIs, and learn the textual embeddings of POIs by predicting their contexts. The transfer learning technique is applied to reducing the discrepancy between distributions over POIs caused by city-dependent features. To address the imbalanced distributions over POIs, a density-based resampling is designed to make POIs identically and independently distributed. Extensive experiments are conducted on Foursquare and Yelp, respectively. Compared with the state-of-the-art algorithms, ST-TransRec achieves better performance for the crossing-city POI recommendations. Besides, we validate that factors considered in ST-TransRec are effective to improve its performance.

Acknowledgments

This work was supported by the National Key R&D Program of China (2019YFB1600704), FDCT (SKL-IOTSC-2018-2020, FDCT/0045/2019/A1, FDCT/0007/2018/A1, FDCT/007/2016/AFJ), GSTIC (EF005/FST-GZG/2019/GSTIC), and University of Macau (MYRG2017-00212-FST, MYRG2018-00129-FST).

References

[1] J. Bao, Y. Zheng, and M. F. Mokbel, "Location-based and preference-aware recommendation using sparse geo-social networking data," in *Proc. 20th Int. Conf. Advances Geogr. Inf. Syst.*, 2012, pp. 199–208.

[2] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, "Item-based collaborative filtering recommendation algorithms," in *Proc. 10th Int. Conf. World Wide Web*, 2001, pp. 285–295.

[3] E. Cho, S. A. Myers, and J. Leskovec, "Friendship and mobility: User movement in location-based social networks," in *Proc. 17th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2011, pp. 1082–1090.

[4] G. Ference, M. Ye, and W.-C. Lee, "Location recommendation for out-of-town users in location-based social networks," in *Proc. 22nd ACM Int. Conf. Inf. Knowl. Manage.*, 2013, pp. 721–726.

[5] J. Ding, G. Yu, Y. Li, D. Jin, and H. Gao, "Learning from hometown and current city: Cross-city POI recommendation via interest drift and transfer learning," *Proc. ACM Interactive Mobile Wearable Ubiquitous Technol.*, vol. 3, no. 4, pp. 1–28, 2019.

[6] Y.-L. Zhao, L. Nie, X. Wang, and T.-S. Chua, "Personalized recommendations of locally interesting venues to tourists via cross-region community matching," *ACM Trans. Intell. Syst. Technol.*, vol. 5, no. 3, 2014, Art. no. 50.

[7] H. Yin, Y. Sun, B. Cui, Z. Hu, and L. Chen, "LCARS: A location-content-aware recommender system," in *Proc. 19th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2013, pp. 221–229.

[8] S. Jiang, X. Qian, J. Shen, Y. Fu, and T. Mei, "Author topic model-based collaborative filtering for personalized POI recommendations," *IEEE Trans. Multimedia*, vol. 17, no. 6, pp. 907–918, Jun. 2015.

[9] H. Yin, X. Zhou, Y. Shao, H. Wang, and S. Sadiq, "Joint modeling of user check-in behaviors for point-of-interest recommendation," in *Proc. 24th ACM Int. Conf. Inf. Knowl. Manage.*, 2015, pp. 1631–1640.

[10] D. Li, Z. Gong, and D. Zhang, "A common topic transfer learning model for crossing city POI recommendations," *IEEE Trans. Cybern.*, vol. 49, no. 12, pp. 4282–4295, Dec. 2019.

[11] H. Yin, W. Wang, H. Wang, L. Chen, and X. Zhou, "Spatial-aware hierarchical collaborative deep learning for POI recommendation," *IEEE Trans. Knowl. Data Eng.*, vol. 29, no. 11, pp. 2537–2551, Nov. 2017.

[12] C. Yang, L. Bai, C. Zhang, Q. Yuan, and J. Han, "Bridging collaborative filtering and semi-supervised learning: A neural approach for POI recommendation," in *Proc. 23rd ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2017, pp. 1245–1254.

[13] X. He, L. Liao, H. Zhang, L. Nie, X. Hu, and T.-S. Chua, "Neural collaborative filtering," in *Proc. 26th Int. Conf. World Wide Web*, 2017, pp. 173–182.

[14] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Proc. 26th Int. Conf. Neural Inf. Process. Syst.*, 2013, pp. 3111–3119.

[15] A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Schölkopf, and A. Smola, "A kernel two-sample test," *J. Mach. Learn. Res.*, vol. 13, no. Mar, pp. 723–773, 2012.

[16] M. Long, H. Zhu, J. Wang, and M. I. Jordan, "Deep transfer learning with joint adaptation networks," in *Proc. 34th Int. Conf. Mach. Learn.*, 2017, pp. 2208–2217.

[17] K. Fukunaga and L. Hostetler, "The estimation of the gradient of a density function, with applications in pattern recognition," *IEEE Trans. Inf. Theory*, vol. IT-21, no. 1, pp. 32–40, Jan. 1975.

[18] J. Guo and Z. Gong, "A density-based nonparametric model for online event discovery from the social media data," in *Proc. 26th Int. Joint Conf. Artif. Intell.*, 2017, pp. 1732–1738.

[19] J. W. Butler, "Machine sampling from given probability distributions," in *Proc. Symp. Monte Carlo Methods*, 1956, pp. 249–264.

[20] Y. Liu, T.-A. N. Pham, G. Cong, and Q. Yuan, "An experimental evaluation of point-of-interest recommendation in location-based social networks," *Proc. VLDB Endowment*, vol. 10, no. 10, pp. 1010–1021, 2017.

[21] M. Saveski and A. Mantrach, "Item cold-start recommendations: Learning local collective embeddings," in *Proc. 8th ACM Conf. Recommender Syst.*, 2014, pp. 89–96.

[22] C. Zhang and K. Wang, "POI recommendation through cross-region collaborative filtering," *Knowl. Inf. Syst.*, vol. 46, no. 2, pp. 369–387, 2016.

[23] H. Yin, X. Zhou, B. Cui, H. Wang, K. Zheng, and Q. V. H. Nguyen, "Adapting to user interest drift for POI recommendation," *IEEE Trans. Knowl. Data Eng.*, vol. 28, no. 10, pp. 2566–2581, Oct. 2016.

[24] M. Ye, P. Yin, and W.-C. Lee, "Location recommendation for location-based social networks," in *Proc. 18th SIGSPATIAL Int. Conf. Advances Geographic Inf. Syst.*, 2010, pp. 458–461.

[25] M. Ye, P. Yin, W.-C. Lee, and D.-L. Lee, "Exploiting geographical influence for collaborative point-of-interest recommendation," in *Proc. 34th Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, 2011, pp. 325–334.

[26] Q. Yuan, G. Cong, Z. Ma, A. Sun, and N. M. Thalmann, "Time-aware point-of-interest recommendation," in *Proc. 36th Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, 2013, pp. 363–372.

[27] Q. Yuan, G. Cong, and A. Sun, "Graph-based point-of-interest recommendation with geographical and temporal influences," in *Proc. 23rd ACM Int. Conf. Inf. Knowl. Manage.*, 2014, pp. 659–668.

[28] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *Computer*, vol. 42, no. 8, pp. 30–37, 2009.

[29] B. Liu, Y. Fu, Z. Yao, and H. Xiong, "Learning geographical preferences for point-of-interest recommendation," in *Proc. 19th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2013, pp. 1043–1051.

[30] B. Liu and H. Xiong, "Point-of-interest recommendation in location based social networks with topic and location awareness," in *Proc. SIAM Int. Conf. Data Mining*, 2013, pp. 396–404.

[31] B. Liu, H. Xiong, S. Papadimitriou, Y. Fu, and Z. Yao, "A general geographical probabilistic factor model for point of interest recommendation," *IEEE Trans. Knowl. Data Eng.*, vol. 27, no. 5, pp. 1167–1179, May 2015.

[32] Y. Liu, W. Wei, A. Sun, and C. Miao, "Exploiting geographical neighborhood characteristics for location recommendation," in *Proc. 23rd ACM Int. Conf. Inf. Knowl. Manage.*, 2014, pp. 739–748.

[33] 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 *Proc. 20th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2014, pp. 831–840.

[34] 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 *Proc. 38th Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, 2015, pp. 433–442.

[35] H. Li, Y. Ge, and H. Zhu, "Point-of-interest recommendations: Learning potential check-ins from friends," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2016, pp. 975–984.

[36] H. Li, R. Hong, S. Zhu, and Y. Ge, "Point-of-interest recommender systems: A separate-space perspective," in *Proc. IEEE Int. Conf. Data Mining*, 2015, pp. 231–240.

[37] X. Liu, Y. Liu, K. Aberer, and C. Miao, "Personalized point-of-interest recommendation by mining users' preference transition," in *Proc. 22nd ACM Int. Conf. Inf. Knowl. Manage.*, 2013, pp. 733–738.

[38] J.-D. Zhang and C.-Y. Chow, "GeoSoCa: Exploiting geographical, social and categorical correlations for point-of-interest recommendations," in *Proc. 38th Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, 2015, pp. 443–452.

[39] J.-D. Zhang, C.-Y. Chow, and Y. Zheng, "ORec: An opinion-based point-of-interest recommendation framework," in *Proc. 24th ACM Int. Conf. Inf. Knowl. Manage.*, 2015, pp. 1641–1650.

[40] H. Gao, J. Tang, X. Hu, and H. Liu, "Content-aware point of interest recommendation on location-based social networks," in *Proc. 29th AAAI Conf. Artif. Intell.*, 2015, pp. 1721–1727.

[41] B. Hu and M. Ester, "Spatial topic modeling in online social media for location recommendation," in *Proc. 7th ACM Conf. Recommender Syst.*, 2013, pp. 25–32.

[42] T. Kurashima, T. Iwata, T. Hoshide, N. Takaya, and K. Fujimura, "Geo topic model: Joint modeling of user's activity area and interests for location recommendation," in *Proc. 6th ACM Int. Conf. Web Search Data Mining*, 2013, pp. 375–384.

[43] Y. Liu, M. Ester, B. Hu, and D. W. Cheung, "Spatio-temporal topic models for check-in data," in *Proc. IEEE Int. Conf. Data Mining*, 2015, pp. 889–894.

[44] Q. Yuan, G. Cong, Z. Ma, A. Sun, and N. M. Thalmann, "Who, where, when and what: Discover spatio-temporal topics for twitter users," in *Proc. 19th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2013, pp. 605–613.

[45] Q. Yuan, G. Cong, K. Zhao, Z. Ma, and A. Sun, "Who, where, when, and what: A nonparametric Bayesian approach to context-aware recommendation and search for twitter users," *ACM Trans. Inf. Syst.*, vol. 33, no. 1, 2015, Art. no. 2.

[46] W. Wang, H. Yin, L. Chen, Y. Sun, S. Sadiq, and X. Zhou, "Geo-SAGE: A geographical sparse additive generative model for spatial item recommendation," in *Proc. 21th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2015, pp. 1255–1264.

[47] H. Wang, Y. Fu, Q. Wang, H. Yin, C. Du, and H. Xiong, "A location-sentiment-aware recommender system for both home-town and out-of-town users," in *Proc. 23rd ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2017, pp. 1135–1143.

[48] M. Xie, H. Yin, H. Wang, F. Xu, W. Chen, and S. Wang, "Learning graph-based POI embedding for location-based recommendation," in *Proc. 25th ACM Int. Conf. Inf. Knowl. Manage.*, 2016, pp. 15–24.

[49] 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 Trans. Knowl. Data Eng.*, vol. 28, no. 8, pp. 1945–1958, Aug. 2016.

[50] H. Wang, N. Wang, and D.-Y. Yeung, "Collaborative deep learning for recommender systems," in *Proc. 21th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2015, pp. 1235–1244.

[51] C. Ma, Y. Zhang, Q. Wang, and X. Liu, "Point-of-interest recommendation: Exploiting self-attentive autoencoders with neighbor-aware influence," in *Proc. 27th ACM Int. Conf. Inf. Knowl. Manage.*, 2018, pp. 697–706.

[52] 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 Trans. Inf. Syst.*, vol. 35, no. 4, pp. 1–28, 2017.

[53] J. Wang, N. Wu, W. X. Zhao, F. Peng, and X. Lin, "Empowering a* search algorithms with neural networks for personalized route recommendation," in *Proc. 25th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2019, pp. 539–547.

[54] F. Zhou, R. Yin, K. Zhang, G. Trajcevski, T. Zhong, and J. Wu, "Adversarial point-of-interest recommendation," in *Proc. World Wide Web Conf.*, 2019, pp. 3462–34 618.

Dichao Li is currently working toward the PhD degree in the State Key Laboratory of Internet of Things for Smart City and Department of Computer and Information Science, University of Macau, China, under the supervision of Prof. Zhiguo Gong. Her current research focuses on data mining and recommender system.

Zhiguo Gong (Senior Member, IEEE) received the PhD degree in computer science from the Institute of Mathematics, Chinese Academy of Science, China. He is currently a professor in State Key Laboratory of Internet of Things for Smart City and Department of Computer and Information Science, University of Macau, China. His research interests include machine learning, data mining, database, and information retrieval.