IRLM: Inductive Representation Learning Model for Personalized POI Recommendation

Junyang Chen, *Member, IEEE*, Mengzhu Wang, Haodi Zhang, Zhenghua Xu, Xueliang Li, Zhiguo Gong, *Senior Member, IEEE*, Kaishun Wu, *Senior Member, IEEE*, and Victor C. M. Leung, *Life Fellow, IEEE*

***Abstract*— With the rapid development of the Internet of Things technology, the concept of smart cities that aims to help residents improve their quality of life has raised much attention in several application areas. In the context of smart cities, the provision of point of interest (POI) recommendations become an important requirement because a wide range of POIs are available for urban dwellers. Location-based social networks (LBSNs) such as Foursquare and Gowalla provide a massive volume of user check-in records that can assist users in choosing new POIs. However, user trajectories are mostly sparse in the real world. For example, users only check in a few POIs, and this makes it difficult to provide recommendations based on limited history trajectories. Though some attempts have adopted auxiliary geographical information to enhance POI recommendation, they still encounter the following problems: 1) the geographical trajectories of users are usually sparse in real-world datasets; 2) users may be more interested in the remote POIs; and 3) the previous models inherently perform transductive learning that cannot handle well the recommendation of unseen users and POIs. To address these problems, we propose an inductive representation learning model (IRLM) for location recommendation. IRLM contains two parts, namely geographic feature extraction and inductive representation learning. IRLM first captures global geographical influences among POIs through a standard Gaussian mixture model (GMM). Then IRLM adopts an attention neural network for the recommendation. Experimental results indicate that our proposed model can achieve superior performance over state-of-the-art models.**

***Index Terms*— Inductive representation learning, location-based social network (LSBN), POI recommendation, smart cities.**

Manuscript received 29 December 2021; revised 15 March 2022, 3 June 2022, and 1 August 2022; accepted 16 August 2022. This work was supported in part by the National Natural Science Foundation of China under Grant 62102265, Grant 61906063, and Grant 61902249; in part by the Open Research Fund from the Guangdong Laboratory of Artificial Intelligence and Digital Economy (SZ) under Grant GML-KF-22-29; in part by the Natural Science Foundation of Guangdong Province of China under Grant 2022A1515011474; in part by the Natural Science Foundation of Hebei Province, China, under Grant F2021202064; and in part by the Science and Technology Development Fund, Macau, China, under Grant 0068/2020/AGJ and Grant 0045/2019/A1. *(Corresponding author: Victor C. M. Leung.)*

Junyang Chen is with the College of Computer Science and Software Engineering and the Guangdong Laboratory of Artificial Intelligence and Digital Economy (SZ), Shenzhen University, Shenzhen 518060, China (e-mail: junyangchen@szu.edu.cn).

Mengzhu Wang is with the College of Computer, National University of Defense Technology, Changsha 410073, China (e-mail: dreamkily@gmail.com).

Haodi Zhang, Xueliang Li, Kaishun Wu, and Victor C. M. Leung are with the College of Computer Science and Software Engineering, Shenzhen University, Shenzhen 518060, China (e-mail: hdzhang@szu.edu.cn; lixueliang@szu.edu.cn; wu@szu.edu.cn; vleung@ieee.org).

Zhenghua Xu is with the State Key Laboratory of Reliability and Intelligence of Electrical Equipment, Hebei University of Technology, Tianjin 300130, China (e-mail: zhenghua.xu@hebut.edu.cn).

Zhiguo Gong is with the State Key Laboratory of Internet of Things for Smart City, Department of Computer Information Science, University of Macau, Macau, China (e-mail: fstzgg@um.edu.mo).

Digital Object Identifier 10.1109/TCSS.2022.3201053

I. Introduction

SMART city is a concept that has gained a lot of attention in recent years as it aspires to improve citizen quality of life [1]. With the fast-growing Internet of Things technology, it is more convenient for people to share their personal information. The location-based social network (LBSN) services of sharing check-in records, such as Foursquare and Yelp, are emerging and become increasingly popular in the real world [2]. The massive volume of daily generated data is valuable for researchers to extract users' mobility patterns and provide personalized recommendation services. In LBSN, the point of interest (POI) recommendation is one of the most important applications [3], [4], [5], [6], which can assist users to explore potential interesting places and show its value in decision making of business areas.

The previous recommender algorithms are often designed under the assumption that users' check-in records present their preference ratings of POIs. For example, Ye *et al.* [7] adopt a friend-based collaborative filtering method to calculate the user's ranking of new POIs, and Wang *et al.* [6] utilize a matrix completion approach to incorporate geographical features with user-POI affinity information. However, most of the existing POI recommendation models, including [5], [6], [7], are inherently performing transductive learning which requires all users and POIs should be shown up in the training process. Thus, it is hard for those models to make effective recommendations for unseen users and POIs. Applied to the recommendation community, this challenge is also known as the cold-start problem, and work has been done on estimating the interests of new users based on factors like geographical influences. Specifically, they follow a common assumption that users prefer to visit nearby POIs than ones with far distance [8]. For instance, Ye *et al.* [9] model the relationship between users and POIs with a power-law distribution and Wang *et al.* [10] make a more comprehensive comparison among techniques of using exponential, power-law, and hyperbolic functions. Nevertheless, all these approaches assume that the influences of POIs on users follow certain distributions, which would encounter the following problems: 1) the users' trajectories are usually sparse in the real world, thus it is difficult to estimate the influence distributions of POIs; 2) users may be more interested in some remote POIs, which is against the assumption mentioned before; and 3) more importantly, when new users and POIs are emerging, these models inherently perform transductive learning that cannot deal with the unseen ones not being trained before.

To solve the problems outlined above, we propose an inductive representation learning model (IRLM) for personalized location recommendations. Specifically, IRLM consists of two parts, geographic feature extracting with a standard Gaussian mixture model (GMM) and inductive representation learning for the recommendation. GMM is widely used for representing normally distributed clusters within an overall data distribution. The advantage of the mixture model is that they do not require which cluster a data point belongs. It allows the model to learn the probability distribution of clusters automatically [11]. As such, we adopt it as the first part that aims to capture the geographical influences among POIs in a global context, then utilize these influences as additional features for the recommendation. The second part is an inductive learning model that is based on graph convolutional neural networks (GCNs) to learn the function of aggregating the neighbor information (local context) of both users and POIs from an attention convolutional perspective. Before performing the graph learning, since a single user may only contain sparse historical trajectories, we employ a graph reconstruction method that can build edges between users and POIs with random walks [12]. More concretely, we can obtain the edges of user-user, user-POI, and POI-POI with a sliding window and reconstruct a more dense graph to alleviate the mentioned sparsity problem. In this way, our method can learn personalized trajectory recommendations beyond the geographical information and be aware of the interest of remote POIs with information propagation from multihop neighbors. Besides, IRLM enables fuse these two parts to perform representation learning that can efficiently generate embeddings for the unseen users and POIs to benefit cold-star POI recommendation. In a conclusion, this article provides the following contributions.

1) We propose an IRLM for POI recommendation, which jointly makes use of both the global and local contexts of user and POI information. To the best of our knowledge, this is the first attempt to investigate how to effectively generate embeddings for unseen users and POIs to benefit the recommendation.
2) We first employ the standard GMM to capture the global geographical influences among POIs. Then, we employ a graph reconstruction method and adopt an attention neural network to learn the function of aggregating the neighbor information of users and POIs. By doing so, we can effectively alleviate the sparse problem of user trajectories and extend the recommended POI scope to include the remote ones. More importantly, we can generate embeddings for the unseen users and POIs with the learned aggregation function.
3) We compare the performance of the proposed IRLM with several baselines of POI recommendations on two real-world datasets of LBSN to demonstrate its effectiveness. The results of experiments show that IRLM achieves better performance in subsequent tasks at different training scales.

The source code and data will be available after acceptance.[1] The remainder of the article is arranged as follows. Section II gives the introduction of the proposed model. Experimental analyses are presented in Section III. Section IV reviews the related work and Section V summarizes this work.

II. Proposed IRLM Model

We provide the formulation and notations of the problem in this section. Following that, we will present an overview of the proposed IRLM framework and describe its components in detail.

A. Problem Formulation and Notations

In POI recommendation tasks, we present the user check-in data as a graph network $G = (U, P, F, E)$, where U denotes a set of users, P represents a set of POIs, F is the associated features of POIs, and $E \subseteq U \times P$ denotes the set of edges generated based on user check-in behaviors. In this article, we aim to learn a low-dimensional embeddings $W \in \mathbb{R}^d$ for each user and POI, where $d \ll |U|$ and $d \ll |P|$. Users and POIs, where similar users and POIs are assigned to nearby areas, are described in the low-dimensional space that preserves the relations between them, thereby making the learned representation helpful for the next-to-follow graphical applications, such as POI recommendation [6] and data visualization [13].

B. Overview of IRLM

As the previous studies [9], [10] showed, there are geographical relationships among users and POIs that can be employed to benefit the performance of POI recommendation. Without loss of generality, our proposed IRLM contains two parts: 1) geospatial features are extracted via GMMs to leverage their geographic influence and 2) a graph attention convolutional neural network for inductive representation learning. The whole framework of the proposed model is shown in Fig. 1. The following details will be provided regarding the training components.

C. Step 1: Geographic Feature Extraction

As shown in the top part of Fig. 1, to begin with, we first perform the GMM model [11] to extract the geographic features of user check-in data. Specifically, each POI is denoted as a tuple of latitude and longitude. After conducting an unsupervised clustering method with GMM, we can gain the geographic influences of POIs using Gaussian mixture distributions. Then we can utilize a vector $[x_1, x_2, \ldots, x_{|K|}]$ to present the feature of a user or POI, where K denotes the set of

[1] https://github.com/junyachen/IRLM

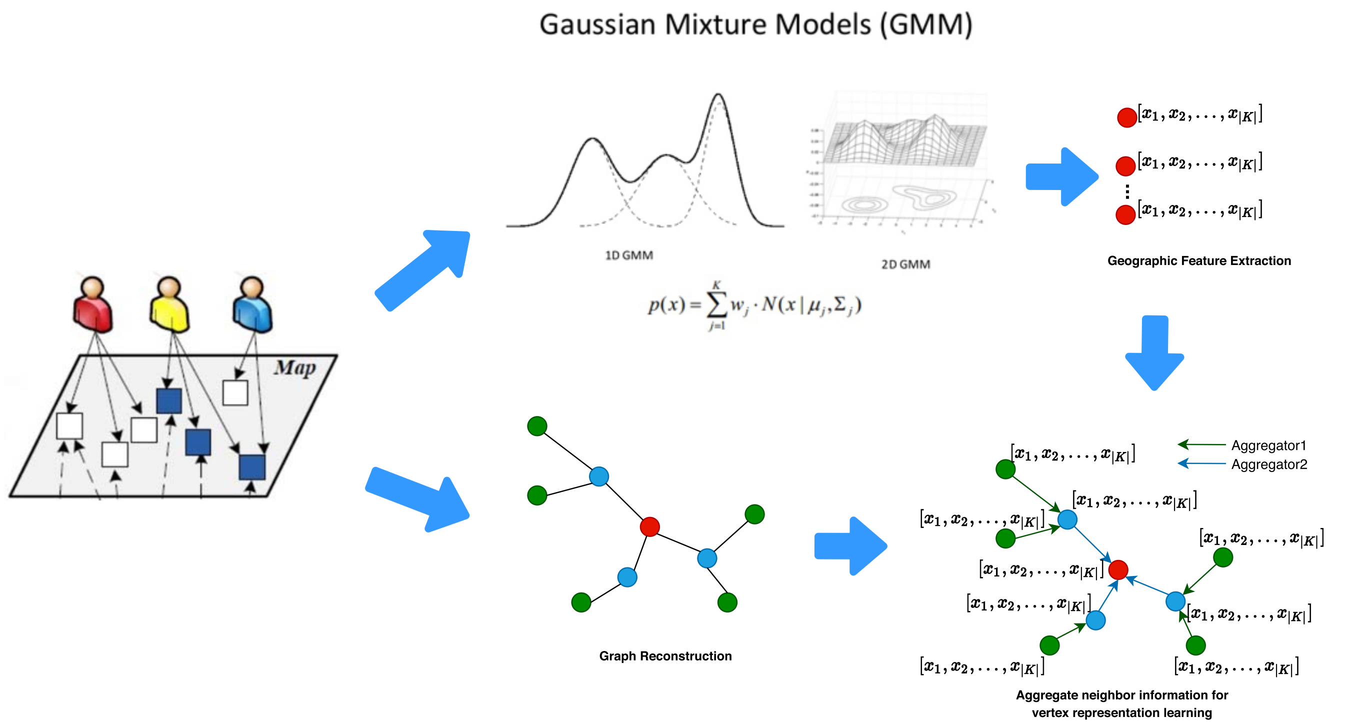


Fig. 1. Overview of the proposed IRLM model. The leftmost figure presents the user check-in data. The top part of the right figure denotes that we exploit the GMM model to extract the geographic features of user check-in data. And the bottom half figure denotes the graph reconstruction and the inductive representation learning. The detailed description can be found in Section II.

clusters and x_k represents the probability belonging to cluster k. Note that since a user may visit multiple POIs, we simply average its vectors to aggregate the feature information. The obtained features can be formulated as follows:

$$x_k \sim p(x) = \sum_{k=1}^{K} w_k \cdot N(x|\mu_k, \Sigma_k) \tag{1}$$

where μ_k and Σ_k denote the mean and covariance values of cluster k, respectively, w_k represents the learning weight, and x denotes the latitude and longitude information of geographical position.

D. Step 2: Graph Reconstruction

As shown in the bottom part of Fig. 1, we present all users and POIs as vertices and build edges between them. Then, we conduct the random walk [12] to construct a user check-in graph. Specifically, we first obtain a set of walk sequences starting from each user. Next, we can gain the edges of user-user, user-POI, and POI-POI with a sliding window and reconstruct a more dense graph. Compared with the original edges only containing user-POI check-in information, our proposed graph reconstruction method consider more types of relations between users and POIs which can benefit the following aspects: 1) alleviating the sparse user trajectories and 2) explicitly modeling the long-distance influence propagated by the initial check-in relations. Finally, we perform inductive representation learning with the features extracted from Step 1, and the graph reconstructed from Step 2.

E. Step 3: Inductive Representation Learning

To perform inductive representation learning effectively, the designed model must allow embedded representations to be generated efficiently for the unseen vertices, such as users and POIs. However, most of the existing POI recommendation models are inherently perform transductive learning that requires all users and POIs being shown up in the training process. Inspired by the recent upstarts of GCN [14] and its variants [15], [16] which have shown promising performance in graph mining areas, we adopt an attention aggregator for representation learning. The detail is defined by

$$\mathbf{v}_i^l = \sigma\left(\sum_{v_j \in \mathcal{N}_{v_i}} \alpha_{v_i,v_j} \cdot \mathbf{W}^l \cdot \mathbf{v}_j^{l-1}\right) \tag{2}$$

where $\mathbf{v} \in \mathbb{R}^d$ denotes the hidden embedding with dimension d, σ is the LeakyReLU function [16], l is the number of layers, $\mathcal{N}(v_i)$ denotes the neighbors of vertex v_i, $\mathbf{W}^l \in \mathbb{R}^{d\times d}$ is a shared weight matrix at layer l, and α_{v_i,v_j} denotes a weight coefficient between vertex v_i and v_j (the details will be introduced in the followings). Then, the representation of node v_i^l is obtained by aggregating the messages passing from its neighbors at layer $l-1$ with the weight coefficients, which is defined by

$$\alpha_{v_i,v_j} = \frac{\exp\left(\sigma\left(\mathbf{a}^T \cdot \left[\mathbf{W}^{l-1}\mathbf{v}_i^{l-1} \| \mathbf{W}^{l-1}\mathbf{v}_j^{l-1}\right]\right)\right)}{\sum_{v_k \in \mathcal{N}_{v_i}} \exp\left(\sigma\left(\mathbf{a}^T \cdot \left[\mathbf{W}^{l-1}\mathbf{v}_i^{l-1} \| \mathbf{W}^{l-1}\mathbf{v}_k^{l-1}\right]\right)\right)} \tag{3}$$

where $\mathbf{W}^{l-1} \in \mathbb{R}^{d\times d}$ is a shared weight matrix at layer $l-1$, and $\mathbf{a} \in \mathbb{R}^{2d}$ denotes a weight vector for the concatenation operation $||$. Note that in our article, we uniformly denote all entities, including users U and POIs P, as the vertices V, and the neighbor relations are modeled by user check-in behaviors (the details can be referred to Section II-A). We let $\mathbf{v}^0 \leftarrow \mathbf{x}_v$, where $\mathbf{x}$ denotes the geographical features extracted by (1). Then, in the training process, we not only learn the vertex embeddings but also learn the attention aggregation function which benefits the inductive learning. More concretely, when an unseen entity of users or POIs comes, we can present it by aggregating its neighbor embeddings that exist in the previous training with the learned aggregator (parameterized by $\mathbf{W}^l$). In addition, to learn vertex representations in an unsupervised way, Skip-Gram with negative sampling (NS) [17] provides an optimization method which keeps a target vertex far from one's negative samples in the embedding space but close to its neighbors. Following is a definition of the objective function:

$$\mathcal{J}(v_i) = -\log\big(\sigma\left(\mathbf{v}_p^T \cdot \mathbf{v}_i\right)\big) - \sum_{j=1}^{N} \mathbb{E}_{v_j \sim P_{\mathrm{NS}}(v)} \log\big(\sigma\left(-\mathbf{v}_j^T \cdot \mathbf{v}_i\right)\big) \tag{4}$$

where v_i, v_p, and σ represent a target vertex, its neighboring vertex, and the Sigmoid function, respectively, $\sigma(x) = 1/(1+\exp(-x))$, the NS distribution is denoted by $P_{\mathrm{NS}}(v)$ (the details will be given below), negative samples are drawn from $P_{\mathrm{NS}}(v)$ to form v_j, N denotes the number of negative samples used for training, and $\mathbf{v}_p$, $\mathbf{v}_i$, and $\mathbf{v}_j$ are the embeddings at the final layer that are aggregated from the features involving their local neighbors. In this objective, we aim to encourage all nearby vertices to have similar embeddings but to be distinct from their negative counterparts. Moreover, the distribution of NS $P_{\mathrm{NS}}(v)$ can be formulated in the following manner:

$$P_{\mathrm{NS}}(v) = \frac{f_v^{\beta}}{\sum_{v\in V} f_v^{\beta}} \tag{5}$$

where the vertex degree of a graph is f_v, and the empirical degree power is β, which is usually set to 3/4 [17], [18]. For its low computational complexity, NS is one of the most widely used ways of optimizing objective functions for unsupervised representation learning [17], [18], [19], [20].

F. IRLM Algorithm

Until now, we have explained the entire training process of IRLM, illustrated in Fig. 1. More concretely, we present Algorithm 1 to demonstrate the implementation steps. First, we perform the GMM [11] to extract the geographical features from user check-in data (Line 2). Next, we obtain multiple types of relations among users and POIs by conducting random walks with a sliding window (Line 3). Then, we start inductive representation learning by employing NS [17] as the objective function [referring to (4)] and using stochastic gradient descent [21] for optimization (Lines 4–16). Until convergence has been reached, the previous step is repeated.

Algorithm 1 Training Process of IRLM

Input: User check-in data $G = (U, P, F, E)$, dimension d, geographical cluster K, and number of neural network layer $|L|$
Result: Vertex representation $\mathbf{v}, \forall v \in V$ and $V = \{U, P\}$
1 **begin**
2 **Geographic feature extraction:** performing the GMM to extract the geographic features of user check-in data. Then, for each vertex v, we can obtain its global geographical influence denoted as $\mathbf{x}_v = [x_1, x_2, \ldots, x_{|K|}]$, where K represents the set of clusters;
3 **Graph Reconstruction:** Reconstructing a user check-in graph by conducting random walks to obtain multiple types of relations between users and POIs including user-user, user-POI, and POI-POI;
4 **Inductive representation learning**;
5 Randomly initialize the learning parameters $\mathbf{W}^l, \forall l \in \{1, \ldots, |L|\}$;
6 $\mathbf{v}^0 \leftarrow \mathbf{x}_v, \forall v \in V$;
7 **while** *not converge* **do**
8 **for** $l \in [1, |L|]$ **do**
9 **for** $v \in V$ **do**
10 $\mathbf{v}_i^l = \sigma\left(\sum_{v_j \in \mathcal{N}_{v_i}} \alpha_{v_i, v_j} \cdot \mathbf{W}^l \cdot \mathbf{v}_j^{l-1}\right)$ referring to Eq. (2);
11 **end**
12 $v^l \leftarrow v^l/||v^l||_2, \forall v \in V$;
13 **end**
14 $\mathbf{v} \leftarrow \mathbf{v}^{|L|}, \forall v \in V$;
15 Optimization is carried out using stochastic gradient descent [21] with NS [17] as the objective function (Eq.(4));
16 **end**
17 **end**

III. Experiments

In experiments, in order to evaluate the effectiveness of our IRLM in terms of unseen POI recommendation, we conduct experiments including quantitative and qualitative analysis on two widely used LSBN datasets. Moreover, we conduct an ablation test to investigate the effect of variant aggregators in the IRLM. Besides, we also investigate the sensitivity of IRLM to the settings of key parameters.

A. Datasets

With the statistics shown in Table I, we run experiments on four widely used LSBN datasets. All of the above datasets are available online.[2]

Foursquare records digital footprints of users when using mobile phones to study the problems of personalized location recommendation. We use a portion of data collected in Singapore from August 2010 to July 2011. A total of 194 108 user check-in trajectories have been recorded.

[2]https://github.com/junyachen/GAIMC/tree/master/upload

TABLE I
STATISTICS OF DATASETS

	Foursquare	Gowalla
# of users	2,321	10,162
# of POIs	5,596	24,250
# of check-ins	194,108	456,988
Time window	2010/08 - 2011/11	2009/02 - 2010/10
Avg. check-ins of users	30.62	17.06
Avg. check-ins of POIs	12.71	7.18

Gowalla is another popular LSBN dataset. We have collected a total of 456 988 check-ins of these users over the period of February 2009 to October 2010.

B. Baseline Models

Our method is compared to three baseline approaches that are useful for representation learning in POI recommendation. In addition to these baseline methods, there are many other learning methods that we will not discuss here due to either their inferior performance as evidenced by corresponding papers or their transductive nature which is incompatible with a POI inductive representation learning approach. The baselines are described as follows.

DeepWalk [12] is an efficient representation learning approach by conducting random walks on networks to generate vertex sequences and using the Skip-Gram [17] to learn vertex embeddings. In the experiment, we employ DeepWalk on the constructed graph to obtain the vertex representations.

***K*-means++** [22] is a classical yet effective model. By augmenting K-means with a simple, randomized seeding technique, K-means++ is competitive with the optimal clustering. Besides, its simplicity and speed are very appealing in practice. We adopt this model to obtain the probability distributions over the clusters in the user check-in trajectories. The learned distributions can be regarded as the representations of users and POIs.

GMM [11] is a widely used Bayesian mixture model that can automatically infer the number of clusters and learn the probability distributions of input data. We include GMM as one of the baseline models because we apply it to extract the geographic features of user check-in data for our model.

GAIMC [6] is a geography-aware inductive matrix completion approach for personalized POI recommendation. The technique comprises two steps, including the extraction of geographic features through the use of a GMM and the completion of the matrix inductively with recommendations. In general, it is a matrix factorization-based method that cannot learn the nonlinear deep relations among vertices.

C. Parameter Settings and Evaluation Metrics

For K-means++ model requiring a predefined cluster number, we use the one automatically detected by GMM. For DeepWalk, we follow [12] and set the window size, the walk length, and the number of walks as 10, 30, and 50, respectively. Since IRLM exploits an aggregated two-layer neural network, we follow [15] by setting the number of network layers $l = 2$, and the neighborhood sample sizes of layers $S_1 = 25$ and $S_2 = 10$, respectively. Furthermore, we perform stochastic gradient descent with the Adam optimizer [21] using the initial learning rate $1e - 3$. In order to make all comparisons fair, we uniformly set $d = 128$ as the embedding size for all models.

To evaluate the performance of POI recommendation with the learned representations, it is natural to predict a link between users and POIs. We adopt a standard evaluation metric area under curve (AUC) [23], which represents the probability that vertices in a random unobserved link are more similar than those in a random nonexistent link. The AUC metric has been widely used in recommendation tasks [6]. When the prediction results perfectly match the ground truth, AUC value will be one, otherwise, it will be zero.

D. Evaluation on POI Recommendation

In this part, we conduct the POI recommendation with link prediction to verify the learned representations of our proposed IRLM. Tables II and III show the comparison results with different training ratios on Foursquare and Gowalla, respectively. A boldfaced font is used to highlight the highest scores. Observations derived from these tables are as follows.

1) We propose a model, IRLM, which performs significantly better than other models on all datasets with different training ratios, showing how our method effectively aggregates neighbor information of vertices (such as users and POIs) to acquire representations.
2) In general, the order of AUC performance is IRLM $>$ GAIMC $>$ K-means++ $>$ GMM $>$ DeepWalk. We may reasonably conclude that exploiting the geographical information can generate more positive effort than merely using the check-in trajectories, i.e., DeepWalk. Moreover, we can observe that GMM performance fluctuates along with the training ratios on both datasets. One possible reason is that the GMM method is sensitive to the distributions of user check-in data. In contrast, our method can achieve consistent improvements, which demonstrates that the advantage of IRLM comes beyond the prior knowledge from GMM (we apply it to extract geographic information as the input features of vertices and more details can be referred to Algorithm 1).
3) More concretely, IRLM can achieve 32.21%, 15.33%, 15.09%, and 5.24% performance gains over DeepWalk, K-means++, GMM, and GAIMC on average of training ratios in Foursquare, respectively. Besides, IRLM can also obtain 99.77%, 20.95%, 28.02%, and 7.85% improvements over the baselines in Gowalla. We conclude that our proposed IRLM can gain more benefits by jointly learning the constructed graph structures with the extracted geographic features.

E. Convergence Analysis

In this section, we perform the convergence analysis of Algorithm 1. As shown in Fig. 2, we report the training losses

TABLE II
RESULTS OF AUC COMPARISON ON FOURSQUARE DATASET

Percentages of Training Set	10%	20%	30%	40%	50%	60%	70%	80%	90%
DeepWalk	0.4161	0.4376	0.6268	0.6368	0.6363	0.6358	0.6362	0.6342	0.6200
K-means++	0.6496	0.6630	0.6628	0.6676	0.6716	0.6780	0.6763	0.6835	0.7001
GMM	0.6452	0.6651	0.6638	0.6932	0.6604	0.6810	0.6747	0.6940	0.6874
GAIMC	0.7184	0.7355	0.7407	0.7491	0.7470	0.7402	0.7387	0.7394	0.7237
IRLM	**0.7217**	**0.7452**	**0.7609**	**0.7780**	**0.7876**	**0.7939**	**0.7965**	**0.7962**	**0.8002**

TABLE III
RESULTS OF AUC COMPARISON ON GOWALLA DATASET

Percentages of Training Set	10%	20%	30%	40%	50%	60%	70%	80%	90%
DeepWalk	0.4797	0.4455	0.4873	0.4161	0.3863	0.4357	0.3929	0.3977	0.4286
K-means++	0.6739	0.6826	0.7034	0.7056	0.7114	0.7285	0.7274	0.7223	0.7365
GMM	0.6503	0.6516	0.6752	0.6902	0.7091	0.7307	0.5697	0.7444	0.6172
GAIMC	0.7407	0.7534	0.7831	0.8028	0.8071	0.8021	0.8257	0.8364	0.8169
IRLM	**0.7510**	**0.8249**	**0.8514**	**0.8713**	**0.8821**	**0.8851**	**0.8859**	**0.8866**	**0.8923**

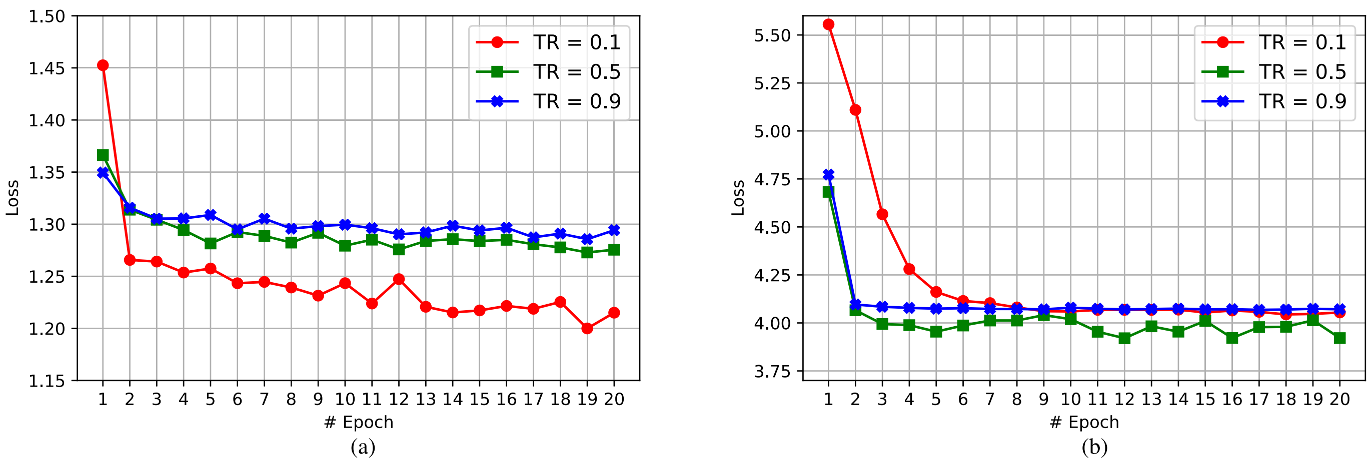


Fig. 2. Convergence analysis on (a) Foursquare and (b) Gowalla datasets. We report their training losses in terms of the epochs. "TR" represents the training percentages of data.

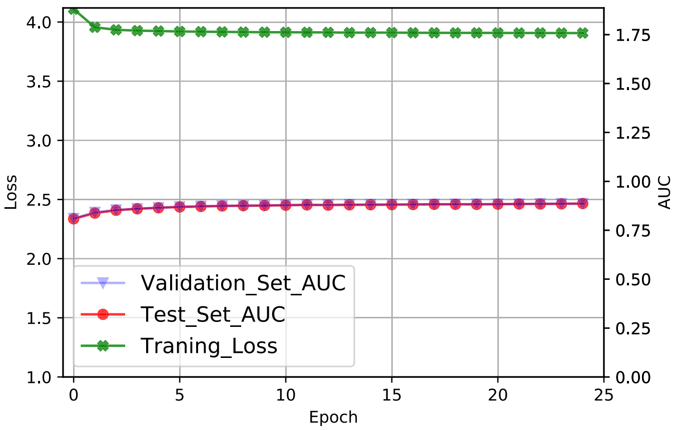
with the epochs on Foursquare and Gowalla, respectively, where we can see that our method can generally achieve convergence after ten epochs with different training ratios.

Moreover, we use ten-fold cross-validations for these two datasets. Specifically, we adopt the percentages of 6:2:2 to partition the training set, the validation set, and the testing set. The experimental results are reported in Fig. 3, where we have the following observation.

From the training loss curves of Fig. 3(a) and (b), our method can obtain convergence after ten epochs on Foursquare and Gowalla. At the same time, both the performance trends of the test set and validation set are consistent on these datasets. More concretely, the AUC performances of the validation and test sets become stable around 0.8729 and 0.8734 after ten epochs. One possible reason is that our model is an unsupervised learning method without node labels, as such, it is not quite easy to be over-fitting in the training process.

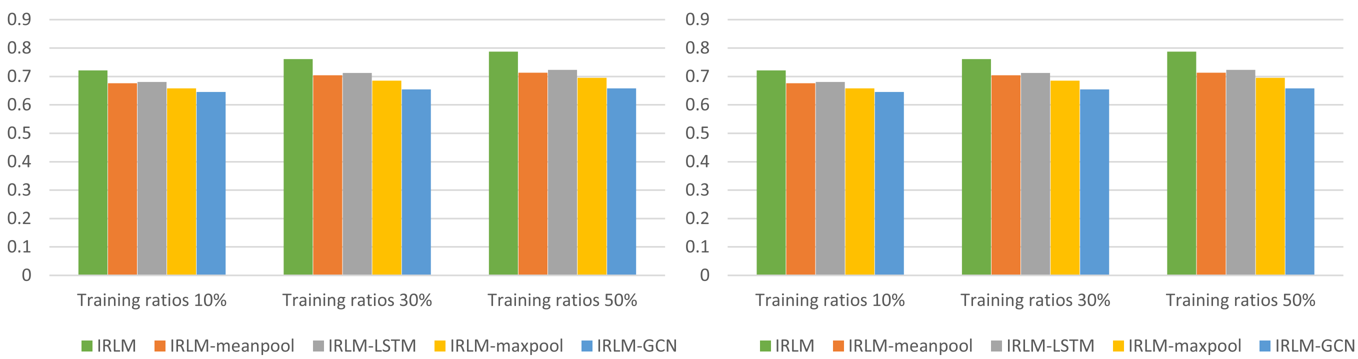
F. Ablation Test of Variant Aggregators for Representation Learning

In this part, we try to investigate the effect of variant aggregators in the proposed IRLM. Apart from using an attention aggregator [as mentioned in (2)], we also exploit four types of aggregate functions including GCN [14], meanpool, maxpool, and LSTM [24] to conduct an ablation test for representation learning. We report the AUC performance of these methods with training ratios $\{0.1, 0.3, 0.5\}$ on Foursquare and Gowalla, as shown in Fig. 4. Following are our observations. From Fig. 4(a), we can observe that the order of AUC performance is IRLM > IRLM-LSTM > IRLM-meanpool > IRLM-maxpool > IRLM-GCN. More concretely, IRLM outperforms IRLM-LSTM by 6.05%, 6.87%, and 8.96% on three sets of training ratios, respectively. Besides, from Fig. 4(b), we can see that the performance order is IRLM > IRLM-LSTM > IRLM-GCN > IRLM-maxpool >



(a) (b)

Fig. 3. N-fold cross-validations for the convergence analysis on (a) Foursquare and (b) Gowalla datasets. We adopt the percentages of 6:2:2 to partition the training set, the validation set, and the testing set.



(a) (b)

Fig. 4. Ablation study of variant aggregators on AUC performance over (a) Foursquare and (b) Gowalla.

TABLE IV
RESULTS OF THE COLD-START PERFORMANCE ON FOURSQUARE

Foursquare	Avg. check-ins of users in the traning set	Avg. check-ins of POIs in the training set	New users in the test set	New POIs in the test set	AUC
IRLM w/o Graph Reconstruction	7.57	3.71	238	1347	0.7005
IRLM	9.35	5.42	238	1347	0.7217
Improv.	23.51%	46.09%	-	-	3.03%

IRLM-meanpool, and IRLM outperforms IRLM-LSTM by 4.48%, 6.33%, 6.65%, respectively. In general, we can conclude that our method is more effective than the other aggregators based on the comparison.

G. Evaluation on the Cold-Start Performance

In this section, we aim to conduct the performance comparisons of the cold-start. As shown in Tables IV and V, we report the average check-ins of users and the average check-ins of POIs in the training sets, new users, and new POIs in the test sets, and the AUC performances on Foursquare and Gowalla, respectively. More concretely, as shown in Table IV, our IRLM obtains 23.51%, 46.09%, and 3.03% improvements over IRLM without graph reconstruction (refer to Section II-D) on the density of the average check-ins and AUC performance, respectively. Besides, as shown in Table V, IRLM achieves up to 35.40%, 67.69%, and 1.36% improvements accordingly on Gowalla. These demonstrate that our method can get a boost from the graph reconstruction and also can perform well in the cold-start scenario.

H. Parameter Sensitivity

Fig. 5 shows the influence of the key parameters, neighbor sample size (as mentioned in Section III-C), on the AUC performance of our proposed method. Here, we use the IRLM method for evaluating the parameter influence on Foursquare dataset. Specifically, the axes are S_1 and S_2, with their numbers varying in $\{10, 15, 20, 25, 30\}$. The results indicate that our model is generally robust to the neighbor sample size settings, producing the best performance when $S_1 = 15$ and $S_2 = 15$.

TABLE V
RESULTS OF THE COLD-START PERFORMANCE ON GOWALLA

Gowalla	Avg. check-ins of users in the traning set	Avg. check-ins of POIs in the training set	New users in the test set	New POIs in the test set	AUC
IRLM w/o Graph Reconstruction	4.52	2.29	2461	9063	0.7409
IRLM	6.12	3.84	2461	9063	0.7510
Improv.	35.40%	67.69%	-	-	1.36%

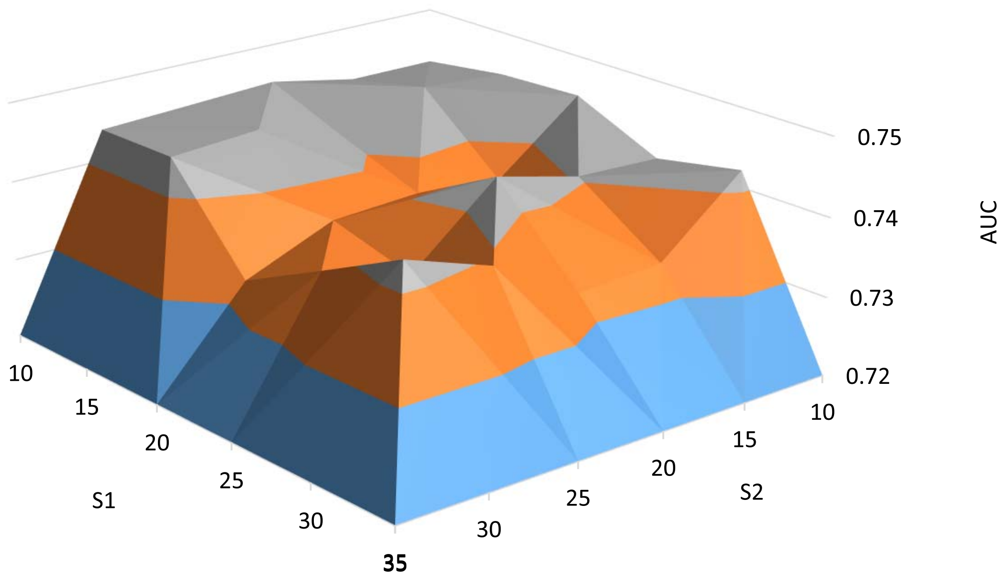


Fig. 5. In this case, the parameter influence is evaluated using IRLM on the Foursquare dataset. The variables S_1 and S_2 are both used as axes, and their values are continuously varied in a range {10, 15, 20, 25, 30} in order to evaluate how they affect AUC performance.

IV. Related Work

In this part, we demonstrate the related work of POI recommendation with LBSNs.

A. Spatio-Temporal Influences in POI Recommendation

To begin with, classical recommender systems conducting POI recommendations are mostly based on the explicit ratings of POIs [25] or exploiting users' social preferences as data supplements [26]. However, in the real world, user check-in data without rating information and user social preferences are more common and easy to acquire. Therefore, many works based on trajectories [6], [27] are proposed in recent years. They regard the POIs as the items in E-commerce, then, many traditional recommendation methods can be adopted for POI recommendation. For example, Ye *et al.* [7] incorporate a collaborative filtering approach with friendly relations. Berjani and Strufe [28] propose a more general matrix factorization method with regularization on user check-in data. Nevertheless, these models do not take into account the geographical influences which are useful for improving the performance of POI recommendation.

To estimate the interests of new users via geographical influences, Ye *et al.* [9] employ a power-law distribution to model the relationship between users and POIs. Then, Wang *et al.* [10] make further improvement by employing more complicated distributions. All these models follow the assumption that the influences of POIs on users are under certain distributions. However, in real-world data, the user check-in trajectories are usually sparse. As such, it is difficult to estimate the impact of POI distributions for users with limited user behavior tracks. Moreover, these mentioned models cannot deal with the case that users may be more interested in remote POIs. Besides, all these models are transductive learning methods, thereby being incapable of dealing with the unseen users and POIs untrained before.

B. Addressing Sparse User Trajectories in POI Recommendation

To address the sparse user check-in trajectories, some efforts have been devoted to exploiting supplementary data. For instance, gSCorr [29] proposes to use social correlations with limited user historical behaviors for solving the cold-start location recommendation problem. Similar geo-social correlation models such as [30], [31], [32] are also proposed to recommend the target user with the set of locations visited by their friends. In general, these methods address the cold-star problem based on the following assumptions. First, the target users and their friends share common interests. The target users are interested in the locations with high preferences by their friends. However, in practice, getting sufficient context information (i.e., social ties of users) is difficult while pure POI check-in records are more prevalent [33].

To address the sparse problem with the general user check-in data, in this article we propose an IRLM for personalized location recommendation. Moreover, our model can conduct inductive learning which is suitable for dealing with the unseen users and POIs not being trained before.

V. Conclusion

The goal of this article is to develop an IRLM for personalized location recommendations. Specifically, our model can jointly consider the global and local perspectives of users and POI information. We first extract the geographical features by taking all check-in data into account. Then, we learn the user and POI embeddings with an attention convolutional network by considering the aggregation of vertex neighbors. In general, our IRLM can efficiently generate embeddings for the users and POIs that have not been visited before in the training process. Therefore, when new users and POIs are emerging, IRLM can benefit POI recommendations. In our experiments on real-world LBSN datasets, the effectiveness of our method is demonstrated at different training scales.

References

[1] V. Albino, U. Berardi, and R. M. Dangelico, "Smart cities: Definitions, dimensions, performance, and initiatives," *J. Urban Technol.*, vol. 22, no. 1, pp. 3–21, 2015.
[2] J. S. Rosen, "System and method for location based social networking," U.S. Patent 8 019 692, Sep. 13, 2011.
[3] Y. Yu and X. Chen, "A survey of point-of-interest recommendation in location-based social networks," in *Proc. Workshops 29th AAAI Conf. Artif. Intell.*, 2015, pp. 1–30.
[4] 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, Jun. 2017.
[5] W. Wang, J. Chen, J. Wang, J. Chen, J. Liu, and Z. Gong, "Trust-enhanced collaborative filtering for personalized point of interests recommendation," *IEEE Trans. Ind. Informat.*, vol. 69, no. 9, pp. 6124–6132, Dec. 2019.
[6] W. Wang, J. Chen, J. Wang, J. Chen, and Z. Gong, "Geography-aware inductive matrix completion for personalized point-of-interest recommendation in smart cities," *IEEE Internet Things J.*, vol. 7, no. 5, pp. 4361–4370, May 2020.
[7] M. Ye, P. Yin, and W.-C. Lee, "Location recommendation for location-based social networks," in *Proc. 18th SIGSPATIAL Int. Conf. Adv. Geographic Inf. Syst.*, 2010, pp. 458–461.
[8] W. Tobler, "A computer movie simulating urban growth in the Detroit region," *Econ. Geography*, vol. 46, no. 2, pp. 234–240, Feb. 1970.
[9] 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. (SIGIR)*, 2011, pp. 325–334.
[10] H. Wang, H. Shen, W. Ouyang, and X. Cheng, "Exploiting poi-specific geographical influence for point-of-interest recommendation," in *Proc. IJCAI*, 2018, pp. 3877–3883.
[11] C. E. Rasmussen, "The infinite Gaussian mixture model," in *Proc. Adv. Neural Inf. Process. Syst.*, 2000, pp. 554–560.
[12] B. Perozzi, R. Al-Rfou, and S. Skiena, "DeepWalk: Online learning of social representations," in *Proc. 20th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2014, pp. 701–710.
[13] Q. Dai, Q. Li, J. Tang, and D. Wang, "Adversarial network embedding," in *Proc. 32nd AAAI Conf. Artif. Intell.*, 2018, pp. 1–8.
[14] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," 2016, *arXiv:1609.02907*.
[15] W. Hamilton, Z. Ying, and J. Leskovec, "Inductive representation learning on large graphs," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 1024–1034.
[16] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lió, and Y. Bengio, "Graph attention networks," 2017, *arXiv:1710.10903*.
[17] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," 2013, *arXiv:1301.3781*.
[18] J. Tang, M. Qu, M. Wang, M. Zhang, J. Yan, and Q. Mei, "Line: Large-scale information network embedding," in *Proc. 24th Int. Conf. World Wide Web*, 2015, pp. 1067–1077.
[19] A. Grover and J. Leskovec, "Node2vec: Scalable feature learning for networks," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2016, pp. 855–864.
[20] L. F. Ribeiro, P. H. Saverese, and D. R. Figueiredo, "Struc2vec: Learning node representations from structural identity," in *Proc. 23rd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2017, pp. 385–394.
[21] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.
[22] S. Vassilvitskii and D. Arthur, "k-means++: The advantages of careful seeding," in *Proc. 18th Annu. ACM-SIAM Symp. Discrete Algorithms*, 2006, pp. 1027–1035.
[23] J. A. Hanley and B. J. McNeil, "The meaning and use of the area under a receiver operating characteristic (ROC) curve," *Radiology*, vol. 143, no. 1, pp. 29–36, 1982.
[24] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.
[25] D. Gavalas, C. Konstantopoulos, K. Mastakas, and G. Pantziou, "Mobile recommender systems in tourism," *J. Netw. Comput. Appl.*, vol. 39, pp. 319–333, Mar. 2014.
[26] W. Ji, X. Meng, and Y. Zhang, "STARec: Adaptive learning with spatiotemporal and activity influence for POI recommendation," *ACM Trans. Inf. Syst.*, vol. 40, no. 4, pp. 1–40, 2021.
[27] H. Li, Y. Ge, R. Hong, and H. Zhu, "Point-of-interest recommendations: Learning potential check-ins from friends," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2016, pp. 975–984.
[28] B. Berjani and T. Strufe, "A recommendation system for spots in location-based online social networks," in *Proc. 4th Workshop Social Netw. Syst.*, 2011, pp. 1–6.
[29] H. Gao, J. Tang, and H. Liu, "Addressing the cold-start problem in location recommendation using geo-social correlations," *Data Mining Knowl. Discovery*, vol. 29, no. 2, pp. 299–323, 2015.
[30] H. Gao and J. Tang, "GSCorr: Modeling geo-social correlations for new check-ins on location-based social networks," in *Proc. 21st ACM Int. Conf. Inf. Knowl. Manage.*, 2012, pp. 1582–1586.
[31] D. Lian *et al.*, "Content-aware collaborative filtering for location recommendation based on human mobility data," in *Proc. IEEE Int. Conf. Data Mining*, Nov. 2015, pp. 261–270.
[32] J. Xu, Y. Yao, H. Tong, X. Tao, and J. Lu, "Ice-breaking: Mitigating cold-start recommendation problem by rating comparison," in *Proc. 24th Int. Joint Conf. Artif. Intell.*, 2015, pp. 3981–3987.
[33] P. Mazumdar, B. K. Patra, and K. S. Babu, "Cold-start point-of-interest recommendation through crowdsourcing," *ACM Trans. Web*, vol. 14, no. 4, pp. 1–36, 2020.

Junyang Chen (Member, IEEE) received the Ph.D. degree in computer and information science from the University of Macau, Macau, China, in 2020.

He is currently an Assistant Professor with the College of Computer Science and Software Engineering, Shenzhen University, Shenzhen, China. His research interests include graph neural networks, text mining, and recommender systems.

Mengzhu Wang received the master's degree from Chongqing University, Chongqing, China, in 2018. She is currently pursuing the Ph.D. degree with the Science and Technology on Parallel and Distributed Laboratory, School of Computer Science, National University of Defense Technology, Changsha, China.

Her current research interests include transfer learning and computer vision.

Haodi Zhang received the Ph.D. degree from the Department of Computer Science and Engineering, Hong Kong University of Science and Technology, Hong Kong, in 2016.

He is currently an Assistant Professor with the College of Computer Science and Software Engineering, Shenzhen University, Shenzhen, China. His research interests include knowledge representation and reasoning, deep learning, crowdsourcing over probabilistic and uncertain databases, and natural language processing.

Zhenghua Xu received the M.Phil. degree in computer science from The University of Melbourne, Melbourne, VIC, Australia, in 2012, and the D.Phil. degree in computer Science from the University of Oxford, Oxford, U.K., in 2018.

From 2017 to 2018, he worked as a Research Associate with the Department of Computer Science, University of Oxford. He is currently a Professor with the Hebei University of Technology, Tianjin, China.

Prof. Xu was a recipient of the "100 Talents Plan" of Hebei Province.

Xueliang Li received the Ph.D. degree in computer science from Roskilde University, Roskilde, Denmark, in 2017.

He is currently a Research Fellow with the College of Computer Science and Software Engineering, Shenzhen University, Shenzhen, China.

Zhiguo Gong (Senior Member, IEEE) received the Ph.D. degree in computer science from the Institute of Mathematics, Chinese Academy of Science, Beijing, China, in 1998.

He is currently a Professor with the Faculty of Science and Technology, University of Macau, Macau, China. His current research interests include machine learning, data mining, database, and information retrieval.

Kaishun Wu (Senior Member, IEEE) received the Ph.D. degree in computer science and engineering from The Hong Kong University of Science and Technology (HKUST), Hong Kong, in 2011.

He is currently a Distinguished Professor with the College of Computer Science and Software Engineering, Shenzhen University, Shenzhen, China.

Victor C. M. Leung (Life Fellow, IEEE) is currently a Distinguished Professor of computer science and software engineering with Shenzhen University, Shenzhen, China. He is also an Emeritus Professor of electrical and computer engineering and the Director of the Laboratory for Wireless Networks and Mobile Systems, University of British Columbia (UBC), Vancouver, BC, Canada. His research interests include wireless networks and mobile systems.