Deep Representation Learning for Location-Based Recommendation

Zhenhua Huang, Xiaolong Lin, Hai Liu, Bo Zhang, Yunwen Chen, *Member, IEEE*, and Yong Tang

***Abstract*—Location-based recommendation has recently received a lot of attention in the communities of information service and mobile application. Its task is to provide personalized recommendations of points of interest (POIs) to users at a certain time and location. However, existing location-based recommendation models have at least two main drawbacks: first they cannot adequately capture semantic features of POIs and users, which may lead to unsatisfactory recommendations and second they cannot effectively address the cold-start problem. To address the above drawbacks, in this article, we first propose a novel deep representation learning-based model (DRLM) for improving the recommendation accuracy. In DRLM, we mainly focus on learning to accurately represent semantic features of POIs and users. Specifically, four co-occurrence matrices are constructed to produce four different original features for each POI, and a principal component analysis (PCA) algorithm is utilized to generate a semantic feature of each POI from its four original features. On the other hand, a three-modal simple recurrent unit (TMSRU) network is given to constructed semantic features of users using semantic features of POIs, times, and locations. We further propose minimum description length (MDL)-based and skyline-based strategies to address the cold-start issues for new users and new POIs, respectively. Through experiments on two real-world data sets, we show that compared with the state-of-the-art approaches, the proposed model DRLM can achieve the superior performance in terms of high recommendation accuracy and effectiveness in handling the cold-start problem.**

***Index Terms*—Location-based recommendation, neural network, point of interest (POI), representation learning, social network.**

I. INTRODUCTION

THE exponential growth of information in the Internet causes the information overload problem, making it impossible for users to quickly and accurately obtain their interested contents. Recommender systems are an effective tool to address this problem [1]. In recent years, more and more location-based social networks (LBSNs), such as Gowalla, Foursquare, and Yelp, have emerged owing to the popularity of smartphones. People on LBSN are willing to share their experiences with their friends about points of interest (POIs). At present, the providers of LBSN have collected large amounts of user check-in data. Under these circumstances, location-based recommendation becomes one of the most important services in LBSN, and has attracted much attention from researchers in both academia and industry [2]. It aims to suggest unvisited POIs to users based on history users' check-in data.

In the early days (before 2015), researchers focused on traditional location-based recommendation models. They mainly utilize temporal or spatial influences with collaborative filtering (CF) [3]–[7] and Markov transition (MT) approaches [8]–[10]. While in most real applications, the recommendation performance of these traditional models is unsatisfactory [11].

With great successes in the areas of computer vision (CV) and natural language processing (NLP), deep learning begins to attract large interest in the area of location-based recommendation and bring more opportunities to improve the recommendation performance of traditional models. And, for producing high-quality recommendation results, it is very important for deep learning-based recommendation models to accurately capture semantic features of POIs, users, times, and locations. Since a feature representation of times (FR-T) and a feature representation of locations (FR-L) are relatively simple, at present, existing works mainly focus on feature extraction of POIs and users by utilizing deep neural networks [12]. Generally, the feature extraction of POIs mainly employs multilayer perception (MLP), convolutional neural network (CNN), and graph embedding [13]–[17]. And based on semantic features of POIs, the feature extraction of users generally employs recurrent neural network (RNN), such as long short-term memory (LSTM) network and gated recurrent unit (GRU) network [18]–[21].

However, existing works have at least two main drawbacks: 1) they cannot adequately capture semantic features of POIs and users, which may lead to unsatisfactory recommendations of POIs for users and 2) they cannot effectively solve the cold-start problem. For the first drawback, the main reason is that existing works usually only consider the tag features of POIs and use simple aggregation strategies to generate each user' feature based on his latest visited POIs. For the second drawback, the main reason is that they cannot accurately produce the features of new users and new POIs.

Manuscript received September 6, 2019; revised December 27, 2019; accepted February 2, 2020. Date of publication March 4, 2020; date of current version June 10, 2020. This work was supported in part by the National Natural Science Foundation of China under Grant 61772366, Grant 61572326, Grant 61802258, and Grant U1811263 and in part by the Shanghai Committee of Science and Technology under Grant 17ZR1445900, Grant 17070502800, and Grant 18ZR1428300. *(Corresponding author: Bo Zhang.)*

Zhenhua Huang, Xiaolong Lin, Hai Liu, and Yong Tang are with the School of Computer Science, South China Normal University, Guangzhou 510631, China (e-mail: jukiehuang@163.com; 75856599@qq.com; liuhai@scnu.edu.cn ytang@m.scnu.edu.cn).

Bo Zhang is with the College of Information, Mechanical and Electrical Engineering, Shanghai Normal University, Shanghai 200234, China (e-mail: zhangbo@shnu.edu.cn).

Yunwen Chen is with DataGrand Inc., Shanghai 201203, China (e-mail: chenyunwen@datagrand.com).

Digital Object Identifier 10.1109/TCSS.2020.2974534

To address the first drawback, in this article, a novel deep representation learning-based model (DRLM) is proposed to improve the recommendation accuracy. In DRLM, we accurately capture semantic features of POIs, users, times, and locations; and by following the previous works, we mainly focus on learning to represent semantic features of POIs and users. Inspired by the idea of global vectors (GloVe) [22], we construct four co-occurrence matrices for POIs that are then used to produce four different original feature vectors for each POI. These four co-occurrence matrices are produced from POI-user observation matrix, POI-time observation matrix, POI-location observation matrix, and POI-tag observation matrix, respectively. We then employ a principal component analysis (PCA) method [23] to extract semantic feature vectors of all POIs from their four original feature vectors. Then, based on semantic feature vectors of POIs, we get a semantic feature vector for each time and each location using max pooling and 2-max pooling, respectively. Finally, based on a simple recurrent unit (SRU) [24], we introduce a three-modal SRU (TMSRU) neural network to construct semantic feature vectors of users through employing semantic feature vectors of POIs, times, and locations, and optimize its parameters by learning user preferences. The proposed TMSRU neural network can effectively model nonlinear and nontrivial quaternary relationships among user, POI, time, and location, and thereby can substantially improve the accuracy of user feature representation.

To address the second drawback, we further propose two effective strategies to deal with new users and new POIs, respectively. A minimum description length (MDL) [25]-based strategy is proposed to obtain existing users who are most relevant to a given new user, and then to produce his/her feature representation, while a skyline [26]-based strategy is presented to get the most similar existing POIs to a given new POI, and then to generate its feature representation.

Through extensive experiments on two real-world data sets, we show that compared with the state-of-the-art approaches, the proposed model DRLM can achieve the superior performance in terms of high recommendation accuracy and effectiveness in dealing with the cold-start problem.

The rest of this article is organized as follows. Section II introduces related works of our proposed model DRLM. Section III presents DRLM in detail. Section IV gives two strategies to solve the cold-start problem. Experimental results are presented in Section V. Finally, Section VI concludes this article.

II. Related Work

In this section, we introduce the related works of the proposed model DRLM, which mainly contains two categories: traditional location-based recommendation models and deep learning-based ones.

Traditional models mainly employ temporal or spatial influences with CF [3]–[7] and MT approaches [8]–[11]. Ye *et al.* [3] designed a unified framework to implement location-based collaborative recommendation, which fuses user preference to a POI with social and geographical influences. Zheng *et al.* [4] presented a cross-region CF approach (CRCFA) for recommending new POIs, which is based on hidden topics mined from user check-in records. Luan *et al.* [5] proposed a collaborative tensor factorization-based model (CTFM) to realize the task of location-based recommendation. Griesner *et al.* [6] considered utilizing matrix factorization (MF) to carry out the task of location-based recommendation, and make an attempt to integrate both geographical influence and temporal influence into MF. In particular, the authors propose GeoMF-TD, which is an extension of geographical MF with temporal dependencies. Ren *et al.* [7] presented topic model, geographical correlations, social correlations and categorical correlations (TGSC)-PMF, a context-aware probabilistic MF (PMF) model to perform location-based recommendation. Specifically, TGSC-PMF is able to adequately employ textual, geographical, social, categorical, and popularity information, and can effectively fuse these factors together.

Based on two important properties in user check-in sequences, Cheng *et al.* [8] proposed a novel model, namely factorized personalized Markov chain with local region constraints (FPMC-LR), to carry out the task of location-based recommendation through embedding of personalized Markov chains and localized regions. Zhao *et al.* [9] pointed out the drawbacks of FPMC-LR, and presented STELLAR, a spatial–temporal latent ranking model to depict the interactions among users, POIs, and times. Specifically, it utilizes a ranking-based pairwise tensor factorization framework to efficiently realize the task of location-based recommendation through capturing user–POI, POI–time, and POI–POI interactions. In addition, Debnath *et al.* [10] considered the following factors that can influence a user to choose preferred POIs: user's sequential patterns, categorical preferences, temporal activities, and location preferences, and popularity of POIs. On this basis, a unified model is proposed to recommend POIs.

While in most industrial applications, the recommendation performance of these traditional models is unsatisfactory [11].

In recent years, deep learning has attracted large interest in the area of location-based recommendation and has brought more opportunities to improve the recommendation quality of traditional models. Rahmani *et al.* [13] developed an efficient neural model, namely category-aware POI embedding (CATAPE), to generate a POI embedding by combining sequential and categorical information from POIs. Zhang *et al.* [14] proposed NEXT, a simple but effective neural network model, for next POI recommendation. Specifically, it can efficiently learn the hidden intent based on user's next move, by fusing different factors into a unified manner. Xing *et al.* [15] presented CPC, a content-aware POI recommendation model based on CNN. The proposed CPC model adequately employs the three types of content information, containing POI properties, user interests, and sentiment indications. Xie *et al.* [16] designed a graph-based embedding (GE) model, which incorporates the four factors (sequential effect, geographical influence, temporal cyclic effect, and semantic effect) in a unified framework and embeds the four bipartite graphs (POI–POI, POI–region, POI–time, and POI–word) into a shared low-dimensional space. In addition, inspired by knowledge graph embedding [27], Qian *et al.* [17] devel-

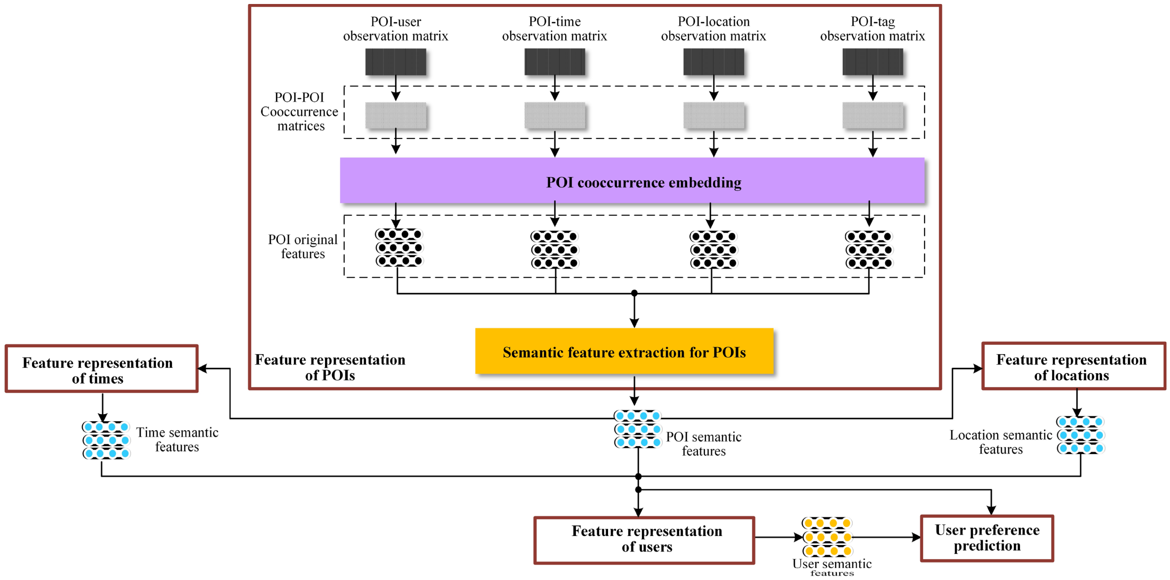


Fig. 1. Overall training framework of DRLM.

oped a spatiotemporal context-aware and translation-based model (STA) for location-based recommendation, by capturing the third-order relationship among POIs, users, and spatiotemporal contexts. In particular, STA embeds both POIs and users into a transition space, and models spatiotemporal contexts as translation vectors operating over POIs and users.

Inspired by the idea of deep and broad learning [28], Wang *et al.* [18] proposed a deep context-aware POI recommendation (DCPR) model for learning user and POI features. In particular, DCPR utilizes an RNN-based layer for user feature extraction. Based on the LSTM architecture [29], Zhao *et al.* [19] proposed a novel model (i.e., ST-LSTM) to embed user features through capturing spatiotemporal intervals between history check-ins. Specifically, ST-LSTM is able to efficiently incorporate time and distance gates, and models short-term and long-term preferences of users simultaneously. Through employing an attention mechanism [30], Huang *et al.* [20] designed an attention-based spatiotemporal LSTM (ATST-LSTM) model for location-based recommendation. ATST-LSTM can pay attention on the relevant historical check-in information in a check-in sequence. In addition, Altaf *et al.* [21] developed an efficient model spatiotemporal attention over GRUs (STA-GRU) for POI sequence modeling and user feature representation.

Yet, as discussed in Section I, existing works have at least two main drawbacks, and are likely to lead to the poor recommendation performance.

III. Methodology and Implementation for DRLM Model

In this section, we first overview DRLM, and then present implementation details of DRLM.

A. Overview of DRLM

The overall training framework of DRLM is shown in Fig. 1. It contains five components: FR-T, feature representation of POIs (FR-POI), FR-L, feature representation of users (FR-U), and user preference prediction (UPP).

1) *Feature Representation of POIs:* In the FR-POI component, inspired in [22], [31], and [32], we first construct four different POI–POI co-occurrence matrices from POI-user observation matrix, POI-time observation matrix, POI-location observation matrix, and POI-tag observation matrix, respectively. Then, for each POI–POI co-occurrence matrix, we use the idea of GloVe [22] to capture the co-occurrence relationship between any two POI, and based on POIs' co-occurrence relationships, we generate an original feature vector for each POI (POI co-occurrence embedding shown in Fig. 1). In this way, we are able to obtain four different original feature vector for each POI. On this basis, we further employ a PCA method [23] to extract semantic feature vectors of all POIs from their four original feature vectors (semantic feature extraction (SFE) for POIs shown in Fig. 1).
2) *Feature Representation of Times:* By following previous work [16], [17], [20], we split time into 24 time slots that correspond to 24 h in a day and are associated with time numbers 1, 2, ..., and 24, respectively. On this basis, for each time slot, we first get all the POIs visited at this time slot, and then aggregate these POIs to produce its semantic feature vector by utilizing a max pooling operation.
3) *Feature Representation of Locations:* In the FR-L component, for each location, we first obtain all the POIs inside it, and then aggregate these POIs to generate its semantic feature vector by using a 2-max pooling operation.
4) *Feature Representation of Users:* In the FR-U component, we take semantic feature vectors of POIs, times, and locations as input, and design a TMSRU network to effectively construct the semantic feature vector for each user. Unlike the traditional GRU [21], TMSRU is

based on SRU [24] and can effectively model the joint effects of POIs, times, and locations on users, and thus can improve the accuracy of FR-U.

5) *User Preference Prediction:* Based on semantic feature vectors of POIs, times, and locations, in the UPP component, we optimize all the parameters in the TMSRU network and refine semantic feature vectors of users through accurately learning user preferences. Specifically, we use a two-layer MLP neural network to perform the task of user preference learning.

After the model training is completed, we can obtain the final FR-POI, FR-T, FR-L, and users, and a two-layer MLP utilized to predict user preferences. Then, for a given quadruple (user, POI, time, and location), we can calculate the preference of user on *POI* at time and location by utilizing the trained two-layer MLP neural network.

B. FR-POI

We describe the FR-POI component in detail. Suppose that there are m users U: $\{u_1 \sim u_m\}$, n POIs V: $\{v_1 \sim v_n\}$, s locations L: $\{l_1 \sim l_s\}$, and g tags E: $\{e_1 \sim e_g\}$ in a location-based recommender system.

In FR-POI, we first obtain four observation matrices.

1) POI-user observation matrix $\mathbf{M}_u \in \mathbb{R}^{n\times m}$. If u_j $(1 \leq j \leq m)$ has visited v_i $(1 \leq i \leq n)$, then $\mathbf{M}_u[i, j] = 1$, otherwise $\mathbf{M}_u[i, j] = 0$.
2) POI-time observation matrix $\mathbf{M}_t \in \mathbb{R}^{n\times 24}$. If v_i has been visited at t_j $(1 \leq j \leq 24)$, then $\mathbf{M}_t[i, j] = 1$, otherwise $\mathbf{M}_t[i, j] = 0$.
3) POI-location observation matrix $\mathbf{M}_s \in \mathbb{R}^{n\times s}$. If v_i is inside l_j $(1 \leq j \leq \mathrm{s})$, then $\mathbf{M}_s[i, j] = 1$, otherwise $\mathbf{M}_s[i, j] = 0$.
4) POI-tag observation matrix $\mathbf{M}_e \in \mathbb{R}^{n\times g}$. If v_i is associated with e_j $(1 \leq j \leq \mathrm{g})$, then $\mathbf{M}_e[i, j] = 1$, otherwise $\mathbf{M}_e[i, j] = 0$.

Based on the above four observation matrices, we respectively construct four different POI–POI co-occurrence matrices $\mathbf{M}_c^u$, $\mathbf{M}_c^t$, $\mathbf{M}_c^l$, and $\mathbf{M}_c^e \in \mathbb{R}^{n\times n}$:

$$\begin{cases} \mathbf{M}_c^u[i, j] = (\mathbf{M}_u[i, *] \wedge \mathbf{M}_u[j, *]) \\ \mathbf{M}_c^t[i, j] = (\mathbf{M}_t[i, *] \wedge \mathbf{M}_t[j, *]) \\ \mathbf{M}_c^l[i, j] = (\mathbf{M}_l[i, *] \wedge \mathbf{M}_l[j, *]) \\ \mathbf{M}_c^e[i, j] = (\mathbf{M}_e[i, *] \wedge \mathbf{M}_e[j, *]) \end{cases} \quad 1 \leq i, \quad j \leq n \tag{1}$$

where $\mathbf{M}_u[i, *]$ is the ith row vector of $\mathbf{M}_u$, "$\wedge$" represents an elementwise AND operation between two vectors, and $\uplus(\cdot)$ is a function of calculating the count of elements equal to 1 in a given vector.

Then, for each of four POI–POI co-occurrence matrices, based on the idea of GloVe [22], we introduce a procedure POI co-occurrence embedding (POI-CE) to produce an original feature vector for each POI. We use $\mathbf{M}_c^u$ as an example to demonstrate POI-CE.

For each POI v_i, POI-CE aims to learn a d-dimensional original feature vector $\mathbf{v}_i^u$ for v_i. According to [22], we are able to obtain the following equation:

$$\begin{aligned} \left(\mathbf{v}_i^u\right)^T \mathbf{v}_j^u &= \log\left(\frac{\mathbf{M}_c^u[i, j]}{\uplus(\mathbf{M}_u[i, *])}\right) \\ &= \log\left(\mathbf{M}_c^u[i, j]\right) - \log\left(\uplus(\mathbf{M}_u[i, *])\right). \end{aligned} \tag{2}$$

Then, we can get the following equation based on (2):

$$\left(\mathbf{v}_i^u\right)^T \mathbf{v}_j^u = \log\left(\mathbf{M}_c^u[i, j]\right) - \Delta_{ij} \tag{3}$$

where Δ_{ij} is a global bias with respect to $\mathbf{v}_i^u$ and $\mathbf{v}_j^u$.

Based on (3), POIs' original feature vectors can be learned using the loss function

$$\mathrm{L}_{co} = \left(\left(\mathbf{v}_i^u\right)^T \mathbf{v}_j^u - \log\left(\mathbf{M}_c^u[i, j]\right) + \Delta_{ij}\right)^2. \tag{4}$$

During the training, we minimize (4) and optimize the parameters $\{\mathbf{v}_*^u, \Delta_*\}$. And, once the training of POI-CE is completed, we can get an original feature vector for each POI.

Similarly, based on $\mathbf{M}_c^t$, $\mathbf{M}_c^l$, and $\mathbf{M}_c^e$, we can use the POI-CE procedure to obtain another three d-dimensional original feature vectors of each POI v_i, respectively, i.e., $\mathbf{v}_i^t$, $\mathbf{v}_i^l$, and $\mathbf{v}_i^e$.

On this basis, another procedure SFE is proposed to extract semantic feature vectors of all POIs from their four original feature vectors. In SFE, for each POI, we first concatenate its four original feature vectors as a combination vector whose dimensionality is $4d$. Then, we utilize a PCA algorithm [23] to obtain a d_v-dimensional semantic feature vector $\mathbf{v}_i$ for each POI $v_i \in V$.

C. FR-T and FR-L

We describe the FR-T and FR-L components as follows.

In the FR-T component, there are 24 time slots in total that correspond to 24 h in a day. We denote them as t_1, t_2,..., and t_{24}, respectively. For each time slot t_i $(1 \leq i \leq 24)$, we first obtain the set of all the POIs visited at t_i: $A_i = \{v_1, v_2, \ldots, v_y\}$. Then, using max pooling, we aggregate these yPOIs' semantic feature vectors to generate a semantic feature vector $\mathbf{t}_i$ for t_i

for $j = 1 \sim d_v$

1) $x \leftarrow \mathrm{top}{-}1\{\mathbf{v}_1[j], \mathbf{v}_2[j], \ldots, \mathbf{v}_y[j]\}$;
2) $\mathbf{t}_i[j] \leftarrow x$.

Clearly, the dimensionality of $\mathbf{t}_i$ is equal to d_v.

Similarly, in the FR-L component, for each location $l_i \in L$, we first obtain the set of all the POIs inside l_i: $P_i = \{v_1, v_2, \ldots, v_z\}$. Then, by utilizing 2-max pooling, we aggregate these zPOIs' semantic feature vectors to generate a semantic feature vector $\mathbf{l}_i$ for l_i

for $j = 1 \sim d_v$

1) $x_1, x_2 \leftarrow \mathrm{top} - 2\{\mathbf{v}_1[j], \mathbf{v}_2[j], \ldots, \mathbf{v}_z[j]\}$;
2) $\mathbf{l}_i[2\ j - 1] \leftarrow x_1, \mathbf{l}_i[2\ j] \leftarrow x_2$.

Clearly, the dimensionality d_l of $\mathbf{l}_i$ is equal to $2d_v$.

D. FR-U and UPP

We first describe the FR-U component in detail. It is introduced to accurately model each user's semantic feature vector, which is based on his preference history consisting of

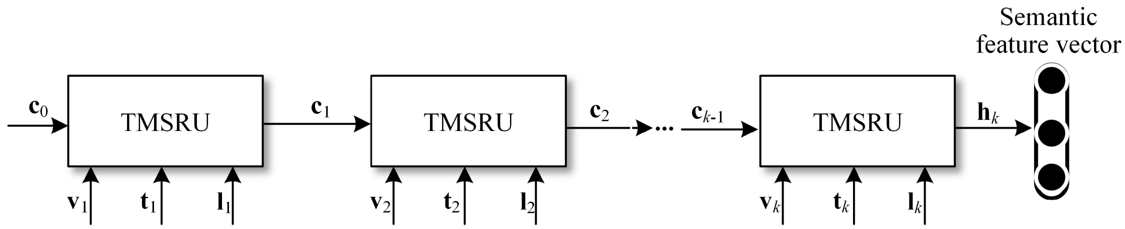


Fig. 2. Representation framework of user semantic features used in the FR-U component.

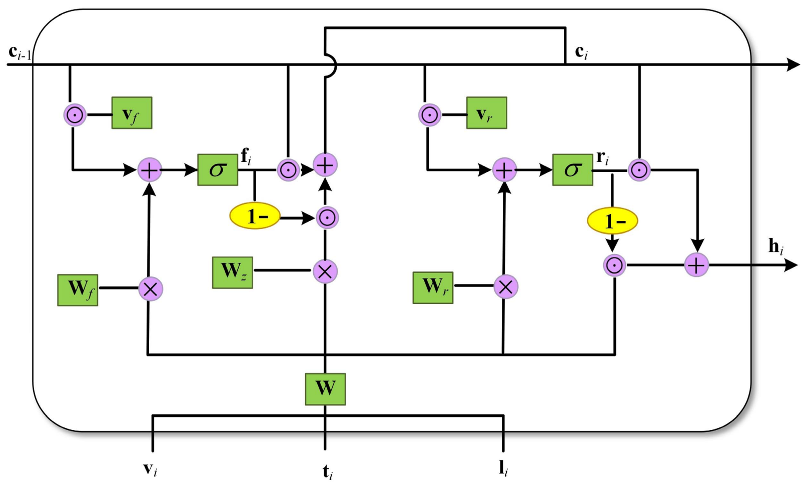


Fig. 3. Single TMSRU unit at the ith time step.

the k latest visited POIs $\{(v_1, t_1, l_1), (v_2, t_2, l_2),\ldots,(v_k, t_k, l_k)\}$. Fig. 2 shows the framework of FR-U, in which a TMSRU neural network is proposed to realize the task of modeling. Specifically, the proposed TMSRU neural network utilizes k sequential TMSRU units to construct a user semantic feature vector (i.e., $\mathbf{h}_k$).

In the following section, we propose the design details of TMSRU. Fig. 3 shows a single TMSRU unit at the ith ($1 \leq i \leq k$) time step. TMSRU is based on SRU [24] and employed to model the joint effect of POI, time, and location on a user. Its implementation process is as follows.

1) Calculate the fusion gate based on $\mathbf{v}_i$, $\mathbf{t}_i$, and $\mathbf{l}_i$

$$\mathbf{g}_i = \sigma\left(\mathbf{W}\left(\mathbf{v}_i \oplus \mathbf{t}_i \oplus \mathbf{l}_i\right)\right) \tag{5}$$

where $\oplus$ is a concatenation operation, $\sigma(\cdot)$ is a Sigmoid function, and $\mathbf{W}$ is a parameter to be learned via training.

2) Calculate the forget gate, respectively

$$\mathbf{f}_i = \sigma\left(\mathbf{W}_f\mathbf{g}_i + \mathbf{v}_f \odot \mathbf{c}_{i-1}\right) \tag{6}$$

where $\odot$ is a an Hadamard product (i.e., elementwise multiplication), $\mathbf{c}_{i-1}$ is a state vector at the (i-1)th time step, and $\mathbf{W}_f$ and $\mathbf{v}_f$ are the parameters to be learned via training.

3) Calculate the current state vector, respectively

$$\mathbf{c}_i = \mathbf{f}_i \odot \mathbf{c}_{i-1} + (1 - \mathbf{f}_i) \odot (\mathbf{W}_z\mathbf{g}_i) \tag{7}$$

where $\mathbf{W}_z$ is a parameter to be learned via training.

4) Calculate the reset gate, respectively

$$\mathbf{r}_i = \sigma\left(\mathbf{W}_r\mathbf{g}_i + \mathbf{v}_r \odot \mathbf{c}_{i-1}\right) \tag{8}$$

where $\mathbf{W}_r$ and $\mathbf{v}_r$ are the parameters to be learned via training.

5) Calculate the current activation $\mathbf{h}_t$

$$\mathbf{h}_i = \mathbf{r}_i \odot \mathbf{c}_i + (1 - \mathbf{r}_i)\,\mathbf{g}_i. \tag{9}$$

In addition, we adopt the following initialization strategy to produce $\mathbf{h}_0$:

$$\mathbf{c}_0 = \psi_c\left(\mathbf{v}_1 \oplus \mathbf{t}_1 \oplus \mathbf{l}_1\right) \tag{10}$$

where $\cdot\psi_c$ is a three-layer MLP neural network, containing three fully-connected layers.

We next propose the UPP component, which aims to optimize all the parameters in the FR-U component and refine user semantic feature vectors by learning user preferences. In UPP, we employ a two-layer MLP neural network to perform the task of user preference learning. For an input sample $T = (\mathbf{u}, \mathbf{v}, \mathbf{t}, \mathbf{l})$, the output of FC_1 can be expressed as

$$\mathbf{y}_1 = \sigma\left(\mathbf{Y}_1(\mathbf{t} \oplus \mathbf{l}) + \mathbf{b}_1^p\right) \tag{11}$$

where $\mathbf{Y}_1$ and $\mathbf{b}_1^p$ are the parameters to be learned via training. Then, the output of FC_3 can be expressed as

$$\sigma y_2 = \left(\mathbf{Y}_2\mathbf{y}_2 + b_2^p\right) \tag{12}$$

where $\mathbf{Y}_2$ and b_2^p are the parameters to be learned via training.

On this basis, we can learn UPP by minimizing the following loss function:

$$\mathcal{L}_{\text{upp}} = -\frac{1}{|B|}\sum_{T\in B}\left(\underbrace{\log y_2}_{\text{positive sample}} + \underbrace{\log(1-y_2)}_{\text{negative sample}}\right). \quad (13)$$

During the training, we replace $\mathbf{u}$ with $\mathbf{h}_k$ (in FR-U) and optimize all the parameters contained in FR-U and UPP. And, once the training process is completed, we are able to obtain: 1) a trained TMSRU network and 2) a trained two-layer MLP network. Based on the former, we can obtain a d_u-dimensional semantic feature vector for each user. Then, for a given quadruple (u, v, t, l), using the latter, we can easily get the preference of user u on POI v at time t and location l.

IV. Cold-Start Recommendation

In this section, we present two effective strategies to handle new users and new POIs, respectively.

A. Cold-Start Recommendation for New Users

For a new user u, our proposed model DRLM are not able to directly obtain u's semantic feature vector since u has no preference history. It will lead DRLM to fail to carry out the recommendation task. To address this problem, an MDL [25]-based strategy is proposed. And its implementation process is described as follows.

We first obtain the friend set of u in a given LBSN: $\mathcal{F}_u = \{u_1, u_2, \ldots, u_\xi$, and sort $\mathcal{F}_u$ according to the following relevance measure:

$$\mathcal{R}(u, u_i) = \gamma_i \frac{|\mathcal{F}_u \cap \mathcal{F}_{u_i}| + 1}{|\mathcal{F}_u| + |\mathcal{F}_{u_i}|} 1 \le i \le \xi \quad (14)$$

where $\gamma_i = \text{NPOIs}(u_i)/\max_{1\le i\le\xi}(\{\text{NPOIs}(u_i)\})$ is the activity of u_i. Here, $\text{NPOIs}(u_i)$ is the number of POIs visited by u_i in the latest period of time. Then, we are able to obtain a sorting list $\mathcal{T}_u =< u'_1, u'_2, \ldots, u'_\xi >$, satisfying for each $2 \le z \le \xi$, $\mathcal{R}(u, u'_{z-1}) \ge \mathcal{R}(u, u'_z)$.

We next prune some users in $\mathcal{T}_u$ with low relevance measure values. They are likely to become noise in cold-start recommendation. A straightforward approach is to prune the users whose relevance measure values are below a given threshold. However, this threshold is difficult to be specified. Therefore, we propose an automatic strategy based on the MDL principle, to decide which users in $\mathcal{T}_u$ can be pruned. First of all, we transform each user's relevance measure value to an integer

$$\mathcal{R}(u, u'_i) \leftarrow 10^\epsilon \cdot \mathcal{R}(u, u'_\xi)$$

where ϵ is equal to the decimal point number of u'_ξ. We then split $\mathcal{T}_u$ into two separate sublists: $\mathcal{T}_u^1 = \langle u'_1, u'_2, \ldots, u'_z\rangle$ and $\mathcal{T}_u^2 = \langle u'_{z+1}, u'_{z+2}, \ldots, u'_\xi\rangle$. Our goal is to achieve an optimal pruning point z:

We first calculate the mean of relevance measure values for $\mathcal{T}_u^1$: $a_u^1 = \lceil\sum_{i=1}^{z}\mathcal{R}(u, u'_i)/z\rceil$. Then, for each user u'_i in $\mathcal{T}_u^1$, we calculate the difference from the mean a_u^1, i.e., $d_i^1 = |a_u^1 - \mathcal{R}(u, u'_i)|$. Similarly, $a_u^2 = \lceil\sum_{x=z+1}^{\xi}\mathcal{R}(u, u'_x)/(\xi - z)\rceil$ is calculated for $\mathcal{T}_u^2$, and $d_x^2 = |a_u^2 - \mathcal{R}(u, u'_x)|$ is then calculated for each user u'_x in $\mathcal{T}_u^2$. For relevance measure values of users in $\mathcal{T}_u$, the code length $\text{CL}(z)$ can be expressed as the sum of the bit lengths of the numbers we need to store, that is

$$\text{CL}(z) = \log_2^{a_u^1} + \sum_{i=1}^{z}\log_2^{d_i^1} + \log_2^{a_u^2} + \sum_{i=z+1}^{\xi}\log_2^{d_i^2}. \quad (15)$$

Then, based on the MDL principle, the pruning point z is optimal if $\text{CL}(z)$ is minimum. Ultimately, we use semantic feature vectors of users in $\mathcal{T}_u^1$ to produce u's semantic feature vector

$$\mathbf{u}[j] = \sum_{i=1}^{z}\lambda_i \mathbf{u}'_i[j], \quad 1 \le j \le d_u \quad (16)$$

where

$$\lambda_i = \frac{\mathcal{R}(u, u'_i)}{\sum_{x=1}^{z}\mathcal{R}(u, u'_x)}$$

is an importance weight of $\mathbf{u}'_i$, and d_u is the dimensionality of user semantic feature vectors.

B. Cold-Start Recommendation for New POIs

Clearly, for a new POI v, it has no corresponding semantic feature vector, and hence cannot be recommended for users by utilizing DRLM. To address this problem, a skyline [26]-based strategy is introduced, and the implementation process is described below.

Let $l(v)$ be the location containing v. We first obtain the ω nearest locations to l, and then obtain all the POIs inside them, denoted as $V_c = \{v_1, v_2, \ldots, v_\phi\}$. ω is a predefined threshold to control the number of candidates' POIs. We next calculate two measure values for each $v_i \in V_c$.

1) Distance measure $d(v_i)$ is equal to the geographic distance between $l(v_i)$ and $l(v)$, i.e., $d(v_i) = gd(l(v_i), l(v))$.
2) Semantic measure $s(v_i)$ is equal to the Jaccard coefficient [34] between the tag sets $E(v_i)$ and $E(v)$ for v_i and v, i.e.,

$$s(v_i) = \frac{|E(v_i) \bigcap E(v)|}{|E(v_i) \cup E(v)|}.$$

Then, we normalize these two measure values for each $v_i \in V_c$, respectively

$$d(v_i) = d(v_i) / \sum_{a=1}^{\phi} d(v_a) \text{ and}$$

$$s(v_i) = s(v_i) / \sum_{a=1}^{\phi} s(v_a).$$

On this basis, we can get the skyline set of V_c: SKY $= \{v\text{'} \in V_c | \neg\exists q \in V_c, q \succ v\text{'}\}$, where "$\succ$" is a dominance relationship [34]. Specifically, q is said to dominate v' if they satisfy: 1) $d(q) \le d(v\text{'})$ and $s(q) \ge s(v\text{'})$; and 2) $d(q) < d(v\text{'})$ or $s(q) > s(v\text{'})$. Without loss of generality, we let SKY $= \{v'_1, v'_2, \ldots, v'_\varpi\} \subseteq V_c$.

Ultimately, we use semantic feature vectors of POIs in SKY to produce v's semantic feature vector

$$\mathbf{v}[j] = \sum_{i=1}^{\varpi}\beta_i \mathbf{v}'_i[j], \quad 1 \le j \le d_v \quad (17)$$

TABLE I
STATISTICS OF TWO DATA SETS USED IN EXPERIMENTS

	Foursquare	Gowalla
Users	95,163	98,951
POIs	51,801	1,272,718
Times	24	24
Locations	4,105	2000
Check-ins	1,093,410	6,044,917
Check-ins per user	11.49	61.09
POIs per location	12.62	636.36

where d_v is the dimensionality of semantic feature vectors of POIs,

$$\beta_i = \frac{p \cdot d\left(v_i'\right) + (1-p)\, s\left(v_i'\right)}{\sum_{x=1}^{\varpi}\left(p \cdot d\left(v_i'\right) + (1-p)\, s\left(v_i'\right)\right)}$$

is an importance weight of $\mathbf{v}_i'$, and $p \in (0, 1)$ is a hyperparameter to control the importance of different measures.

V. EXPERIMENTS

In this section, we conduct an empirical study of our proposed model DRLM with two real-world LBSN data sets.

A. Experimental Setup

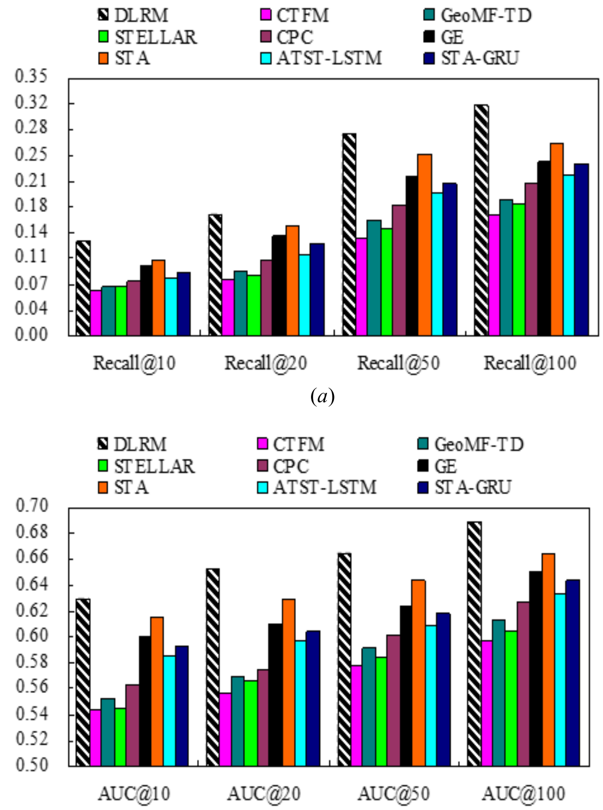
Two real-world LBSN data sets[1] are employed in the experiments: Foursquare and Gowalla.

Foursquare is one of the most widely employed online LBSN data sets, which is collected through Twitter. Each check-in is stored as user-ID, POI-ID, POI-location, including latitude and longitude, check-in timestamp, and POI-content. Like Foursquare, Gowalla is a popular online LBSN data set. The number of check-ins in Gowalla is much more than that in Foursquare. In addition, Gowalla does not include contents of POIs. Therefore, each check-in in Gowalla has the same format with Foursquare except for POI-content. Table I shows the statistics of two used data sets obtained from Foursquare and Gowalla.

In our experiments, the data sets are randomly divided into training (70%), validation (20%), and test (10%) sets. We carry out the experimental evaluation on TensorFlow platform [35] and adopt the ADAM optimizer [35], [36].

In FR-POI, d (dimensionality of POIs' original feature vectors) and d_v (dimensionality of POIs' semantic feature vectors) are set to 50 and 80, respectively. In FR-T, d_t (dimensionality of times' semantic feature vectors) is equal to $d_v = 80$. In FR-L, d_l (dimensionality of locations' semantic feature vectors) is equal to $2d_v = 160$. In FR-U, d_u (dimensionality of users' semantic feature vectors) and k (size of user preference history) are set to 10 and 100, respectively. The learning network of UPP is a two-layer MLP with the neuron counts of 120 and 1. The minimum batch size $|B| = 64$, and the initial learning rate $\eta = 0.001$. Moreover, for cold-start recommendation, ω (predefined threshold) and p (importance weight) are set to 10 and 0.3, respectively, in the skyline-based strategy.

[1]https:/sites.google.com/site/dbhongzhi



(b)

Fig. 4. Comparison with baselines on the data set Foursquare. (a) The Recall@N evaluation metric. (b) The AUC@N evaluation metric.

As for evaluation metrics, we employ two well-known metrics Recall@N [36] and AUC@N [37], which are widely applied for top-N recommendation evaluation. AUC represents the area under receiver operating characteristic (ROC) curve [38], [39]. Recall represents the percentage of correctly predicted true positive items in the samples

$$\text{Recall} = \text{TP}/\left(\text{TP} + \text{FN}\right).$$

Here, TP is the number of positive items that are correctly predicted to be true, and FN is the number of positive items that are falsely predicted to be false.

To verify effectiveness of DRLM, we compare it with eight state-of-the-art models: CTFM [5], GeoMF-TD [6], STELLAR [9], CPC [15], GE [16], STA [17], ATST-LSTM [20], and STA-GRU [21]. The first three [5], [6], [9] are traditional location-based recommendation models, and the last five [15]–[17], [20], [21] are deep learning-based ones. The above compared models are detailedly introduced in Section II; therefore, we do not describe them here for simplicity.

B. Performance Comparison With Baselines

We compare the performance of DRLM with its peers. Figs. 4 and 5 show the Recall@N and AUC@N values for the two data sets with $N = \{10, 20, 50, 100\}$.

From Figs. 4 and 5, we can see that DRLM achieves the superior recommendation accuracy. For example, in Fig. 4(a),

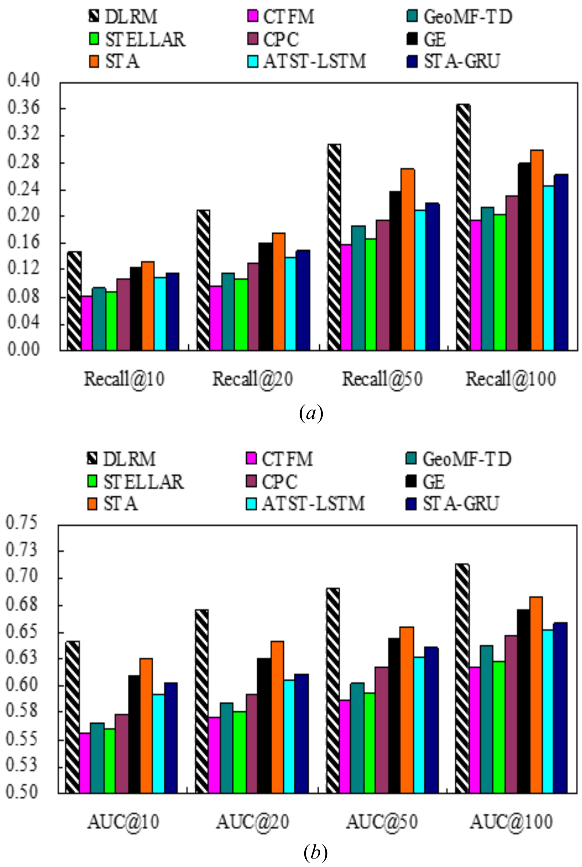


Fig. 5. Comparison with baselines on the data set Gowalla. (a) The Recall@N evaluation metric. (b) The AUC@N evaluation metric.

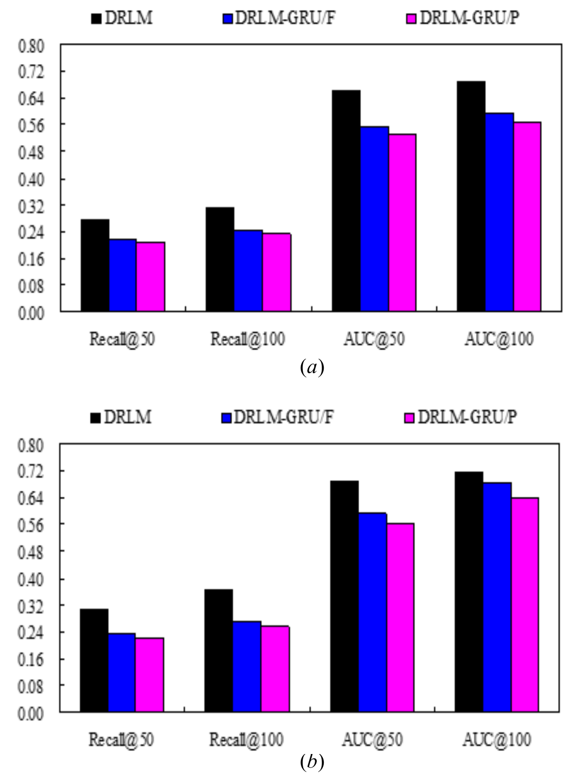


Fig. 6. Experimental evaluation for user semantic features. (a) The dataset foursquare. (b) The dataset Gowalla.

it outperforms CTFM, GeoMF-TD, STELLAR, CPC, GE, and STA, for Recall@100 on the data set Foursquare. The main reasons are twofolds as follows:

1) DRLM is able to adequately capture semantic features of POIs by introducing four novel POI original features generated from four different co-occurrence matrices, respectively.
2) More important, DRLM is able to effectively model the joint effects of POIs, times, and locations on users using the TMSRU neural network, which can substantially improve the accuracy of user feature representation.

From these two figures, we can further see that among the existing location-based recommendation models, traditional models underperform deep learning-based ones in all cases. It is mainly because compared with traditional models, deep learning-based ones can capture semantic features of users, POIs, times, and locations more accurately. For example, in Fig. 4(b), traditional models are 7.35% worse than deep learning-based ones on average for AUC@100 on Foursquare.

Moreover, we can clearly observe that all nine models have higher performance on Gowalla than on Foursquare. A possible reason is that the average number of check-ins of users in Gowalla is greater than that in Foursquare, which enables these models to learn semantic features of users more accurately. For example, the average values of these models are 0.2225 and 0.2546 on Gowalla and Foursquare, respectively, in terms of Recall@100 [see Figs. 4(a) and 5(a)].

C. Effectiveness of User/POI Semantic Features

The overall performance comparison demonstrates that the proposed model DRLM has high recommendation effectiveness. To further understand importance of user/POI semantic features, we carry out two groups of experiments in this section.

The first group of experiments is to investigate the effectiveness of user semantic features. We compare DRLM with its two variants.

1) *DRLM-GRU/F*: In DRLM, we replace TMSRU with the classic GRU and take the concatenation of semantic features of POI, time, and location as input to construct a user semantic feature.
2) *DRLM-GRU/P*: In DRLM, we use three separate GRU networks instead of a TMSRU network, and employ semantic features of POI, time, and location as inputs to construct a user semantic feature, respectively. Then, three constructed user semantic features are concatenated to form a final one.

We use Recall@N and AUC@N as experimental evaluation metrics for two data sets with $N = \{50, 100\}$. The experimental results are shown in Fig. 6.

From Fig. 6, we are able to observe that DRLM outperforms DRLM-GRU/F and DRLM-GRU/P in all cases. For example, in Fig. 6(a), DRLM outperforms DRLM-GRU/F and DRLM-GRU/P by 26.57% and 34.13%, respectively, for

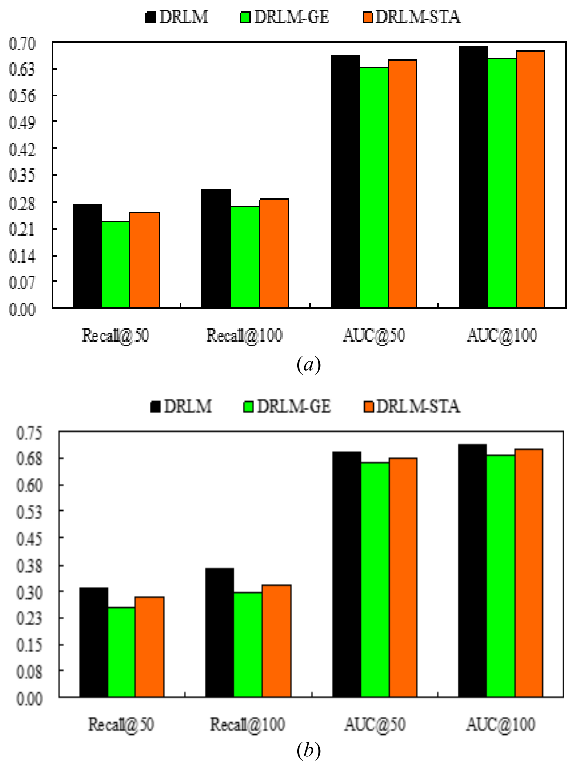


Fig. 7. Experimental study for POI semantic features. (a) The dataset foursquare. (b) The dataset Gowalla.

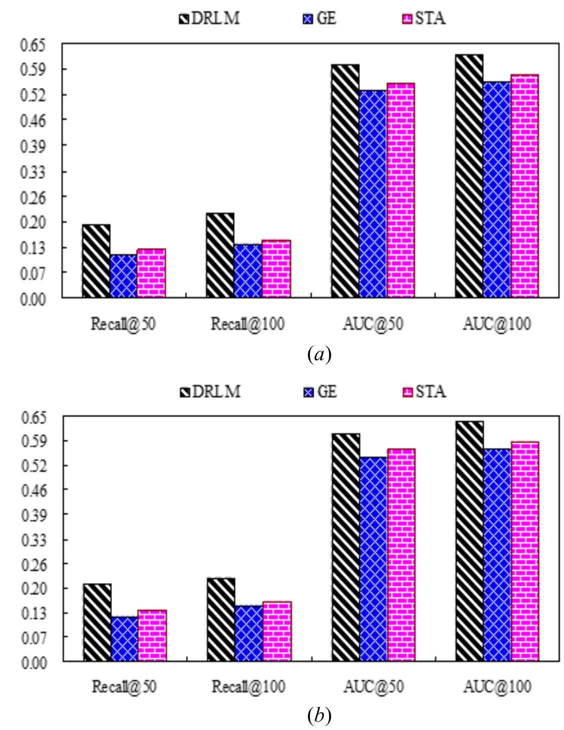


Fig. 8. Experimental evaluation for cold-start users. (a) The dataset foursquare. (b) The dataset Gowalla.

Recall@50 on Foursquare. It clearly shows that recommendation is not effective if we only employ the classic GRU. And we can effectively improve the final recommendation effectiveness by using the TMSRU network to deeply fuse POI, time, and location semantic features. In addition, we find that DRLM-GRU/F slightly outperforms DRLM-GRU/P in all cases. For example, in Fig. 6(b), DRLM-GRU/F outperforms DRLM-GRU/P by 6.73% for AUC@100 on the data set Gowalla. This is mainly because compared with DRLM-GRU/P, DRLM-GRU/F can slightly fuse POI, time, and location semantic features, thus improving the accuracy of user feature representation lightly.

In the second group of experiments, we study the effectiveness of POI semantic features. We compare DRLM with its two variants.

1) *DRLM-GE:* in DRLM, we adopt POI semantic features used in the GE model.
2) *DRLM-STA:* in DRLM, we adopt POI semantic features used in the STA model.

Note that we choose GE and STA, since they have achieved the best performance among the existing models to the best of our knowledge. Similarly, we employ Recall@N and AUC@N as experimental evaluation metrics for two data sets with $N = \{50, 100\}$. The experimental results are shown in Fig. 7.

From Fig. 7, we can see that compared with two variants, DRLM get the superior recommendation accuracy. For example, in Fig.7(a), DRLM outperforms its two variants by 4.35% and 1.76%, respectively, for AUC@100 on Foursquare. It indicates that compared with the models GE and STA, DRLM can learn POI semantic features more accurately. On the other hand, we can see that DRLM-STA outperforms DRLM-GE in all cases. It indirectly reflects that STA performs better than GE, which is consistent with the experimental results shown in Fig 5.

D. Test for Cold-Start Problem

Among existing baselines, a very few models can deal with the cold-start problem. To the best of our knowledge, GE and STA are the two best models developed to handle the cold-start problem. Thus, in this section, we compare DRLM with these two models, and carry out two groups of experiments.

In the first group of experiments, we evaluate the effectiveness of three models for cold-start users. By following [16], [17], users with less than 10 check-in records are chosen as cold-start users. We remove their check-in records from training set and take them as test set. Like in Section V-C, We use Recall@N and AUC@N as evaluation metrics for two data sets with $N = \{50, 100\}$. The experimental results are shown in Fig. 8.

From Fig. 8, we can clearly see that compared with GE and STA, DRLM is able to provide more accurate recommendation for cold-start users. For example, in Fig. 8(b), DRLM outperforms GE and STA by 49.83% and 39.12%, respectively, for Recall@100 on Gowalla. The main reason is that in DRLM, using MDL-based strategy, a cold-start user's semantic feature can be effectively represented by semantic features of his most relevant friends. On the other hand, by comparing the recommendation results shown in Figs. 4, 5, and 8, we are able to find that the accuracy of all three models decreases for cold-start users. Yet, the accuracy of DRLM decreases less than that of GE and STA. This indicates that DRLM has a better stability than GE and STA in handling cold-start users.

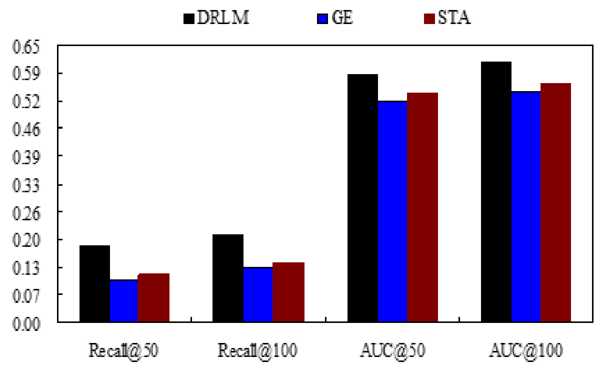


Fig. 9. Experimental evaluation for cold-start POIs on Foursquare.

In the second group of experiments, we further investigate the effectiveness of three models for cold-start POIs. By following [16], [17], POIs with less than 10 check-in records are selected as cold-start POIs. And on this basis, all check-in records related to cold-start POIs are employed as test set, and the remaining ones are used as training set. Meanwhile, users visiting at least one cold-start POI are selected as test users. Like [16] and [17], we only carry out the experimental evaluation on Foursquare, because Gowalla does not involve content information of POIs. We employ Recall@N and AUC@N as experimental evaluation metrics with $N = \{50, 100\}$. The experimental results are shown in Fig. 9.

Similarly, from Fig. 9, we can see that compared with GE and STA, DRLM can provide more accurate recommendation for cold-start POIs. For example, DRLM outperforms GE and STA by 77.25% and 59.14%, respectively, for Recall@50. The main reason is that in DRLM, through employing skyline-based strategy, a cold-start POI's semantic feature can be effectively represented by semantic features of its most similar POIs. In addition, by comparing the recommendation results shown in Figs. 4, 5, and 9, we can observe that the accuracy of all three models decreases for cold-start POIs. However, the accuracy of DRLM decreases less than that of GE and STA. This shows that DRLM has a better stability than GE and STA in handling cold-start POIs.

VI. Conclusion

This article proposes DRLM as a novel location-based recommendation model through deep representation learning. It learns the semantic features of POIs and users more accurately than all the existing models to the best of our knowledge. In particular, the proposed model can adequately capture POIs' semantic features by introducing four novel POI original features produced from four different co-occurrence matrices, respectively, and meanwhile, it can substantially improve the accuracy of users' semantic features by modeling the joint effects of POIs, times, and locations on users through the TMSRU neural network. In addition, this article designs an MDL-based strategy and a skyline-based strategy to address the cold-start issues for new users and new POIs, respectively.

In the future, we plan to utilize more kinds of POI auxiliary information to further improve the recommendation effectiveness. We also plan to propose more effective neural networks to further improve the accuracy of feature extraction.

References

[1] X. Yang *et al.*, "Collaborative filtering-based recommendation of online social voting," *IEEE Trans. Comput. Social Syst.*, vol. 4, no. 1, pp. 1–13, Mar. 2017.

[2] G. Liao, S. Jiang, Z. Zhou, C. Wan, and X. Liu, "POI recommendation of location-based social networks using tensor factorization," in *Proc. 19th IEEE Int. Conf. Mobile Data Manage. (MDM)*, Aalborg, Denmark, Jun. 2018, pp. 116–124.

[3] 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)*, Beijing, China, 2011, pp. 325–334.

[4] N. Zheng, X. Jin, and L. Li, "Cross-region collaborative filtering for new point-of-interest recommendation," in *Proc. 22nd Int. Conf. World Wide Web-WWW Companion*, Rio de Janeiro, Brazil, 2013, pp. 45–46.

[5] W. Luan, G. Liu, and C. Jiang, "Collaborative tensor factorization and its application in POI recommendation," in *Proc. IEEE 13th Int. Conf. Netw., Sens., Control (ICNSC)*, Mexico City, Mexico, Apr. 2016, pp. 1–6.

[6] J. B. Griesner, T. Abdessalem, and H. Naacke, "POI recommendation: Towards fused matrix factorization with geographical and temporal influences," in *Proc. RecSys*, Vienna, Austria, 2015, pp. 301–304.

[7] X. Ren, M. Song, H. E, and J. Song, "Context-aware probabilistic matrix factorization modeling for point-of-interest recommendation," *Neurocomputing*, vol. 241, pp. 38–55, Jun. 2017.

[8] C. Cheng, H. Yang, M. R. Lyu, and I. King, "Where you like to go next: Successive point-of-interest recommendation," in *Proc. IJCAI*, Beijing, China, 2013, pp. 2605–2611.

[9] S. Zhao, T. Zhao, H. Yang, M. R. Lyu, and I. King, "STELLAR: Spatial-temporal latent ranking for successive point-of-interest recommendation," in *Proc. AAAI*, Phoenix, AZ, USA, 2016, pp. 315–321.

[10] M. Debnath, P. K. Tripathi, and R. Elmasri, "Preference-aware successive POI recommendation with spatial and temporal influence," in *Proc. ICSI*, Bellevue, WA, USA, 2016, pp. 347–360.

[11] H. Wang, H. Shen, W. Ouyang, and X. Cheng, "Exploiting POI-specific geographical influence for point-of-interest recommendation," in *Proc. 27th Int. Joint Conf. Artif. Intell.*, Stockholm, Sweden, Jul. 2018, pp. 3877–3883.

[12] B. Chang, Y. Park, D. Park, S. Kim, and J. Kang, "Content-aware hierarchical Point-of-Interest embedding model for successive POI recommendation," in *Proc. 27th Int. Joint Conf. Artif. Intell.*, Stockholm, Sweden, Jul. 2018, pp. 3301–3307.

[13] H. A. Rahmani, M. Aliannejadi, R. Mirzaei Zadeh, M. Baratchi, M. Afsharchi, and F. Crestani, "Category-aware location embedding for point-of-interest recommendation," 2019, *arXiv:1907.13376*. [Online]. Available: http://arxiv.org/abs/1907.13376

[14] Z. Zhang, C. Li, Z. Wu, A. Sun, D. Ye, and X. Luo, "NEXT: A neural network framework for next POI recommendation," 2017, *arXiv:1704.04576*. [Online]. Available: http://arxiv.org/abs/1704.04576

[15] S. Xing, F. Liu, Q. Wang, X. Zhao, and T. Li, "Content-aware point-of-interest recommendation based on convolutional neural network," *Int. J. Speech Technol.*, vol. 49, no. 3, pp. 858–871, Oct. 2018.

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

[17] T. Qian, B. Liu, Q. V. H. Nguyen, and H. Yin, "Spatiotemporal representation learning for translation-based POI recommendation," *ACM Trans. Inf. Syst.*, vol. 37, no. 2, pp. 1–24, Jan. 2019.

[18] F. Wang, Y. Qu, L. Zheng, C.-T. Lu, and P. S. Yu, "Deep and broad learning on content-aware POI recommendation," in *Proc. IEEE 3rd Int. Conf. Collaboration Internet Comput. (CIC)*, San Jose, CA, USA, Oct. 2017, pp. 369–378.

[19] P. Zhao, H. Zhu, Y. Liu, Z. Li, J. Xu, and V. S. Sheng, "Where to go next: A spatio-temporal LSTM model for next POI recommendation," 2018, *arXiv:1806.06671*. [Online]. Available: http://arxiv.org/abs/1806.06671

[20] L. Huang, Y. Ma, S. Wang, and Y. Liu, "An attention-based spatiotemporal LSTM network for next POI recommendation," *IEEE Trans. Services Comput.*, to be published, doi: 10.1109/TSC.2019.2918310.

[21] B. Altaf, L. Yu, and X. Zhang, "Spatio-temporal attention based recurrent neural network for next location prediction," in *Proc. IEEE Int. Conf. Big Data (Big Data)*, Seattle, WA, USA, Dec. 2018, pp. 937–942.

[22] J. Pennington, R. Socher, and C. Manning, "Glove: Global vectors for word representation," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, Doha, Qatar, 2014, pp. 1532–1543.

[23] Z. Huang, G. Shan, J. Cheng, and J. Sun, "TRec: An efficient recommendation system for hunting passengers with deep neural networks," *Neural Comput. Appl.*, vol. 31, no. S1, pp. 209–222, Sep. 2018.

[24] B. Altaf, L. Yu, and X. Zhang, "Spatio-temporal attention based recurrent neural network for next location prediction," in *Proc. IEEE Int. Conf. Big Data (Big Data)*, Brussels, Belgium, Dec. 2018, pp. 4470–4481.

[25] Z. Huang, Y. Xiang, B. Zhang, and X. Liu, "A clustering based approach for skyline diversity," *Expert Syst. Appl.*, vol. 38, no. 7, pp. 7984–7993, Jul. 2011.

[26] H. Zhu, X. Li, Q. Liu, and Z. Xu, "Top-k dominating queries on skyline groups," *IEEE Trans. Knowl. Data Eng.*, to be published, doi: 10.1109/TKDE.2019.2904065.

[27] Y. Zhang, Q. Yao, Y. Shao, and L. Chen, "NSCaching: Simple and efficient negative sampling for knowledge graph embedding," in *Proc. IEEE 35th Int. Conf. Data Eng. (ICDE)*, Macao, China, Apr. 2019, pp. 614–625.

[28] S. Wang, J. Cai, Q. Lin, and W. Guo, "An overview of unsupervised deep feature representation for text categorization," *IEEE Trans. Comput. Social Syst.*, vol. 6, no. 3, pp. 504–517, Jun. 2019.

[29] Z. Huang, G. Shan, J. Cheng, and J. Ni, "A two-level attentive pooling based hybrid network for question answer matching task," in *Proc. DEXA*, Regensburg, Germany, 2018, pp. 361–368.

[30] Y. Tay, L. Anh Tuan, and S. C. Hui, "Latent relational metric learning via memory-based attention for collaborative ranking," in *Proc. World Wide Web Conf. World Wide Web (WWW)*, Lyon, France, 2018, pp. 729–739.

[31] M. Fu, H. Qu, Z. Yi, L. Lu, and Y. Liu, "A novel deep learning-based collaborative filtering model for recommendation system," *IEEE Trans. Cybern.*, vol. 49, no. 3, pp. 1084–1096, Mar. 2019, doi: 10.1109/TCYB.2018.2795041.

[32] D. S. Huang, Z. Huang, C. A. Yuan, and W. Bao, "Pupylation sites prediction with ensemble classification model," *Int. J. Data Mining Bioinf.*, vol. 18, no. 2, pp. 91–104, Jun. 2017.

[33] S. Yu, Y. Wang, M. Yang, B. Li, Q. Qu, and J. Shen, "NAIRS: A neural attentive interpretable recommendation system," in *Proc. 12th ACM Int. Conf. Web Search Data Mining (WSDM)*, Melbourne, Aust, 2019, pp. 790–793.

[34] G. Alain and Y. Bengio, "What regularized auto-encoders learn from the data-generating distribution," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 3563–3593, Nov. 2014.

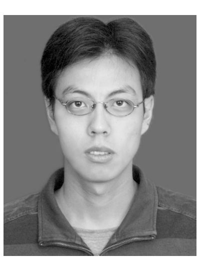
[35] Z. Huang, J. Tang, G. Shan, J. Ni, Y. Chen, and C. Wang, "An efficient passenger-hunting recommendation framework with multitask deep learning," *IEEE Internet Things J.*, vol. 6, no. 5, pp. 7713–7721, Oct. 2019.

[36] S. Bock, J. Goppold, and M. Weiß, "An improvement of the convergence proof of the ADAM-optimizer," 2018, *arXiv:1804.10587*. [Online]. Available: http://arxiv.org/abs/1804.10587

[37] Z. Huang, X. Xu, H. Zhu, and M. Zhou, "An efficient group recommendation model with multiattention-based neural networks," *IEEE Trans. Neural Netw. Learn. Syst.*, to be published, doi: 10.1109/TNNLS.2019.2955567.

[38] Z. Huang, X. Xu, J. Ni, H. Zhu, and C. Wang, "Multimodal representation learning for recommendation in Internet of Things," *IEEE Internet Things J.*, vol. 6, no. 6, pp. 10675–10685, Dec. 2019.

[39] J. Cheng, X. Wu, M. Zhou, S. Gao, Z. Huang, and C. Liu, "A novel method for detecting new overlapping community in complex evolving networks," *IEEE Trans. Syst., Man, Cybern. Syst.*, vol. 49, no. 9, pp. 1832–1844, Sep. 2019.

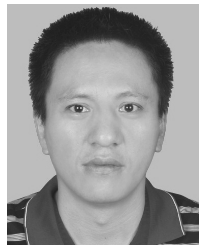


Zhenhua Huang received the Ph.D. degree in computer science from Fudan University, Shanghai, China, in 2008.

He was an Associate Professor and a Professor with the School of Electronics and Information Engineering, Tongji University, Shanghai, China, from 2012 to 2016 and from 2016 to 2018, respectively. He is currently a Professor with the School of Computer Science, South China Normal University, Guangzhou, China. His research interests mainly include deep learning, recommendation systems, data mining, knowledge discovery and big data. Since 2004, he has been publishing three books and more than 100 articles in various journals and conference proceedings.

Xiaolong Lin received the B.S. degree in the computer science from South China Normal University, Guangzhou, China, where he is currently pursuing the master's degree in computer science.

His main research interests include recommendation systems, deep learning, big data analytics, and data mining.



Hai Liu received the Ph.D. degree in computer science from Sun Yat-sen University, Guangdong, China, in 2010.

He is currently an Associate Professor with the School of Computer, South China Normal University, Guangzhou, China. His research interests include database, recommendation system, deep learning, and text mining.

Bo Zhang received the Ph.D. degree in computer science from the College of Electronics and Information Engineering, Tongji University, Shanghai, China, in 2009.

He finished his Post-Doctoral Research with Tongji University in 2012. He is currently a Professor with the College of Information, Mechanical and Electrical Engineering, Shanghai Normal University, Shanghai, China. He is also the Director of the Trust Computation Project, which is funded by the National Natural Science Foundation of China. His current research interests include database, trust computation, and social network analysis.

Yunwen Chen (Member, IEEE) received the Ph.D. degree from Fudan University, Shanghai, China, in 2008.

He is the Founder and the CEO of DataGrand Inc., Shanghai, China, a leading AI company. He is also a member of ACM and a Senior Member of CCF. He had been a Chief Data Officer (CDO) of Shanda Inc., Shanghai, the Senior Director of Tencent Inc., Guangdong, China, and a Researcher of Baidu Inc., Beijing, China. He has 32 patents and many academic publications. His research interests include data mining, natural language processing, deep learning, knowledge graph, and recommendation systems.

Yong Tang received the B.S. degree in computer science from Wuhan University, Wuhan, China, in 1985, and the Ph.D. degree from the University of Science and Technology of China, Hefei, China, in 2001.

He is currently a Professor and the Dean of the School of Computer Science, South China Normal University, Guangzhou, China, and the Director of the Services Computing Engineering Research Center of Guangdong Province. His research interests include database and cooperative software, temporal information processing, social networks, and big data analytics. He has completed more than 30 research and development projects. He has authored or coauthored more than 100 publications in these areas.