IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received September 1, 2019, accepted October 11, 2019, date of publication October 29, 2019, date of current version November 7, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2950061

SSSER: Spatiotemporal Sequential and Social Embedding Rank for Successive Point-of-Interest Recommendation

YANGYANG XU[1,2], **XUEFEI LI**[1], **JING LI**[1], **CHUNZHI WANG**[3], **RONG GAO**[3], **AND YONGHONG YU**[4]

[1]School of Computer Science, Wuhan University, Wuhan 430072, China
[2]2012 Laboratory, Huawei Technologies Company, Ltd., Shenzhen 518129, China
[3]School of Computer Science, Hubei University of Technology, Wuhan 430068, China
[4]College of Tongda, Nanjing University of Posts and Telecommunications, Nanjing 225127, China

Corresponding author: Xuefei Li (snowfly_li@163.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 61772180, in part by the Natural Science Foundation of the Higher Education Institutions of Jiangsu Province under Grant 17KJB520028, in part by the NUPTSF under Grant NY217114, in part by the Tongda College of the Nanjing University of Posts and Telecommunications under Grant XK203XZ18002, in part by the Qing Lan Project of Jiangsu Province, and in part by the Doctoral Scientific Research Foundation of Hubei University of Technology under Grant BSQD2019026.

ABSTRACT Point-of-Interest (POI) recommendation is one of the important services of location-based social networks (LBSNs), which has become an important way to help users discover interesting places and increase the potential income of related companies. Although human movement presents a sequential pattern in the LBSN. There still are the following problems: (1) when modeling the sequence data, most of the existing works assume that the check-in time depends on the location transformation in the location sequence. In particular, these works emphasize the equivalent transition probabilities between locations for all users to capture the check-in sequential pattern, whereas they ignore the spatial and temporal information of personalized context in some actual personal check-in scenarios; (2) most of the existing POI recommendation algorithms fail to utilize the social information related to modeling users to improve the final recommendation performance.To tackle the above challenges, we propose a new personalized successive POI recommendation model called **S**patiotemporal **S**equential and **S**ocial **E**mbedding **R**ank model, named SSSER. First, we use a hybrid deep learning model based on the convolution filter and multilayer perceptron model to mine the sequence pattern among the users' checked-in locations. Then, we use the method of metric learning to model the social relationship among users. Finally, we propose a unified framework to recommend POIs combining the users' personal interests, the check-in sequential influence and social information simultaneously for the successive POI recommendation. And the BPR standard is used to optimize the loss function to fit the user's partial order of POIs. The experimental results on the real datasets show that our proposed POI recommendation algorithm outperforms the other state-of-the-art POI recommendation algorithms.

INDEX TERMS Recommender systems, social network services, sequential analysis, neural networks.

I. INTRODUCTION

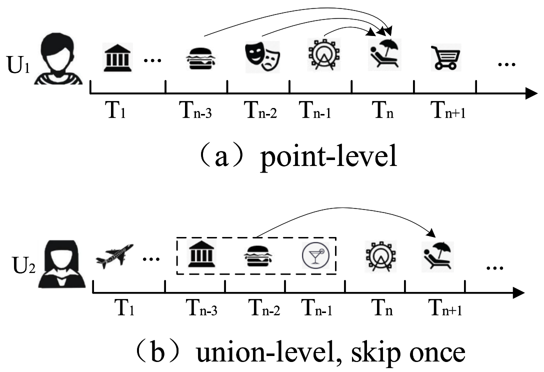
With the rapid development of Web2.0, wireless communication and location collection technology have promoted many location-based social networks (LBSNs), such as Foursquare, Yelp, Facebook and so on. Via these LBSNs, users can establish social connections with other users, explore the surrounding environment, and share their life experiences by checking in points-of-interest (POIs) such as restaurants, shopping centers, and tourist attractions. In addition to providing an interactive platform for users, LBSN contains rich data (check-in data, social relationships, comment information, etc.), which can be applied to predict users' preferences and recommend some unvisited POIs that may be of interest

The associate editor coordinating the review of this manuscript and approving it for publication was Chao Shen.

to users. The recommendation, by means of LBSN, of a geographical location that a user may be interested in is referred to as the recommendation of POI. The POI recommendation, on the one hand, satisfies the individualized needs of users to explore new geographical areas and discover new POIs and at the same time alleviates the problem of information overload faced by users. On the other hand, the recommendation of POI helps LBSN service providers to play a pivotal role in realizing intelligent location services. Therefore, how to provide users with accurate POI recommendation has drawn the attention of more and more researchers in recent years [1].

Recently, as LBSN develops, growing numbers of researchers have observed, through the movement patterns and mobile trajectories of human The characteristics of human movement are also known as spatiotemporal sequence effects [2], [3]. The users' check-in behavior is influenced by a combination of factors, but is most closely related to the spatiotemporal effects. Obviously, the POI recommendation algorithm is a location-aware personalized recommendation model based on context information. It cannot meet the requirements of accurate POI recommendation by capturing the context information, such as the geographical influence of the check-in or the social information between users and their friends [4], [5]. The POI recommendation in consideration of the influence of the spatiotemporal sequence pattern on the user's check-in behavior is called successive POI recommendation or the next POI recommendation [6].

Although researchers have proposed some successive POI recommendation algorithms in recent years, the existing successive POI recommendation algorithms still face many challenges, mainly due to the following problems:



（a）point-level

（b）union-level, skip once

FIGURE 1. An example of point and union level dynamic pattern effects.

1) The current research into successive POI recommendation mainly uses the Markov chain model or RNN to model the checkin spatiotemporal sequence pattern. The study of the literature [7], [8] utilizes the Markov chain model. However, when modeling the spatiotemporal sequence data, these models need to estimate $V_m^{\bar{n}} \cdot (V_m - 1)$ parameters in the $\bar{n}$-order Markov chain model. Each parameter corresponds to V_m states of the ($\bar{n}$-1)-th order probability value. Obviously, it involves extremely complicated calculations. Therefore, in order to reduce the size of the prediction space, most Markov chain-based works [9]–[11] use first-order Markov chain to model sequence effects, which only considers the last in the sequence of locations checked-in by the user to recommend new locations. Obviously, this method does not distinguish the personalized check-ins of users in some particular periods, so that the check-in behavior of all users is finally modeled as a unified check-in location sequence pattern. At the same time, deep learning is an important research direction in machine learning and breakthroughs have been made in the field of recommendation systems in recent years [12]. Among them, RNN brings new opportunities to capture spatiotemporal sequence effects in recommendation systems because it can model the dependencies between sequence data during different periods. However, when modeling spatiotemporal sequence data, RNN assumes that the time dependence varies monotonically with the location in the sequence (as shown in Figure1 (b)). Nevertheless, in some real check-in scenarios, not all adjacent check-ins have dependencies, so that RNN cannot produce satisfactory recommendation results [13] (for example, the skip behaviors of sequential patterns as shown in Figure1 (b)), where the impact from past check-in behaviors may skip a few steps and still have strong impact. A tourist has check-ins sequentially at an airport, hotel, restaurant, bar, and beach. While the check-ins at the airport and hotel do not immediately precede the check-in of the beach, they are strongly associated with the latter. On the other hand, checking in at the restaurant or bar does not have little effect on checkin at the beach (because they don't necessarily happen).

2) With the rapid advancement of LBSN, more and more POI recommendation algorithms use the rich information provided by social networks to improve the performance of recommendation algorithms and alleviate the data sparsity and cold start problems faced by recommendation methods. Most POI recommendation research work, like the literature [14]–[16], prefers to exploit the similarity of users' social relationships based on the traditional collaborative filtering model to model users' social links. These methods make full use of a collaborative filtering model to mine users' social links, but fail to achieve stable and reliable calculation results in the case where the number of common friends or common check-in information is scarce. Yet the literature [17] prefers to use the link-based graph model to model users' social link. However, this method is not effective in the case of active users of social networks, due to the fact that they have too many indirect friends, so the graph model cannot accurately capture the user's preferences based on the link method.

To resolve the above problems, we have studied multiple factors that affect users' check-in decisions in the POI recommendation (such as the saptiotemporal impact between users' check-in locations, check-in time and users' social link information) and the interaction between these factors and users. The key contributions of this paper is as follows:

1) We propose a hybrid deep learning model based on a convolution filter [18] and a multi-layer perceptron model [19] to capture user preferences and the effects of spatiotemporal sequence patterns. This model first embeds a series of spatiotemporal sequences of users' check-in locations and other check-in information into the latent space, regarding them as a series of ''images'' and the spatiotemporal sequence pattern and user check-in information as local features of the ''images''. Then, we use the CNN method in the joint convolution filter based on image recognition to study these local features so as to capture the modeling of spatiotemporal sequence pattern effects on check-in locations and user preferences. Unlike the ''image'' recognition, these ''images'' do not indicate input and must be learned in combination with all filters at the same time. This method effectively improves the predictive ability of users' personalized check-ins based on check-in sequence spatiotemporal effects estimation. In addition, this paper applies a multilayer perceptron to learn the final interaction function of users' check-in location spatiotemporal sequence effects and users' personalized check-ins, which will improve the non-linear modeling ability of the above-mentioned hybrid deep learning model.
2) We propose a model based on metric learning theory [20] to model users' social links. The basic idea is to capture user preferences through spatial distance. This method simultaneously trains the samples and the distance metric, updates the distance metric and the coordinates of different users while satisfying the distance constraint, and embeds the users into a unified low-dimensional space so as to generate a recommendation result via utilizing the distance between different users. Finally, a social weighting function is constructed on the basis of metric learning theory to accurately measure the degree of the social relationship between different users, so that the final recommendation is even more accurate.
3) We develop a unified framework based on matrix factorization technology to integrate users' check-in information, users' check-in the saptiotemporal impact and users' social link. Then, the BPR strategy [21] is used to optimize the loss function of the matrix factorization, so that the above-mentioned context information can be modeled more accurately. Finally, a comprehensive experimental evaluation based on real-world datasets is conducted, which demonstrates that the proposed model is effective and is significantly superior to the state-of-the-art algorithms.

II. RELATED WORK

Our work is related to two lines of literature, POI recommendation and neural networks. We review the recent advances in both areas.

A. POI RECOMMENDATION

The hybrid recommendation method, namely the POI recommendation system incorporated with multi-source heterogeneous side information, attracts even wider attention because it can alleviate the problems of data sparsity and cold start in the traditional POI recommendation systems. However, as side information tends to feature multimodality, data heterogeneity, large scale, sparse data, uneven distribution, among other complicated characteristics, the hybrid recommendation research incorporated with multi-source heterogeneous data still faces severe challenges.

Many recent works [2], [22]–[24] indicate that there is a strong correlation between users' check-in activities and geographic distances, spatiotemporal relationships of different check-in locations and social relationships. Therefore, the current research into POI recommendation mainly focuses on using geographical influences, spatiotemporal relationships of different check-in locations and social influence to improve the accuracy of POI recommendation. For example, Ye et al. proposed a friend-based collaborative filtering (FCF) method for POI recommendation, which proves that users' social link is effective for improving the final recommendation performance [25]. Then, spatiotemporal information and geographic restrictions are proven to be valid for the recommendation. Cheng et al. proposed a tensor-based FPMC-LR model that considers the first-order Markov chain of POI transformation and localized region constraint [26]. Yin *et al.* proposed a unified probability generation model (namely TRM), which is based on the discovery of the semantic, temporal and spatial patterns of users' check-in activities to construct a unified modeling of the combined effects of the above patterns of users, thus deciding to choose the POI to be visited [27]. Liu *et al.* proposed a dual-weighted low-rank graph model that combines static user interests with changing user POI sequence preferences and time intervals so as to achieve the POI recommendation [28]. Zhao *et al.* developed a ranking-based pairwise tensor factorization framework called STELLAR for the recommendation of the POI [4]. STELLAR combines fine-grained contextual information of check-in and generates significant improvements. All these efforts attempt to adapt to the model by maximizing the interaction between users and POI, with recommendations made based on the last POI check-in. In addition, Xie et al. proposed a novel embedded learning-based recommendation model for the next POI recommendation [29], which uses the bipartite graph model to model related contextual factors in the POI recommendation process and is called the GE Model [29]. Four pairs of contextual factors, namely POI-POI, POI-region, POI-time, and POI-word are modeled in a unified optimization framework. The experimental results show that GE is significantly superior to other competitor methods algorithms in the POI recommendation.

B. NEURAL NETWORK

In recent years, neural networks (deep learning) have made great breakthroughs in the fields of image processing, natural language processing, and speech recognition, and have become an upsurge in artificial intelligence, bringing new opportunities for the research of recommendation systems.

In various neural network structures, an MLP with a single hidden layer containing a sufficient number of non-linear elements can approximate any continuous function on the compact input domain to arbitrary precision [30]. Recently, He *et al.* developed a matrix factorization method based on a deep neural network for the collaborative filtering of implicit feedback data [31]. Based on item embedding and user embedding, they apply multi-layer MLP to extract advanced hidden features by maximizing user-item interaction. Chen et al. studied the problem of personalized location-aware news recommendation [32]. By adding a location channel in DSSM, MLP learns the implicit representation of users, news and locations from user information, news information and local location topic distribution, and finally combines the three aspects of information to calculate the degree of relevance between user interest and news content at specific locations so as to generate news recommendations.

Recurrent neural networks (RNN) [33]–[35] has been widely used to model sequence data and are therefore rapidly being introduced into the study of user behavior sequence pattern modeling. However, the main idea of collaborative filtering based on recurrent neural network is to model the influence of user check-in history sequence on user check-in behavior through the recurrent neural network, so as to generate users' item recommendation list and behavior prediction. The collaborative filtering method based on recurrent neural network and its variants, namely the long-term short-term memory (LSTM) [36] and the gated recurrent unit (GRU) [37], are widely used in current recommendation systems, because they have a high degree of applicability and can effectively model the sequence pattern of user check-in behavior. What's more, they can integrate spatiotemporal information and other context information [39] and various types of auxiliary data to improve the quality of recommendation [40] by changing the input of the recurrent neural network [35] and defining different weight matrixes [38]. However, research on the use of neural networks for the next POI recommendation is very limited. Liu et al. proposed a classic RNN-based neural network model [13], called STRNN [13]. STRNN uses a time transition matrix and a distance transition matrix within the framework of the RNN model. Yin et al. gained inspiration from DBN's better performance than SDA in classification tasks and its robustness to noise, used DBN to learn the semantic representation of POI from the heterogeneous features in users' check-in process, and then integrated the obtained spatiotemporal personal preferences and the semantic representation of POI to propose a probabilistic model framework called Spatial-aware Hierarchical Collaborative Deep Learning (SH-CDL) [41].

The model we propose differs from the above-mentioned models in the following aspects. First, this paper proposes a model based on CNN. Convolutional neural networks are mainly used in the recommendation system to extract hidden features of items from images and text content, thereby obtaining low-dimensional vector representations of items, and generating recommendations list for users in combination with users' implicit representation. However, we use the CNN model-based method to capture images' local feature search mode in image recognition to realize the modeling of users' check-in spatiotemporal patterns. Secondly, we have designed a better interaction function based on a multilayer perceptron, which is used to model the potential feature interaction between users, check-in locations and users' check-in spatiotemporal patterns so as to improve its ability to capture the user interaction data structure. The final experiment also verifies this. Thirdly, this paper adopts the metric learning theory to model users' social links, and integrates the above-mentioned context information for POI recommendation. Finally, the BPR strategy is used to optimize the final objective function, which accurately fits users' preferences for POI.

III. PROBLEM FORMULATION AND PRELIMINARIES

In this section, we first formulate the POI recommendation problem of this paper, and then introduce the necessary preliminaries.

A. PROBLEM FORMULATION

In most LBSNs POI recommendation normally consists of two sets: assuming the set of all users is $U = \{u_1, u_2, \cdots, u_{|U|}\}$ and the set of all POIs is $L = \{l_1, l_2, \cdots, l_{|L|}\}$. The set of POI visited by each user u is obtained from the collection L. The sequence of a user is $S = \{S_1^u, S_2^u, \cdots, S_i^u, \cdots S_\iota^u\}$, where $S_i^u \in L$, and its length is ι. The index t ($t \in T$) of S_t^u indicates that the order of action is in the sequence S, where T is the check-in time collection.

Each POI is geocoded using <Longitude, Latitude> in addition to its unique identifier. Users' check-in record constitutes a User-POI check-in frequency matrix $F \in \mathbb{R}^{U \times L}$, and each element f_{u_i,l_n} in the check-in frequency matrix F represents the check-in frequency of the user u_i on the POI l_n. The users' check-in frequency reflects their preference degrees of POI. Normally, the user only visits a small portion of the POIs, so the user-POI check-in frequency matrix F is extremely sparse. The purpose of POI recommendation is to use users' check-in data and other context information in the LBSN to predict the frequency $\hat{f}_{u_i,l_n}$, a user's frequency to visit an unchecked-in POI, and provide the user with a list of POIs that may be of interest to them according to the predicted frequency.

B. METRIC LEARNING

In the machine learning field, a metric (or distance function) is a function that defines the distance between the elements in a set. A set with metrics is called a metric space [42]. The main purpose of metric learning is to learn and obtain the optimal parameters that satisfy the upper constraints. Using the training samples to learn a metric matrix A allows the resulting distance function $d_A(x_i - x_j)$ to improve the performance of the learning algorithms or to meet certain application requirements.

Supervised distance metric learning can be divided into two categories [43]: global distance metric learning and local distance metric learning. In general, the training samples of supervised distance metric learning are divided in the form of pairwise constraints. The former learn distance metrics in a global sense by keeping all the data points in each class close together while satisfying all pairwise constraints at the same time and ensuring that data points from different classes are separated. The purpose of global distance metric learning is to learn and then obtain a distance metric based on given constraints. The latter is to learn distance metrics in local settings, which means that not all pairwise constraints are satisfied at the same time, but only "local" pairwise constraints are satisfied.

The proposed method in this paper applies the global distance metric learning to the recommendation system, and the distance metric is represented as matrix $A \in \mathbb{R}^{l\times l}$. The distance between any two points x_i and x_j is expressed as:

$$g_A^2(x_i, x_j) = \left\|x_i - x_j\right\|^2 = (x_i - x_j)A(x_i - x_j)^T \qquad (1)$$

where A is a semi-positive definite matrix used to ensure that the distance is non-negative and symmetrical. The global optimization problem under constraints can be expressed as:

$$\begin{aligned} &\min_A \sum_{(x_i,x_j)\in B} \left\|x_i - x_j\right\|_A^2 \\ &s,t.\, A \geq 0, \sum_{(x_i,x_j)\in D} \left\|x_i - x_j\right\|_A^2 \geq \theta \end{aligned} \qquad (2)$$

where B denotes an equivalence constraint collection in which x_i and x_j belong to the same class of samples, and D denotes a non-equivalence constraint collection in which x_i and x_j belong to different classes of samples. θ is a constant that limits the minimum distance between data points in different classes of samples.

C. CONVOLUTIONAL NEURAL NETWORK

More attention was paid to applying convolutional neural networks on recommendation [12], [18]. The neural network, which has plentiful neurons, consists of convolutional layers, some of which are followed by pooling layers, and connected layers with a final softmax [45]. Using multiple layers of neurons to represent some functions are much simpler. Each of neurons can have different values of weights and biases. Weights and biases are network parameters. We discuss a model of a neuron. For the neuron **G** is input $(x_1, x_2, \ldots, x_n)$, the weight $(w_1, w_2, \ldots, w_n)$ and the bias (b_0), the neuron output value is $G = f_1(x_1\, w_1 + x_2\, w_2 + \ldots. + x_n\, w_n + b_0)$. Now G is already a certain value. The function f_1 is defined as a non-linear activation function, and since it only accepts a function of one variable, it is not complicated. This not only ensures the flexibility of the artificial neural network, but also makes the neuron function not too complicated. According to the already defined layer and equation, let the calculation pass to the back layer in turn. The result is finally obtained at the output layer.

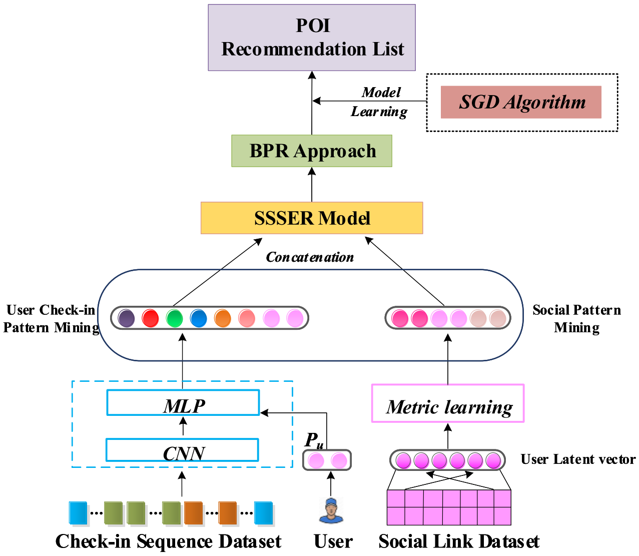


FIGURE 2. The framework of SSSER model.

Our ultimate goal is to obtain a personalized and informative description, so we introduce patiotemporal sequential and social information and further provide an improved definition. The spatiotemporal model utilize the Convolutional Neural Network(CNN) and Multilayer Perceptron(MLP) to capture potential features of check-ins matrix $E^{(u,t)}$ by user. our problem can be formally defined as generating a objective function $\hat{y}$, based on the convolutional neural network, multilayer perceptron as well as the metric learning.

IV. THE SSSER MODEL

In this section, we present the proposed hybrid neural network framework for the next POI recommendation task, named SSSER. Specifically, we first propose a hybrid deep learning model to learn the spatiotemporal characteristics of users' check-in POI in this section. At the same time, the metric learning theory is used to model users' social link, and then the matrix factorization technology is used to integrate the two kinds of information. Finally, the BPR strategy is applied to optimize the loss function and users are fitted to the partial order relationship of the POIs. The SSSER model framework is shown in Figure 2.

A. SPATIOTEMPORAL MODEL

This paper proposes a convolution sequence embedding model for capturing the effects of the top-N locations'

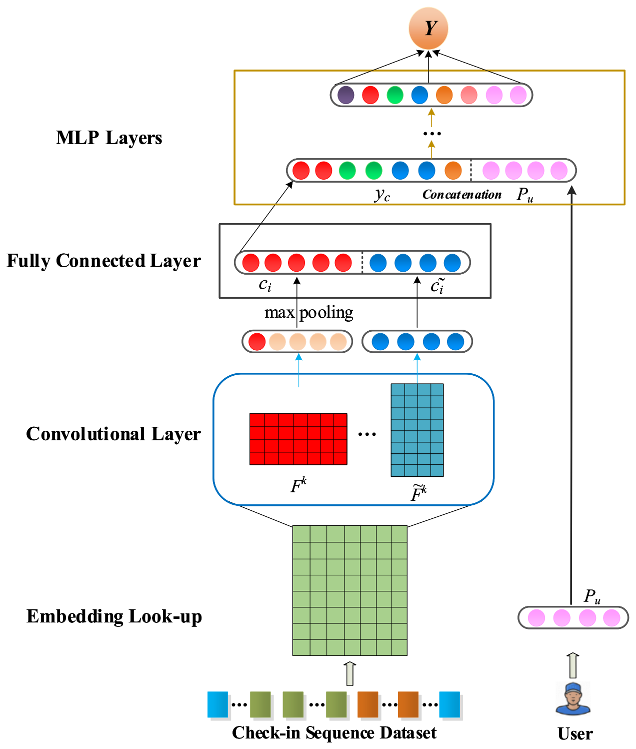


FIGURE 3. The framework of spatiotemporal model.

check-in spatiotemporal sequence information. The proposed model uses a convolutional neural network (CNN) to learn spatiotemporal sequence features while using latent factor models to learn the latent features of users. The model in this paper regards the check-in location matrix as the embedded matrix of the potential space regards this matrix as the "image" of the historical L check-in locations in the potential space, and regards the spatiotemporal sequence of the check-in location as a local feature of the "image". It has also made use of the advantages of the convolution filters in Convolutional Neural Networks (CNN), namely their capacity to efficiently capture local features in image recognition and natural processing. As shown in Figure 3, the spatiotemporal model consists of four parts: the Embedding look-up layer, the Convolution layer, the Fully Connected layer, and the MLP Layers.

1) EMBEDDING LOOK-UP

The task of the Embedding layer is to transform the collection of locations into a document matrix consisting of word vectors. In order to model users' spatiotemporal patterns, each user u is associated with a sequence of locations from the location collection L. Each user u is associated with a sequence of locations $S = \{S_1^u, S_2^u, \cdots, S_i^u, \cdots S_t^u\}$.

We provide the previous M locations to the hybrid neural network proposed in this paper to capture the sequence features in the potential space. Location l is embedded into $Q_l \in \mathbb{R}^d$, d is the number of potential dimensions. The role of Embedding Look-up is to retrieve the embedding of the top M locations and superimpose them to get the matrix $E^{(u,t)} \in \mathbb{R}^{M \times d}$ of the user u during a certain period of time, as represented in the green grid in Figure 3.

$$E^{(u,t)} = \begin{bmatrix} Q_{S^u_{t-M}} \\ \vdots \\ Q_{S^u_{t-2}} \\ Q_{S^u_{t-1}} \end{bmatrix} \tag{3}$$

Apart from the embedding operation on the location, embedding operation $P_u \in \mathbb{R}^d$ is also conducted on users, as represented in the pink circles in Figure 3.

$$P_u = \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_d \end{bmatrix} \tag{4}$$

2) CONVOLUTIONAL LAYER

Inspired by the idea of literature [44], [45], our model represents the top-M check-in locations in the historical check-in record as the $M \times d$ matrix, where d is the number of potential dimensions, and the rows of the matrix save the spatiotemporal sequence of check-in locations. This embedded matrix is then treated as an "image" of the M check-in locations in the potential space, and multiple convolution filters are used to jointly capture the sequence of the local features of this "image". The convolution kernel of the convolutional layer in CNN continually slides from left to right and top to bottom on the matrix E after embedding to capture continuous spatiotemporal features. This paper uses joint horizontal and vertical convolution filters to search for spatiotemporal sequence patterns.

Horizontal convolution kernel. In Figure 3, the horizontal convolution kernel F^k has a size of $h \times d$ and the number of ν, where h is height of a filter and the full width equal to d, respectively. The horizontal convolution kernel F^k will start from the upper left corner of the matrix E and slide from top to bottom in sequence with a step size of 1. It interacts with all horizontal dimensions E of the location l_i, $1 \leq i \leq M - h + 1$. The result of the interaction is calculated by the i^{th} convolution value:

$$c_i^k = f_c(E_i * F^k + b_i), \quad 1 \leq i \leq M - h + 1 \tag{5}$$

where $*$ is the convolutional operator,b_i is the i^{th} biased vector, and $f_c(\cdot)$ is the activation function for convolution layers. In this paper, the activation function uses ReLU. And $f(x) = \max(0, x)$ defines the nonlinear output of the neuron after linear transformation$(w * x + b)$. It is more biologically sound and has proven to not cause supersaturation [45].

Each convolution kernel interacts with matrix E to map a new matrix. The convolution mapping attribute vector formed by all the convolution mapping attribute values extracted by the k^{th} convolution kernel can be expressed as:

$$c_i^k = (c_1^k, c_2^k, \cdots c_{M-h+1}^k) \tag{6}$$

Then conduct the maximum pooling operation for $c_i^k = (c_1^k, c_2^k, \cdots c_{M-h+1}^k)$. The set of c_i^k represents all mapping attributes and does not represent the most significant feature, so the maximum value is extracted, through maximum pooling, from all the values generated by the specific convolution kernel. The maximum value captures the most important feature extracted by the filters.

$$c_i = (\max(c_i^1), \max(c_i^2), \cdots \max(c_i^v)) \tag{7}$$

The horizontal filters interact with each successive h iterms and matrix E by their embedding. The embedding and convolution kernels are learned to minimize the objective function of the prediction error of the encoding target term.

Vertical convolution kernel. In Figure 3, the green dotted line indicates the vertical convolution kernel $\tilde{F}^k$, and its size is $M \times 1$, and its number's range is $1 \leq k \leq \tilde{v}$. There are $\tilde{v}$ vertical filters. Each filter $\tilde{F}^k$ interacts with the matrix E by sliding $(d-1)$ times from left to right on the matrix E. The result of the interaction is calculated by the i-th convolution value:

$$\tilde{c}_i^k = f_c(E_i * \tilde{F}^k + b_i) \tag{8}$$

where $*$ is a convolutional operator. b_i is the i-th biased vector. Each convolution kernel interacts with matrix E to map a new matrix. The convolution mapping attribute vector formed by all the convolution mapping attribute values extracted by the k^{th} convolution kernel can be expressed as:

$$\tilde{c}_i = (\tilde{c}_i^1, \tilde{c}_i^2, \cdots, \tilde{c}_i^{\tilde{v}}) \tag{9}$$

After the convolution of the vertical convolution kernel, we do not conduct the maximum pooling to guarantee that we do not lose too many potential dimensions.

3) FULLY-CONNECTED LAYERS

In the fully-connected layer, this is a process of highly purified feature, and in order to prevent overfitting, a Dropout layer is added to the fully-connected layer. After Dropout, merge c_i and $\tilde{c}_i$ before the fully-connected layer and use this result as an output value, which is used as an input to the fully-connected layer:

$$y_c = f_a(w \begin{bmatrix} c_i \\ \tilde{c}_i \end{bmatrix} + b) \tag{10}$$

where $w \in \mathbb{R}^{d \times (v + d\tilde{v})}$ is a nonlinear mapping matrix which is a weight matrix that projects the fully-connected layer into a d-dimensional hidden layer. $f_a(\cdot)$ is the activation function ReLU for fully connected layer. $b \in \mathbb{R}^d$ is the biased vector.

4) MULTILAYER PERCEPTRON

The two components described above extract the different latent features from two different information sources. It is expected that the integration of the two components could complement each other and yield better prediction performance. We use the standard MLP (Multilayer Perceptron) to learn the interaction between the user and the latent features of the check-in POI sequence, which makes the model to have high-level flexibility and nonlinear modeling capability. More precisely, the MLP model under our SSSER framework is defined as:

$$Y_1 = \varphi_1(P_u, y_c) = \begin{bmatrix} p_u \\ y_c \end{bmatrix} \tag{11}$$

$$\varphi_2(Y_1) = f_2(W_2^T Y_1 + b_2) \tag{12}$$

$$\cdots\cdots\cdots\cdots$$

$$\varphi_{out}(Y_{X-1}) = f_X(W_X^T Y_{X-1} + b_X) \tag{13}$$

where W_X, b_X, f_X, φ_X and φ_{out} represent the weight matrix in the X-layer perceptron, the biased vector (the neuron threshold value of the neural network), the activation function, the x-th layer and the mapping function for the output layer respectively.

The objective function to learn POI spatiotemporal features through CNN and MLP is expressed as:

$$Y = \sum_i^L \left\| \tilde{v}_i - f_X(W_X^T (f_{X-1}(\cdots f_2(W_2^T \begin{bmatrix} Pu \\ y_c \end{bmatrix} + b_2) \cdots)) + b_X) \right\|^2 + \lambda_W \sum_\alpha^{|W|} \| W_\alpha \|^2 \tag{14}$$

where $\tilde{v}_i$ and λ_W represent the initial value of the POI latent features and regularization parameter, respectively.

B. SOCIAL LINK MODEL

In social networks, whether there is a social relationship between users tends to be dependent on the level of mutual trust between users. Considering that users' trust levels are not the same for each friend, then how to estimate the trust levels between users constitutes one of the important issues that affect the further improvement of the recommendation algorithm [46], [47].

The main idea of distance metric learning is to learn an expected distance metric, making the distribution of same-class samples more compact and the distribution of samples from different classes looser. And the metric learning algorithm is designed to minimize the distance between each users and their friends while maximizing the distance from users they don't like. Based on this, if the user is closer to a certain user, it indicates that this user is positively related to that user and vice versa; if the user is far away from a certain user, it indicates that this user is negatively related to that user. Therefore, we apply a metric learning-based algorithm to construct a weight function for the prediction of the degree of the social relationship between different users so as to improve recommendation performance. The metric learning algorithm is used to predict the degree of trust between users.

To learn distance metric, a suitable distance metric form must be defined in the first place. The pairwise constraint set is constructed as follows [48]: given two users, if one user has a closer distance to the other, the user-user pair is categorized into an equivalence constraint collection; if the user is farther away from the other, then the user-user pair is included into a non-equivalence constraint collection. Given a pair of users u_m and u_n. The way we learn these user vectors is their Euclidean distance. The Euclidean distance between the user vectors $u_m \in \mathbb{R}^d$ and $u_n \in \mathbb{R}^d$ can be used to estimate the degree of relationship between them:

$$g(u_m, u_n) = \ ||u_m - u_n|| \tag{15}$$

The higher the degree of the social relationship between the users, the smaller the Euclidean distance is. In order to make the users who have a high degree of relationship much closer, we use the following loss function to model such a constraint:

$$w(g) = \sum_{(u_m,u_n)\in U} \sum_{(u_m,u_r)\notin U} \rho_{mn} \cdot \sigma[\vartheta + g\left(u_m, u_n\right)^2 - g\left(u_m, u_r\right)^2] \tag{16}$$

where the relationship between user u_m and user u_n is closer than that between user u_m and user u_r. $\sigma[x] = \max(x, 0)$ represents the standard hinge loss, ρ_{mn} is the sequence loss weight, and ϑ is the safety margin size to prevent the result from being 0.

To punish lower-rank socially connected users, we use a rank-based weighting scheme to punish lower-rank socially connected users and adopt the same method [49] called Weighted Approximate Rank Pairing (WARP) loss . Given a metric q, let U denote the total number of users and $rank_q(u_m, u_n)$ denote the rank of user u_n as the recommended use of user u_m, and we punish the positive user u_n by the setting according to its rank.

$$\rho_{mn} = \log(rank_q(u_m, u_n) + 1) \tag{17}$$

where q is the given metric. This scheme penalizes a positive user at a lower rank much more heavily than one at the top, and produces the state-of-the-art results in many prior works [48], [49].

C. JOINT FRAMEWORK

Based on the above subsections, we use the matrix factorization method to model users' preferences and effectively combines social relationship and spatiotemporal information so as to integrate the two kinds of heterogeneous information into a unified framework. The SSSER POI recommendation algorithm is proposed to provide the next POI recommendation services. Similarly to [31], we can formulate the objective function as follows:

$$\begin{aligned}\hat{y} &= \min_{U,V} \sum_{(u_m,u_n)\in U} \sum_{(u_m,u_r)\notin U} \rho_{mn} \cdot \sigma[\vartheta + g(u_m, u_n)^2 - g(u_m, u_r)^2] \\ &\quad + \sum_{i}^{L} \left\| \tilde{v}_i - f_X(W_X^T(f_{X-1}(\cdots f_2(W_2^T \begin{bmatrix} Pu \\ y_c \end{bmatrix} + b_2)\cdots)) + b_X) \right\|^2 \\ &\quad + \lambda_W \sum_{\alpha}^{|W|} \|W_\alpha\|^2\end{aligned} \tag{18}$$

According to the literature [21], when providing POIs for users in this paper, the Bayesian Personalized Ranking [21] strategy is used to fit the user's partial ordering on POIs in this paper so as to optimize the models mentioned above. The POI recommended by a given user is modeled as a ranking problem so that the POI-based ranking list is eventually learned.

The goal of our successive personalized POI recommendation is to recommend top-k new POI to users. Thus, we can model it as a ranking over locations. The following definition is then obtained:

$$l_i >_{u,t} \ l_j : \ \Leftrightarrow \hat{y}_{u,t,l_i} \ > \hat{y}_{u,t,l_j} \tag{19}$$

For locations $l_i \in L$, to find the correct-ranking is to maximize the calculation of the following posterior probability based on the Bayesian equation. Then, according to the partial order relationship $>_{u,t}$ of the user u at all POIs at time t, a recommendation list is finally generated. According to BPR strategy, we get:

$$p\left(\Theta| >_{u,t}\right) \propto p\left(>_{u,t} |\Theta\right) p(\Theta) \tag{20}$$

where Θ is the SSSER model parameter.

It is assumed that each user's check-in at the POI is independent, and that for a specific user, the partial order relationship at a specific POI location pair (l_i, l_j) is independent of that at other POI location pairs. Furthermore, the set of training data D_l based on pairwise constraints is defined as $D_l := \left\{(u, t, l_i, l_j)|l_i >_{u,t} l_j\right\}$. The likelihood function of the partial order relationship of all users at all POIs is:

$$\begin{aligned}\prod_{u\in U} p(>_{u,t} \mid \Theta) = \prod_{D_l \in U\times T\times L} p(l_i >_{u,t} \ l_j| \ \Theta)^{\gamma((u,t,l_i,l_j)\in D_l)} \\ \cdot (1 - p(l_i >_{u,t} \ l_j| \ \Theta)^{\gamma((u,t,l_i,l_j)\notin D_l)})\end{aligned} \tag{21}$$

where γ is the indicator function.

$$\gamma\ (b) = \begin{cases} 1 & \text{if } b \text{ is True} \\ 0 & \text{other} \end{cases} \tag{22}$$

We extend $>_{u,t}$ on all POI pairs (l_i, l_j) and consider the overall attribute and asymmetric attribute of users' partial order relationship, Equation (21) is further simplified to:

$$\prod_{u\in U} p(>_{u,t} \mid \Theta) = \prod_{\text{ï£¡}u,t,l_i,l_j\text{ï£¡}\in D_l} p(l_i >_{u,t} \ l_j| \ \Theta) \tag{23}$$

Next we use the definition in Equation (21) to represent $p(l_{i>u,t} \ l_j| \ \Theta)$, then we get:

$$p(l_i >_{u,t} \ l_j| \ \Theta) = p(\hat{y}_{u,t,l_i} > \hat{y}_{u,t,l_j}| \ \Theta) \tag{24}$$

We use the Sigmoid function $\delta(z)$ to define $p(z > 0) = \delta(z) = \frac{1}{1+e^{-z}}$, which indicates the probability of user u's a preference for POI l_i being higher than that for POI l_j at time t:

$$\begin{aligned}p(l_i >_{u,t} \ l_j| \ \Theta) = \delta(\hat{y}_{u,t,l_i,l_j}) = \delta(\hat{y}_{u,t,l_i} - \hat{y}_{u,t,l_j}) \\ = \frac{1}{1 + e^{-(\hat{y}_{u,t,l_i} - \hat{y}_{u,t,l_j})}}\end{aligned} \tag{25}$$

Further, we introduce the zero-mean Gaussian prior with the covariance of $\frac{1}{\lambda_\Theta}$ to the model parameters, which means $\Theta \sim N(0, \frac{1}{\lambda_\Theta})$.

Similarly to GeoMF++ [60], by combining the Equation (20)~(25), we can minimize the following objective function of the SSSER model:

$$\begin{aligned}\hat{L} &= \min \ln p\left(>_{u,t} |\Theta\right) p(\Theta) \\ &= \min\ 1\text{n} \prod_{u\in U} \prod_{L_u^t \in L_u} \prod_{i\in L_u^t} \prod_{j\notin L_u^t} \delta\left(\hat{y}_{u,t,l_i} - \hat{y}_{u,t,l_j}\right) p(\Theta) \\ &= \min \sum_{u\in U} \sum_{L_u^t \in L_u} \sum_{i\in L_u^t} \sum_{j\notin L_u^t} \left(-\ln \delta\left(\hat{y}_{u,t,l_i} - \hat{y}_{u,t,l_j}\right) + \lambda_\Theta \|\Theta\|^2\right)\end{aligned} \tag{26}$$

where λ_Θ is the regularization parameter corresponding to $\delta(z)$ to prevent overfitting during the learning process.

D. OPTIMIZATION

The objecti function optimization can be done by performing stochastic gradient descent (SGD) according to the literature [31].

$$\begin{aligned}\frac{\partial \hat{L}}{\partial u_m} = \frac{-e^{-\hat{y}_{u,t,l_i,l_j}}}{1+e^{-\hat{y}_{u,t,l_i,l_j}}} \sum_{(m,n)\in U} \sum_{(m,r)\notin U} \rho_{mn} \frac{\partial}{\partial u_m} \sigma(g\,(u_m, u_n)^2 \\ - g\,(u_m, u_r)^2) + 2\lambda_{t,m} \cdot u_m\end{aligned} \tag{27}$$

$$\begin{aligned}\frac{\partial \hat{L}}{\partial u_n} = \frac{-e^{-\hat{y}_{u,t,l_i,l_j}}}{1+e^{-\hat{y}_{u,t,l_i,l_j}}} \sum_{(u_m,u_n)\in U} \sum_{u_r\notin U} \rho_{mn} \frac{\partial}{\partial u_n} \sigma(g\,(u_m, u_n)^2 \\ - g\,(u_m, u_r)^2) + 2\lambda_{t,n} \cdot u_n\end{aligned} \tag{28}$$

$$\begin{aligned}\frac{\partial \hat{L}}{\partial v_i} &= \lambda_W \frac{-e^{-\hat{y}_{u,tl_i,l_j}}}{1+e^{-\hat{y}_{u,t,l_i,l_j}}} \\ &\times \sum_{i=1}^{L} (\tilde{v}_i - f_N(W_X^T f_{X-1}(f(w'z+b')))) \frac{\partial \tilde{v}_i}{\partial v_i} + 2\lambda_i v_i\end{aligned} \tag{29}$$

Meanwhile, for each parameter Θ, the update process is performed as follows:

$$\Theta \leftarrow \Theta + \alpha\left(\frac{\partial}{\partial \Theta} \ln \sigma(\hat{y}_{u,t,l_i} - \hat{y}_{u,t,l_j}) - 2\lambda_\Theta \Theta^2\right) \tag{30}$$

where α is the learning rate and Algorithm 1 describes the steps of parameter learning in SSSER model.

E. TIME COMPLEXITY

The time complexity calculation of the SSSER model depends on the calculation of the objective function $\hat{L}$ and the process of calculating the gradient iteration. The model proposed in this paper is mainly divided into two components: spatiotemporal information modeling and social link modeling. For the learning of spatiotemporal information features, when calculating the latent feature vectors used to update the weights, the complexity to update is governed by the calculation of the convolutional layer. Therefore, the time complexity to update the weights and bias variables in the convolutional neural network is $O(L \cdot d_e \cdot \text{n}_l)$. N_p indicates the number of POIs, d_e indicates the embedding dimension, and n_l indicates the number of spatiotemporal features to be learned. The MLP layer requires more time and its time complexity is $O(\sum_{\tilde{n}=1}^{N_{\tilde{m}}} z_{\tilde{n}-1} z_{\tilde{n}})$ with $z_{\tilde{n}-1}$ denoting the size of the $\tilde{n}$-th layer. And the time complexity of social link modeling is $O(d_s^3 + n_s d_s^2)$, where n_s is the number of training samples and d_s is the dimension of the data. Therefore, the total time complexity of SSSER is $O = (N_p \cdot d_e \cdot \text{n}_l + \sum_{\tilde{n}=1}^{N_{\tilde{m}}} z_{\tilde{n}-1} z_{\tilde{n}} + d_s^3 + n_s d_s^2)$.

Algorithm 1 Learning Algorithm of SSSER

```
Initialize hyperparameter
Randomly initialize $F, \tilde{F}^k, w\ ,\ b\ ,\ W_X, b_X\ ,\ \lambda_\Delta,\ \rho_{mn}$
Input: Check-in collection $L$ and social link $U$
Output: $F, \tilde{F}^k, w, b, W_X, b_X, \rho_{mn}, \Theta$
while not convergence do
    for 1 to The maximum number of iterations do
        Iterative update parameters for each calculation
        $w, b, W_X, b_X$
    end for
    for each pair, sample $U$ negative user and approximate
    $\rho_{mn}$
        update parameters $\rho_{mn}$
        Take a gradient step to optimize Equation (16)
    end for
    Update parameters according Equation (27)-(30)
    until Equation (26) converges
end while
```

V. EXPERIMENT

A. EXPERIMENTAL DATASET

We use two public datasets, Foursquare [50] and Gowalla [51], to evaluate the performance of our proposed recommendation algorithm.

Similary to [22], we removed users and POIs which have less than 10 or 15 check-in records respectively in the Foursquare dataset,. After filtering, the number of users is 18,737, the number of POIs is 32,510, the number of check-ins is 1,278,274 and the sparsity of the user POI check-in matrix is 99.79%.

In the Gowalla dataset, users and POIs with less than 10 check-in records are manually removed [22]. In the filtered dataset, the number of users is 24,941, the number of POIs is 28,593, the number of check-ins is 1,196,248 and the sparsity of the user POI check-in matrix is 99.83%. Check-in time ranges from April 2012 to September 2013. Therefore, these two datasets are very sparse.

Therefore, in these two datasets, a user should check-in at least five different POIs. In the experiments, 70% of the dataset is randomly selected as the training dataset, and the next 10% is used as the validation set to search for the optimal

TABLE 1. Statistics of the two datasets.

Dataset	Number of users	Number of POIs	Number of check-ins	Number of social links	Sparsity
Foursquare	18737	32510	1278274	950327	99.79%
Gowalla	24941	28593	1196248	47164	99.83%

hyperparameter settings for all models, with the remaining 20% being used as the test dataset. The datasets statistics are shown in Table 1.

B. EVALUATION METRICS

We evaluate the recommendation quality of the SSSER by the four wide-use metrics. i.e., precision (Precision @ N) [48], recall (Recall @ N) [48], mean average precision (MAP) [48], and normalized depreciation cumulative gain (nDCG) [48]. Given a list of top N predicted POIs for a user, denoted $\hat{R}_N$, and the last actions in users' sequence.

$$Precision@N = \frac{|R \cap \hat{R}_N|}{N} \tag{31}$$

$$Recall@N = \frac{|R \cap \hat{R}_N|}{|R|} \tag{32}$$

where $N \in \{1, 5, 10\}$.

The equation for Average Precision (AP) is:

$$\mathrm{AP} = \frac{\sum_{N=1}^{|\hat{R}|} \mathrm{precision}@N \times \mathrm{rel}(N)}{\left|\hat{R}\right|} \tag{33}$$

when the N-th location in $\hat{R}$ is also in $\boldsymbol{R}$, then $\mathrm{rel}(N) = 1$. The mean of each class of AP is taken as MAP.

nDCG is one of the important indicators used in the field of information retrieval to measure the quality of ranking. For the target user u, the Equation to calculate the nDCG value of the POI at the location of e in the recommendation list is:

$$\mathrm{n}DCG@e = \sum_{u} \frac{1}{Y_u} \sum_{n=1}^{e} \frac{2^{rel_n - 1}}{\log_2(e+1)} \tag{34}$$

where Y_u indicates the largest DCG value of the user u, and rel_n represents the correlation between the n-th POI and the user. $rel_n = 1$, which indicates correlation; otherwise $rel_n = 0$.

C. EXPERIMENTAL SCHEME

In order to verify the effectiveness and advancement of the proposed algorithm, the algorithm is tested from four different perspectives to verify its effectiveness.

(1) The Spatio-SSSER method (This method correspondingly only ignores the social interaction in SSSER.) proposed in this paper is compared with four state-of-the-art spatiotemporal modeling methods to verify the effectiveness and advancement of the proposed Spatio-SSSER method. (2) The social link weight calculation modeling method proposed in this paper is compared with four state-of-the-art social link modeling methods to verify the effectiveness and superiority of the Social-SSSER method (This method correspondingly only ignores the spatiotemporal effect in SSSER.) proposed in this paper. (3) The proposed SSSER model is applied to compare and analyze the contribution of the two components of the users' check-in spatiotemporal pattern effects and users' social link, namely the Spatio-SSSER and Social-SSSER, to the evaluation indicators of the recommendation system. (4) We discuss the influence of relevant parameters on the final recommendation performance.

This paper compares the Spatio-SSSER spatiotemporal modeling method with the following four state-of-the-art spatiotemporal modeling algorithms:

[1] FPMC [52]: FPMC models spatiotemporal sequence pattern based on the first-order Markov model and integrates users' check-in transformation matrix based on the matrix decomposition technique.

[2] Fossil [8]: Fossil simulates high-order Markov chains and uses a similarity model instead of LFM to model user preferences.

[3] HRNN [53]: This is the session-based recommendation proposed in [8]. This model uses RNN to capture sequence dependencies and make predictions.

[4] PRME [54]: A personalized ranking metric model (PRME) is proposed to jointly model sequence information and personal preferences. By mapping each POI into the low-dimensional Euclidean potential space of an object, we then use the metric embedding algorithm to effectively calculate the locational transition in the Markov chain model.

In order to verify the validity and advancement of the social link proposed in this paper, we choose the following three models:

[1] iGSLR-FCF: The literature [55] uses the FCF model to calculate the similarity of users by using the social links of users.

[2] USG-CIFCF: In the literature [25], based on the similarity of the user's check-in POI behavior, the cosine similarity calculation formula is used to calculate users' similarity based on the POI jointly checked-in by users.

[3] Trust-FCF: In the literature [56], based on the assumption that users' behaviors are influenced by the ratings and comments of other users and that their own ratings and comments also influence other users, we consider explicit trust, implicit trust and users' different roles in social links while predicting trust between users.

[4] SPRE [57]: The SPRE model maps each user to the objects in the low-dimensional Euclidean potential space and uses the prevailing metric embedding algorithm to efficiently calculate the social links between users, thus enabling social embedding models to effectively alleviate the data sparsity

problem of social links and further improving the recommendation performance.

The SSSER proposed in this paper is compared with three state-of-the-art POI recommendation models:

[1] FPMC-LR [26]: The FPMC-LR model proposes a POI model based on the extended local region constraints to factorize personalized Markov chains and consider geographical constraints of neighboring locations.

[2] LTSCR: The literature [6] uses a unified linear model to integrate user preferences, social link information, users' check-in time, and the check-in POI sequential effects to conduct POI recommendation. This model also applies weighted estimates of pairwise sorting function to improve recommendation performance.

[3] GE [29]: A recommendation model based on a graph embedding method is proposed. This method uses the bipartite graph to model context-related factors in the context of POI recommendation. Then the context factors, namely POI-POI, POI-region, POI-time, POI-word, are embedded and modeled in a low-dimensional space in the unified optimization framework.

[4] TMCA: the literature [61] employs the LSTM-based encoder-decoder framework to automatically learn deep spatial-temporal representations for historical check-in activities and integrate multiple contextual factors. Furthermore, this model use multilevel context attention mechanisms to adaptively select relevant check-in activities and contextual factors.

Then we analyze the interactions in the SSSER model, that is, in the final objective function, the corresponding ignored social link interaction, POI-time interaction between users, which are represented by Spatio-SSSER and Social-SSSER respectively.

[1] Spatio-SSSER: this method correspondingly only ignore the social link between users and their friends interaction in SSSER model.

[2] Social-SSSER: this method correspondingly only ignore users' check-in spatiotemporal effect in SSSER model.

D. EXPERIMENTAL PREPROCESSING

1) PARAMETER SETTINGS

Hyper-parameters have a significant effect on the performance recommender system [58], [59], we use a grid search to use the validation collection to find the optimal settings of hyperparameters. These include the latent dimension d from {5, 10, 20, 30, 50, 60, 70}. To control the complexity of the model and avoid overfitting, we use two regularization methods: applying the L2 norm to the model parameters and using the Dropout [60] technique with a 50% reduction rate on the fully-connected layer and the MLP layer. We set the learning rate parameter η to the range of $\{1,10^{-1},10^{-2}, 10^{-3}, 10^{-4}\}$. We select 10^{-3} as the parameter of learning rate. The height h of the horizontal convolution kernel of this paper starts from $\{1, \ldots, I\}$, the first I locations are from $\{1, \ldots, 5\}$, the target number L is from {1,2,3}. The parameters we choose for I and L are 5 and 3, respectively. For each height h, the number of corresponding horizontal filters is from {4, 8, 16, 32, 64}, and the number of vertical filters is from {1, 2, 4, 8, 16}, the step size of the sliding of the convolution kernel is set to 1, the proportion of the Dropout of the fully-connected layer is 0.5, the MLP structure utilizes 4 layers, and the SSSER model is implemented based on Linux PyTorch 0.4.0. We compare the results of each method under the optimal hyperparameter settings of each method.

2) PRE-TRAINING

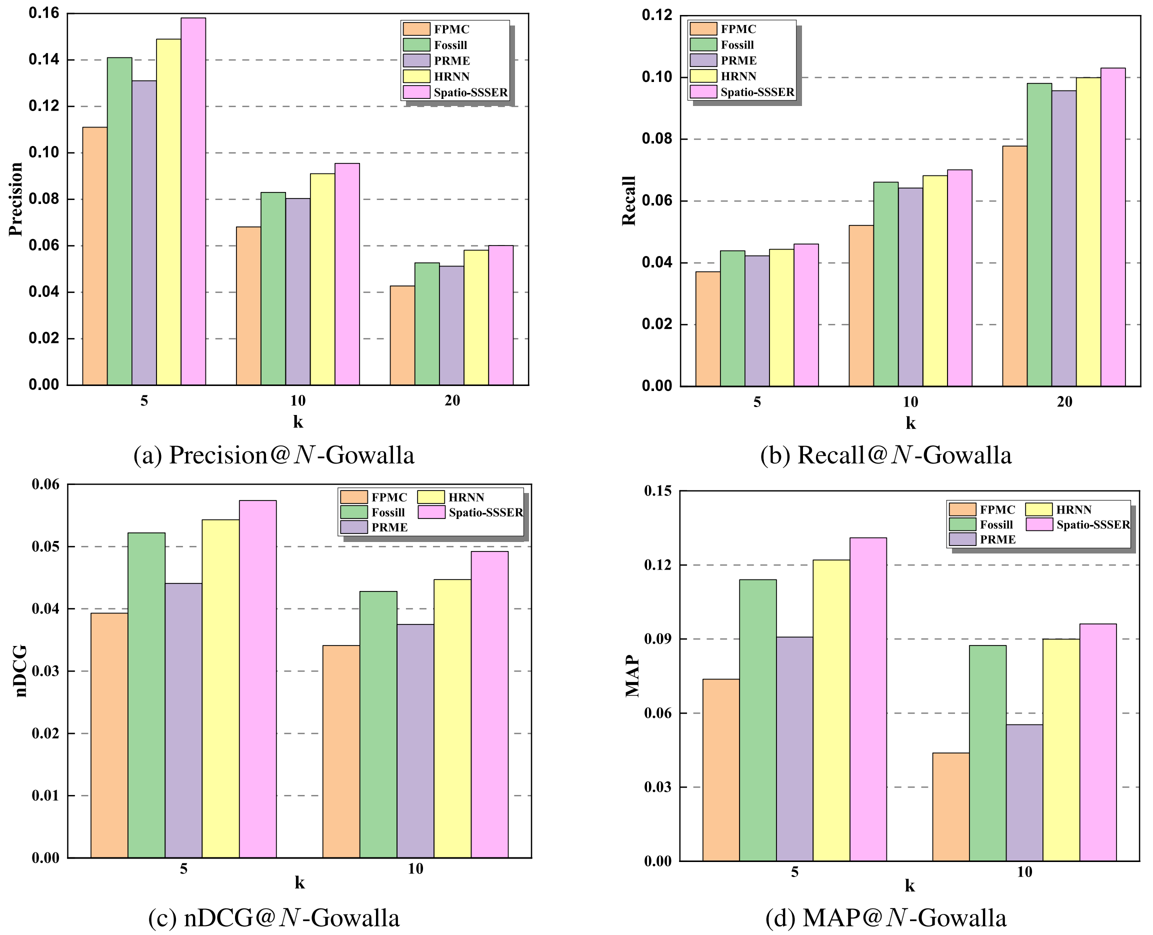
It is well known that neural networks are quite sensitive to initialization [31]. In order to train SSSER better, we use a simplified version to pre-train it. Using the pre-trained model as the initialization, we further train the SSSER model. Note that we use Adam [31] in the pre-training phase, which has fast convergence due to its self-adaptive learning rate strategy. After pre-training, we use SGD optimization, which is a common choice for fine-tuning pre-training models.

E. PERFORMANCE COMPARISON

1) COMPARISON OF SPATIOTEMPORAL SEQUENCE MODELING

The experimental results of the comparison between the proposed model and different state-of-the-art POI recommendation models are shown in Figure 4 to Figure 5. The Spatio-SSSER model achieves better performance than all of the compared models. From the above figures, the following conclusions can be obtained:

1) FPMC: This algorithm is based on the first-order Markov chain model to model and learn users' check-in transformation process. The matrix factorization technique is used to factorize the Markov transformation matrix. To model users' check-in location transformation, FPMC represents each item as two separate vectors, which means the transformation of check-in locations is modeled as the inner product of two independent vectors. However, FPMC cannot model the close relationship between multiple items. FPMC cannot reflect the complex correlation between the above-mentioned check-in locations. At the same time, the prediction of the user's next check-in location should also consider the user's check-in history, rather than merely consider the user's latest check-in record. Therefore, FPMC shows the poorest recommendation performance.
2) Fossil: In order to predict the user's next check-in location, these gorithm models the user's personalized preferences based on two aspects: long-term temporal changes and short-term dynamic temporal changes. Unlike FPMC, Fossil uses a high-order Markov chain model to model the short-term dynamic temporal changes of users. However, this method only uses a simple linear weighting method for the integration of the long-term temporal changes and the short-term dynamic temporal changes. This method is not



(a) Precision@N-Gowalla

(b) Recall@N-Gowalla

(c) nDCG@N-Gowalla

(d) MAP@N-Gowalla

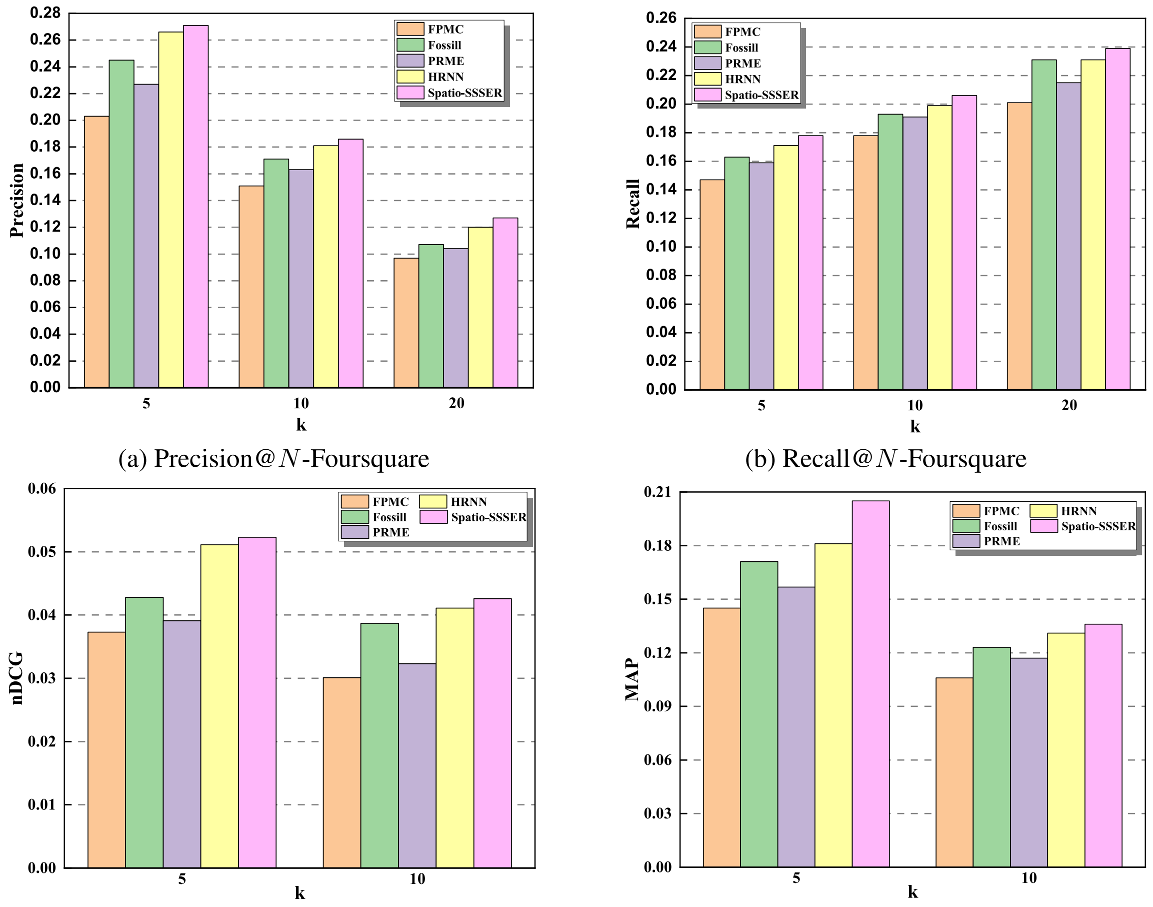
FIGURE 4. Performance comparison on Gowalla dataset.

sufficient to capture the complex interaction between features, and it is also impossible for the high-order Markov chain model to model some real-world scenarios during the user's check-in process, such as the phenomenon of check-in skips. Therefore, this method is ranked as the second poorest in terms of recommendation performance.

3) PRME: This algorithm uses a pair-by-pair sorting scheme to learn parameters and employs metric embedding for POI recommendation tasks. The core of the method is network embedding, the purpose of which is to learn the check-in location as a low-dimensional vector representation of a certain node in the location social network. However, as with most other network-embedding algorithms, this model merely considers those social networks that only contain positive links, that is, only those obvious transformations that are observable in the user's check-in location transformation. The check-in locations with a farther distance in between and those that have not been observed are ignored in this method, while these links, which are considered negative by the network-embedding algorithms, can significantly improve the predictive performance of the positive links and can also enhance the recommendation performance [22]. Therefore, this method is rated as the third in terms of recommendation performance.

4) HRNN: This algorithm, based on a recurrent neural network (RNN), implements the modeling of spatiotemporal sequence patterns. As one of the various deep learning frameworks, RNN exhibits its superiority to the Markov chain model over the prediction process based on sequential patterns. However, when modeling sequential data, RNN simply assumes that the time dependency changes monotonically with the location in the sequence, which means RNN cannot simulate local spatiotemporal sequence patterns very well in some real-world scenarios, especially for personalized check-ins in historical check-in sequences. Therefore, this method ranks second among the above-mentioned POI recommendation methods.

5) Spatio-SSSER: Compared with the state-of-the-art recommendation models, the proposed model achieves a certain degree of improvement in both datasets based



(a) Precision@N-Foursquare

(b) Recall@N-Foursquare

(c)nDCG@N-Foursquare

(d) MAP@N-Foursquare

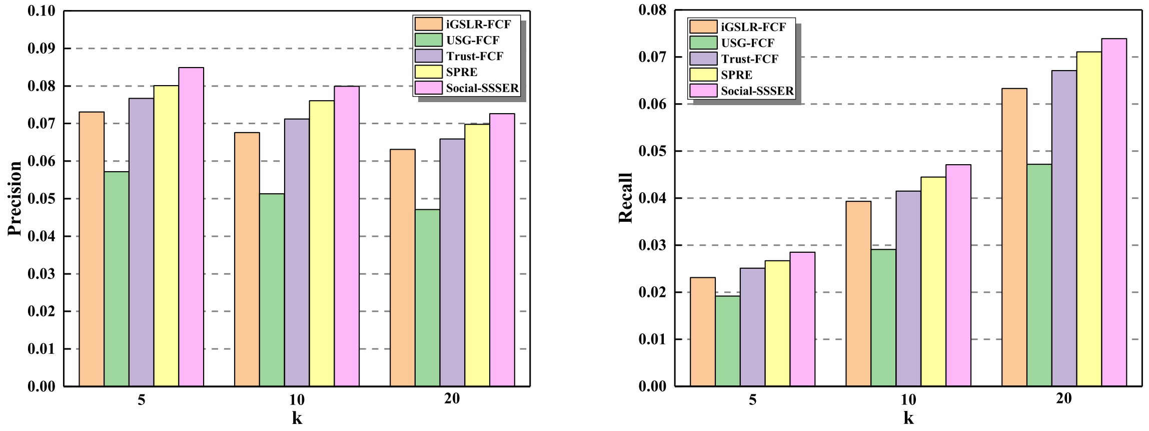
FIGURE 5. Performance comparison on Foursquare dataset.

on all metrics. This verifies the validity of the idea to model sequence information proposed in this paper. The reasons are as follows: (1) This model embeds the most recent item sequences into the "image" of the latent space and uses the sequence pattern as the local features of the "image", thereby adopting a method based on the joint convolution filter for image recognition to capture local features. It is a good complement to the Markov chain model and the RNN model whose shortcoming is that they cannot capture the spatiotemporal sequence information of the user's personalized check-in when modeling sequence information [61]. (2) In this paper, in terms of sequence modeling, we add a multilayer perceptron to learn the interaction function of a user-check-in location-Spatiotemporal pattern so as to better capture the deeper user-location interactions, thus giving the model in this paper a high-level ability of nonlinear modeling.

In addition, among all the compared and evaluated models, as the value of N increases, the precision rate becomes lower, the recall rate higher, and MAP and nDCG also exhibit similar trends with the precision and recall. This is because the more POIs that are recommended to the user, the more POIs the user will find that they may be willing to check-in, but the chances of some recommended POIs to be visited by the user will be reduced. The Spatio-SSSER model performs better in Foursquare than Gowalla, because the average number of user check-in POIs in the Foursquare dataset is higher than that in the Gowalla dataset and the Gowalla dataset is sparser than the Foursquare dataset, as shown in Table 1.

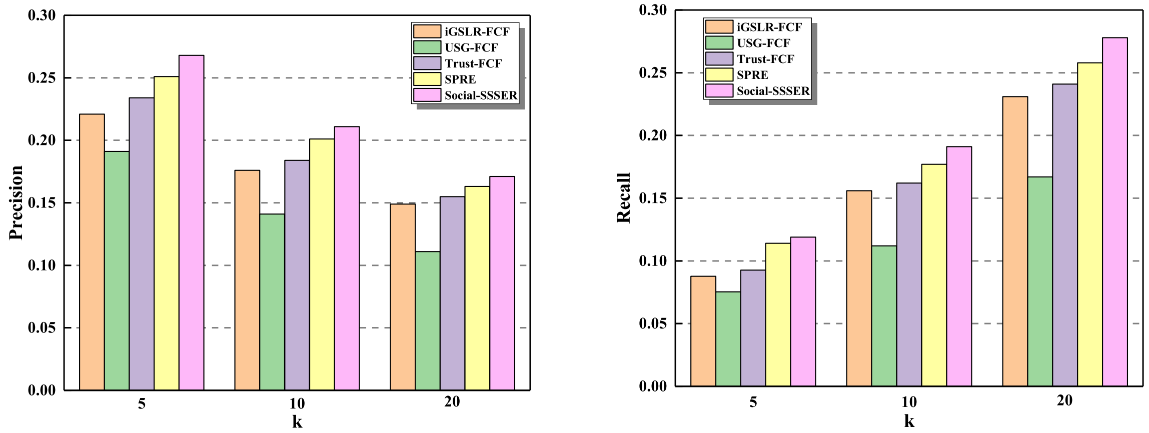
2) COMPARISON OF SOCIAL LINK MODELING

Figures 6 and 7 show the experimental results of a comparison of different social link models. Obviously, iGSLR-FCF is the worst performing social link modeling part of the iGSLR model. One possible explanation is that the model uses the sigmoid function to calculate the effect of this correlation. In the USG-CIFCF model, the interaction between users is reflected by the idea that the similarity of the user's check-in behavior is affected by the similarity of friends and the sigmoid function tends to exhibit the problem of under-fitting in calculating the relevant social link effects, so that the final result is relatively low. The fact that the Trust-FCF model is superior to the USG-CIFCF model indicates that to make



(a) Precision@N-Gowalla (b) Recall@N-Gowalla

FIGURE 6. Recommendation performance comparison on Gowalla dataset.



(a) Precision@N-Foursquare (b) Recall@N-Foursquare

FIGURE 7. Recommendation performance comparison on Foursquare dataset.

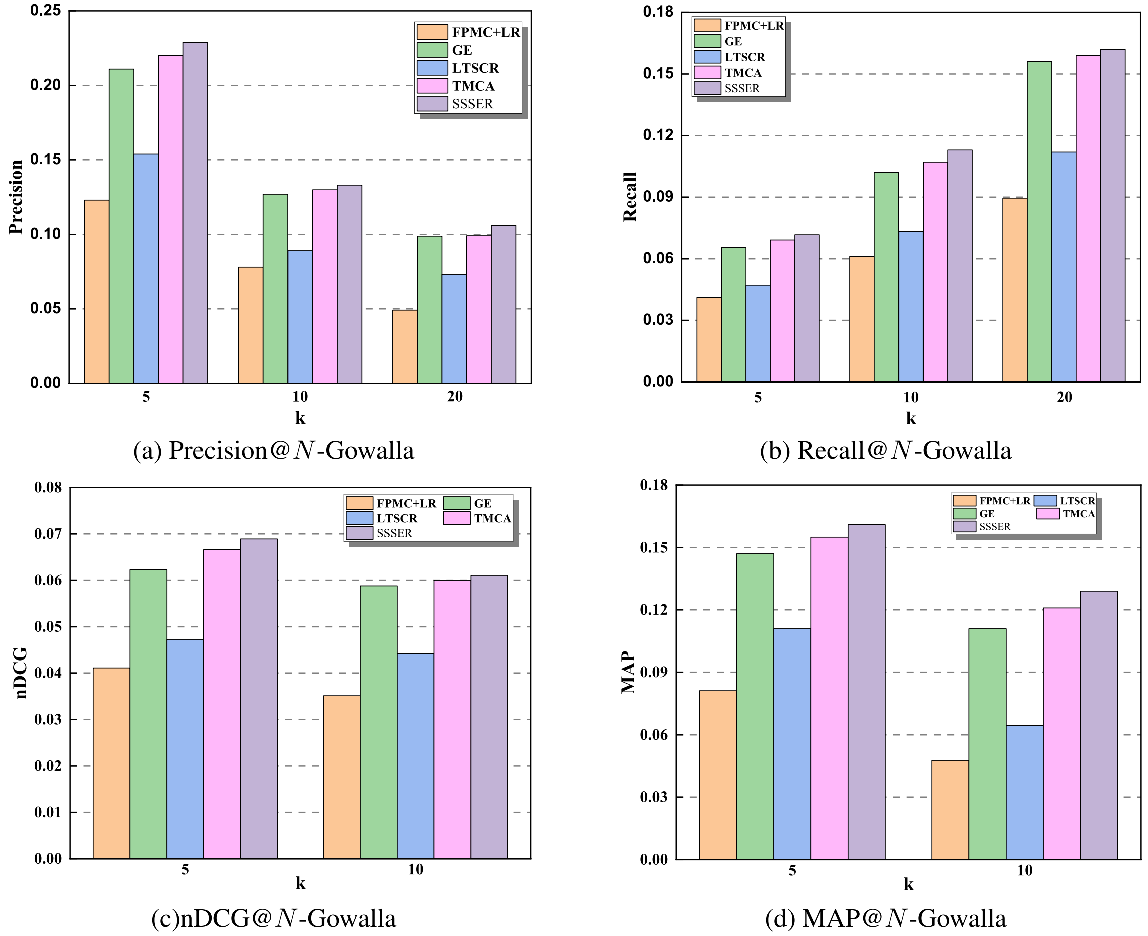
more use of his/her friends in the social link modeling process can help to aggregate more related contextual information of cold-start users based on explicit trust relationship and implicit trust relationship so as to alleviate data sparsity issues and effectively improve recommendation performance. SPRE is essentially a link-based approach to modeling social information, which takes into account all possible friend relationships between the user and other users. Therefore, its performance is superior to that of the Trust-FCF model. However, this approach may affect the final recommendation performance for active users due to the fact that active users have too many indirect friends. At the same time, SPRE only mines positive links in social networks during the modeling process, ignoring negative links, while many studies on social network-based network embedding algorithms have shown that the untrusted or adversarial negative links in social networks provide powerful support for positive links in various predictive tasks.

The proposed method in this paper is superior to other mainstream advanced algorithms based on all metrics in the two datasets. The main reasons are as follows: This paper uses spatial distance to reflect user preferences, and it is different from the general spatial distance. In this paper, the Mahalanobis distance is used instead of the Euclidean distance to measure the spatial distance. All users and items are embedded in a unified space, making the user closer to their favorite items and friends in terms of spatial distance and farther from the items they do not like. The distance between the user and the items is used to generate the recommendation results. Therefore, this method can provide accurate recommendations for both cold-start users and active users. The experimental results also verify the correctness of this method.

In addition, this paper observes that the Social-SSSER model performs better in Foursquare than Gowalla. The reason is that the data concentration in the Foursquare dataset is greater than that in the Gowalla dataset. At the same time, recommending more POI to users helps users to find more POI, which will promote users to be more willing to check in points of interest. Therefore, as the number of POI increases, the precision continues to decrease and the recall continues to rise. Therefore, the results are reasonable, which is also consistent with the conclusions in the literature [55].

3) COMPARISON OF POI RECOMMENDATION ALGORITHMS

1) GE [29]: This algorithm is based on the bipartite graph model to represent the sequence information of users' check-ins, users' check-in information and the check-in time information. The above-mentioned heterogeneous information graphs are then embedded into a shared low-dimensional space to be represented as a low-dimensional space vector. Graph embedding models are often used to deal with sparse data-based data mining tasks and they excel in such tasks. However, the GE model still ignores negative links in social networks just like most graph embedding models, considering only positive links in social networks. With the deepening of the graph embedding model, many researchers have proved that negative links are of great help to the final performance improvement. At the same time, compared with the deep neural network model, the graph embedding model can't perfectly capture the complex feature interaction between social network nodes. Therefore, this model ranks third in all the POI models compared in this paper.
2) FPMC+LR: This algorithm considers the sequence influence of the user's check-in POI by using the latest visited location in the user's check-in sequence to obtain the probability of visiting the next POI, and at the same time the user's candidate recommendation location is regionally constrained. However, in real-world scenarios, it is far from enough to consider only the latest visited location in the check-in sequence. The location prediction of the next POI the user may visit depends on the user's historical visiting information, the user's visiting temporal information and other context-related information. In addition, in the modeling of geographic information, only the constraints of local geographic regions are considered. Therefore, the performance of FPMC+LR based on the evaluation indicators is inferior to several other algorithms, and this algorithm finally exhibits the worst recommendation effect.
3) LTSCR: This algorithm models the location information, the user's social link, and the user's visiting time information based on the extended collaborative retrieval model, and optimizes the loss function using the weighted estimation paired sorting criteria. This approach provides a comprehensive consideration of the user preferences as a result of multidimensional interactions and mutual influences between users, users' friends, check-in time, and check-in POI. However, the algorithm is still based on the linear interpolation method to integrate the above-mentioned several types of context-related information. As analyzed in the literature [6], the general linear weighted integration is not desirable. In practice, the context information influence received by users in different datasets is totally different, so that the distribution of weights can be completely different. At the same time, the social information, time information and geographic information are simply modeled by the similarity calculation equation. Obviously, the similarity calculation cannot reflect the mutual influence of social information, spatiotemporal information, and other information in users' check-in process. Therefore, LTSCR ranks fourth in terms of recommendation performance as shown in Figure 8 and Figure 9.
4) TMCA: this algorithm proposes an encoder-decoder based neural network model to capture the complex spatial and temporal dependencies among historical check-in activities automatically, which leverages the embedding method to incorporate heterogeneous contextual factors to boost recommendation performance. Furthermore, the TMCA model introduces the temporal and multi-level context attention mechanisms to dynamically select the relevant check-ins and discriminative contextual factors. However, limited by the fact that the SPRE model considers only the spatiotemporal sequential influence without social influence in POI recommendation, it only this model ranks second in all the POI models compared in this paper, as illustrated in Figs. 4 and 5 above. That is, the impact of spatiotemporal influence on the final recommendation performance improvement is greater than the social impact, and the integration of multiple context information would greatly enhance the final performance of the POI recommendation. furthermore, this model uses the attention mechanism to select relevant check-in activities and contextual factors. However?the using of attention mechanism is no help for improving the final performance. This phenomenon also conforms to the conclusions of literature [61].
5) SSSER: Compared with the above-mentioned state-of-the-art recommendation models, the proposed model achieves a certain degree of improvement in both datasets according to all metrics. This bears testimony to the validity of the sequence-based modeling idea and the effectiveness of the method to integrate various spatiotemporal features and users' social link influence. The reasons are as follows: (1) In the aspect of sequence modeling, the joint convolution kernel in CNN is used to capture the local features in image recognition. In the meantime, we add a multilayer perceptron to learn the interaction function of user-user's check-in locationcheck-in sequence pattern to capture the deep interaction among them, thus giving the model in this paper a high-level nonlinear modeling ability. The deep user's intent feature extracted by SSSER in a nonlinear way can better capture the user's spatial behavior. (2) We propose to model the social link between users based on metric learning theory to



(a) Precision@N-Gowalla

(b) Recall@N-Gowalla

(c)nDCG@N-Gowalla

(d) MAP@N-Gowalla

FIGURE 8. Performance comparison on Gowalla dataset.

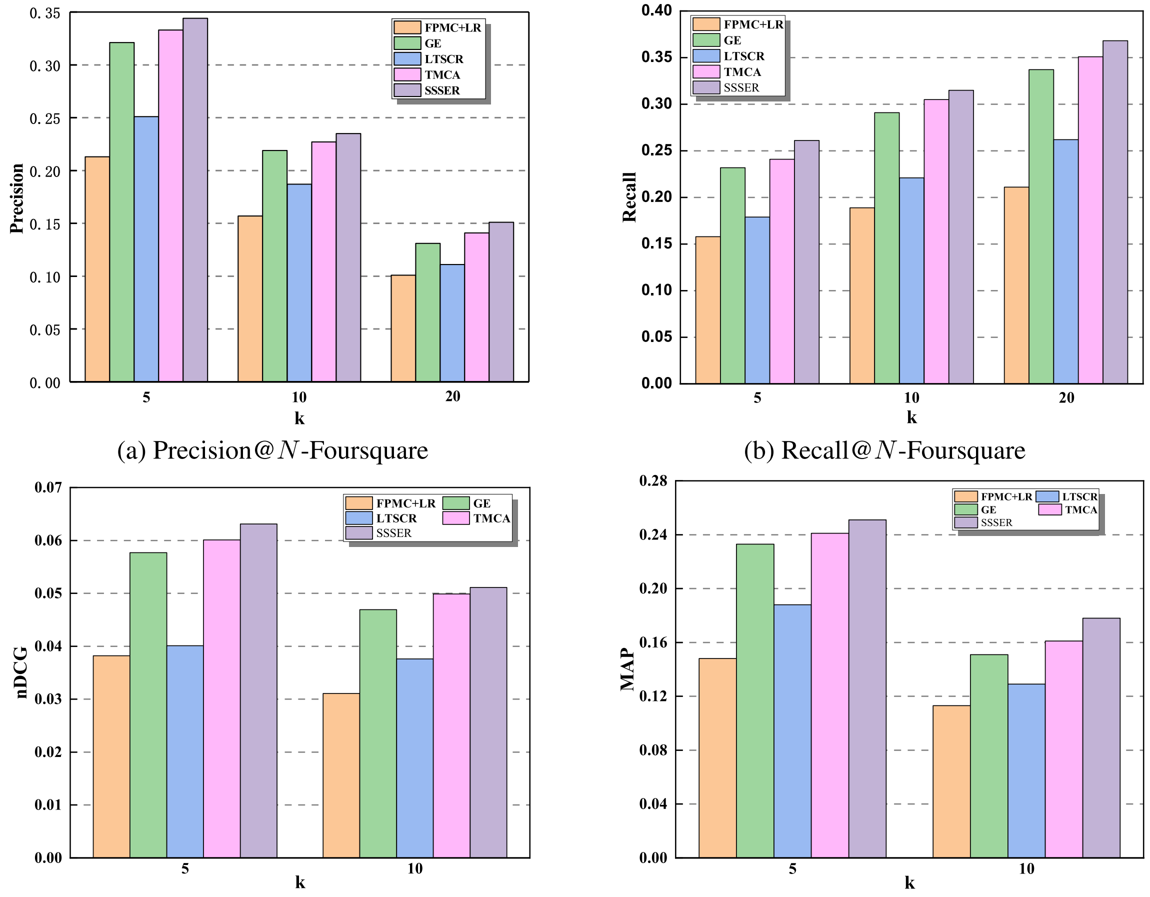
improve the performance of the final recommendation. (3) The BPR strategy is applied to optimize loss function and it fits the partial order relationship at POI pairs. (4) We integrate the above-mentioned spatiotemporal sequence information and social information and generate the fine-grained modeling of the interaction of the context information in the above-mentioned users' check-in process, which is effective for the task of the next POI recommendation.

4) ANALYSIS OF COMPONENTS' INFLUENCE

In this section, we discuss how the two related components, namely temporal sequence information, and social link, influence the final results of the proposed model and also demonstrates the contribution of these two components to the recommendation performance improvement. As shown in the above figures, we observe that the recommendation performance of the SSSER model has been improved on two datasets according to the evaluation indicators. This paper presents the following conclusions: First, SSSER is constantly superior to its components based on two evaluation indicators in the two datasets, which indicates that the use of social links can effectively improve the final recommendation performance and also verifies the effectiveness of the proposed model. In addition, as shown in the above figures, the spatiotemporal sequence effects bear on users' check-in behavior. The temporal information exerts a greater impact on the final recommendation performance when compared with social links, but the two components themselves compete with each other. One feasible explanation is that in the LBSN, spatiotemporal location interaction is possible only when users check-in at the geographical location during certain periods. Finally, the integration of relevant contextual information contributes to the improvement of recommendation performance. This is evident from the fact that the SSSER model outperforms the two components in terms of precision and recall, which is consistent with the observations in the literature [61]. This is because, in real-world activities, people more or less are affected by spatiotemporal sequences and social links, etc., and only one type of contextual information cannot correctly and comprehensively model the check-in behavior of users.

5) THE IMPACT OF DIMENSION

The change of the latent factor dimension will have a significant impact on the robustness of the proposed model. The size of the dimension also affects the number of



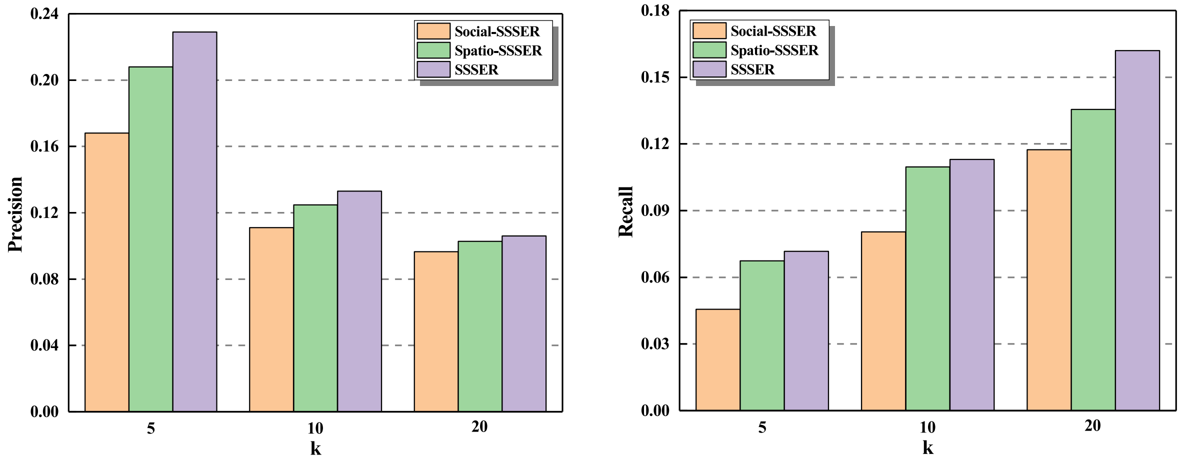
(a) Precision@N-Foursquare

(b) Recall@N-Foursquare

(c)nDCG@N-Foursquare

(d) MAP@N-Foursquare

FIGURE 9. **Performance comparison on Foursquare dataset.**

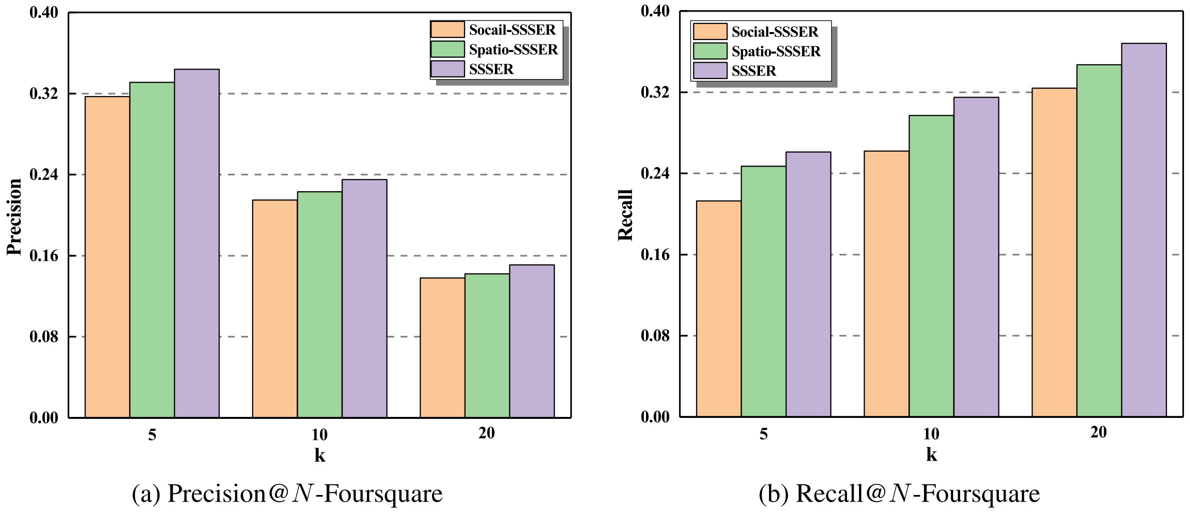


(a) Precision@N-Gowalla

(b) Recall@N-Gowalla

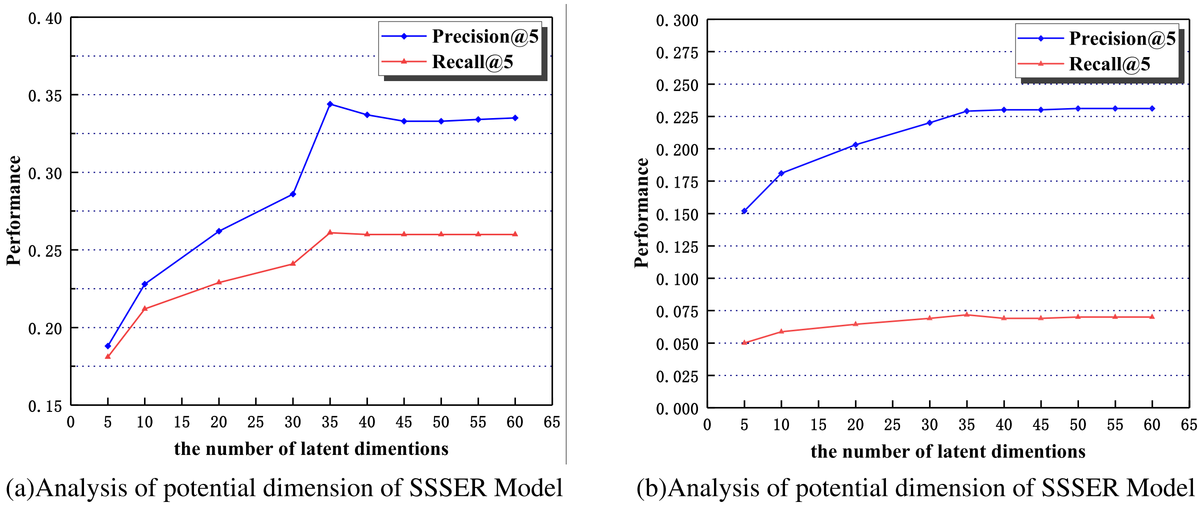
FIGURE 10. **Recommendation performance comparison on Gowalla dataset.**

iterations in the optimization process of the recommendation model, the running time of the algorithm, and the memory used. However, the larger size of the latent dimension does not necessarily lead to better model performance. The recommendation model achieves optimal performance only when the dimension selection is based on a balanced



(a) Precision@N-Foursquare (b) Recall@N-Foursquare

FIGURE 11. Recommendation performance comparison on Foursquare dataset.



(a)Analysis of potential dimension of SSSER Model (b)Analysis of potential dimension of SSSER Model

FIGURE 12. Analysis of the impact of SSSER model on dimensionality based on the two datasets.

consideration between recommendation performance and calculation cost.

Figures 12 clearly show that this paper uses the two evaluation indicators, Precision@5 and Rrecall@5, to verify the performance impact of different latent factor dimensions on the proposed algorithm based on two datasets. As shown in Figures 9, as the dimension increases, the performance of the recommendation model improves. When the dimension is larger than 35, for both datasets, Precision@5 and Recall@5 become stable and do not fluctuate because of further changes in the dimension. One possible explanation is that an oversized dimension can lead to overfitting of the recommendation model. Therefore, the experiments in this paper select a dimension of 35, so that the balance performance and calculation cost are reasonable and effective, and the recommendation model proposed in this paper obtains stable performance.

VI. CONCLUSION

In this paper, we develop a hybrid recommendation model called SSSER to integrate convolutional neural networks and metric learning for POI recommendation. On the one hand, we use CNN to model the spatiotemporal association between the check-in POIs and the user. On the other hand, we apply users' social link and user/user potential relationships to alleviate the data scarcity problem that is common in the POI recommendation process. Our experiments demonstrate the effectiveness and efficiency of the proposed method.

For future work, we hope to integrate richer contextual information into the model proposed in this paper, which may generate additional performance improvements in SSSER. In addition, recent advances in deep learning, e.g., attention mechanism, dilated convolutional neural network, and generative adversarial network, have shown great potential in the fields of natural language processing and computer vision. Hence, applying the above deep learning techniques to recommender systems would be an interesting direction.

REFERENCES

[1] D. Lian, K. Zheng, Y. Ge, L. Cao, E. Chen, and X. Xie, "GeoMF++: Scalable location recommendation via joint geographical modeling and matrix factorization," *ACM Trans. J. Inf. Syst.*, vol. 36, no. 3, Apr. 2018, Art. no. 33.

[2] 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. (IJCAI)*, Jul. 2018, pp. 3301–3307.

[3] G. Yang, Y. Cai, and C. K. Reddy, "Recurrent spatio-temporal point process for check-in time prediction," in *Proc. ACM 27th Int. Conf. Inf. Knowl. Manage.*, Oct. 2018, pp. 2203–2211.

[4] 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. the 30th AAAI. Conf. Artif. Intell.*, Feb. 2016, pp. 315–321.

[5] J. He, X. Li, L. Liao, D. Song, and W. K. Cheung, "Inferring a personalized next point-of-interest recommendation model with latent behavior patterns," in *Proc. 30th AAAI. Conf. Artif. Intell.*, Feb. 2016, pp. 137–143.

[6] W. Zhang and J. Wang, "Location and time aware social collaborative retrieval for new successive point-of-interest recommendation," in *Proc. ACM 24th Int. Conf. Inf. Knowl. Manage.*, Oct. 2015, pp. 1221–1230.

[7] J.-D. Zhang and C.-Y. Chow, "Spatiotemporal sequential influence modeling for location recommendations: A gravity-based approach," *ACM Trans. Intell. Syst. Technol.*, vol. 7, no. 1, 2015, Art. no. 11.

[8] R. He and J. McAuley, "Fusing similarity models with Markov chains for sparse sequential recommendation," in *Proc. IEEE 16th Int. Conf. Data Mining (ICDM)*, Dec. 2016, pp. 191–200.

[9] Y.-T. Zheng, Z.-J. Zha, and T.-S. Chua, "Mining travel patterns from geotagged photos," *ACM Trans. Intell. Syst. Technol.*, vol. 3, no. 3, pp. 56:1–56:18. May 2012.

[10] Z. Chen, H. T. Shen, and X. Zhou, "Discovering popular routes from trajectories," in *Proc. IEEE 27th Int. Conf. Data Eng. (ICDE)*, Apr. 2011, pp. 900–911.

[11] A.-J. Cheng, Y.-Y. Chen, Y.-T. Huang, W. H. Hsu, and H.-Y. M. Liao, "Personalized travel recommendation by mining people attributes from community-contributed photos," in *Proc. ACM 19th Int. Conf. Multimedia*, Nov./Dec. 2011, pp. 83–92.

[12] L. Huang, B. Jiang, and S. Lv, "A review of research on recommendation systems based on deep learning," *Proc. Chin. J. Comput.*, vol. 41, no. 7, pp. 191–219, 2018.

[13] Q. Liu, S. Wu, L. Wang, and T. Tan, "Predicting the next location: A recurrent model with spatial and temporal contexts," in *Proc. 30th AAAI Conf. Artif. Intell.*, Feb. 2016, pp. 194–200.

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

[15] E. H.-C. Lu, C.-Y. Chen, and V. S. Tseng, "Personalized trip recommendation with multiple constraints by mining user check-in behaviors," in *Proc. ACM 20th Int. Conf. Adv. Geographic Inf. Syst.*, Nov. 2012, pp. 209–218.

[16] D. Lian, X. Xie, V. W. Zheng, N. J. Yuan, F. Zhang, and E. Chen, "CEPR: A collaborative exploration and periodically returning model for location prediction," *ACM Trans. Intell. Syst. Technol.*, vol. 6, no. 1, 2015, Art. no. 8.

[17] H. Wang, M. Terrovitis, and N. Mamoulis, "Location recommendation in location-based social networks using user check-in data," in *Proc. ACM 21st SIGSPATIAL Int. Conf. Adv. Geographic Inf. Syst.*, Nov. 2013, pp. 374–383.

[18] L. Zheng, V. Noroozi, and P. S. Yu, "Joint deep modeling of users and items using reviews for recommendation," in *Proc. ACM 10th ACM Int. Conf. Web Search Data Mining*, Feb. 2017, pp. 425–434.

[19] Y. Shan, T. R. Hoens, J. Jiao, H. Wang, D. Yu, and J. Mao, "Deep crossing: Web-scale modeling without manually crafted combinatorial features," in *Proc. ACM 22nd SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2016, pp. 255–262.

[20] J. Wang, Z. Deng, K.-S. Choi, Y. Jiang, X. Luo, F.-L. Chung, and S. Wang, "Distance metric learning for soft subspace clustering in composite kernel space," *Pattern Recognit.*, vol. 52, pp. 113–134, Apr. 2016.

[21] S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme, "BPR: Bayesian personalized ranking from implicit feedback," in *Proc. 25th Conf. Uncertainty Artif. Intell. (UAI)*. Arlington, VA, USA: AUAI Press, 2009, pp. 452–461.

[22] 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.

[23] H. Yin, Z. Hu, X. Zhou, H. Wang, K. Zheng, Q. V. H. Nguyen, and S. Sadiq, "Discovering interpretable geo-social communities for user behavior prediction," in *Proc. IEEE 32nd Int. Conf. Data Eng. (ICDE)*, May 2016, pp. 942–953.

[24] W.-Y. Zhu, W.-C. Peng, L.-J. Chen, K. Zheng, and X. Zhou, "Modeling user mobility for location promotion in location-based social networks," in *Proc. ACM 21th Int. Conf. Knowl. Discovery Data Mining*, Aug. 2015, pp. 1573–1582.

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

[26] C. Cheng, H. Yang, M. R. Lyu, and I. King, "Where you like to go next: Successive point-of-interest recommendation," in *Proc. 23rd Int. Conf. Artif. Intell.*, Aug. 2013, pp. 2605–2611.

[27] H. Yin, B. Cui, X. Zhou, W. Wang, Z. Huang, and S. Sadiq, "Joint modeling of user check-in behaviors for real-time point-of-interest recommendation," *ACM Trans. Inf. Syst.*, vol. 35, no. 2, Dec. 2016, Art. no. 11.

[28] Y. Liu, C. Liu, B. Liu, M. Qu, and H. Xiong, "Unified point-of-interest recommendation with temporal interval assessment," in *Proc. ACM 22nd Int. Conf. Knowl. Discovery Data Mining*, Aug. 2016, pp. 1015–1024.

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

[30] S. Zhang, L. Yao, A. Sun, and Y. Tay, "Deep learning based recommender system: A survey and new perspectives," *ACM Comput. Surv.*, vol. 52, no. 1, Feb. 2017, Art. no. 5.

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

[32] C. Chen, X. Meng, Z. Xu, and T. Lukasiewicz, "Location-aware personalized news recommendation with deep semantic analysis," *IEEE Access*, vol. 5, pp. 1624–1638, 2017.

[33] S. Okura, Y. Tagami, S. Ono, and A. Tajima, "Embedding-based news recommendation for millions of users," in *Proc. ACM 23rd Int. Conf. Knowl. Discovery Data Mining*, Aug. 2017, pp. 1933–1942.

[34] Y. Song, A. Elkahky, and X. He, "Multi-rate deep learning for temporal recommendation," in *Proc. ACM 39th Int. Conf. Res. Develop. Inf. Retr.*, Jul. 2016, pp. 909–912.

[35] C. Wu, J. Wang, J. Liu, and W. Liu, "Recurrent neural network based recommendation for time heterogeneous feedback," *Knowl.-Based Syst.*, vol. 109, pp. 90–103, Oct. 2016.

[36] X. Wu, B. Shi, Y. Dong, C. Huang, and N. V. Chawla, "Neural tensor factorization for temporal interaction learning," in *Proc. ACM 12th Int. Conf. Web Search Data Mining*, Feb. 2019, pp. 537–545.

[37] R. Zhao, D. Wang, R. Yan, K. Mao, F. Shen, and J. Wang, "Machine health monitoring using local feature-based gated recurrent unit networks," *IEEE Trans. Ind. Electron.*, vol. 65, no. 2, pp. 1539–1548, Feb. 2018.

[38] Q. Liu, S. Wu, and L. Wang, "Multi-behavioral sequential prediction with recurrent log-bilinear model," *IEEE Trans. Knowl. Data Eng.*, vol. 29, no. 6, pp. 1254–1267, Jun. 2017.

[39] C. Wu, A. Ahmed, A. Beutel, A. J. Smola, and H. Jing, "Recurrent recommender networks," in *Proc. ACM 10th Int. Conf. Web Search Data Mining*, Feb. 2017, pp. 495–503.

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

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

[42] J. Ye, Z. Zhao, and H. Liu, "Adaptive distance metric learning for clustering," in *Proc. 17th IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2007, pp. 1–7.

[43] P. Wu, S. C. H. Hoi, P. Zhao, C. Miao, and Z.-Y. Liu, "Online multi-modal distance metric learning with application to image retrieval," *IEEE Trans. Knowl. Data Eng.*, vol. 28, no. 2, pp. 454–467, Feb. 2016.

[44] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei, "Large-scale video classification with convolutional neural networks," in *Proc. IEEE 18th Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 1725–1732.

[45] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. 26th Conf. Neural Inf. Process. Syst.*, Dec. 2012, pp. 1097–1105.

[46] N. M. Villegas, C. Sánchez, J. Díaz-Cely, and G. Tamura, "Characterizing context-aware recommender systems: A systematic literature review," *Knowl.-Based Syst.*, vol. 140, pp. 173–200, Jan. 2018.

[47] K. Verbert, N. Manouselis, X. Ochoa, M. Wolpers, H. Drachsler, I. Bosnic, and E. Duval, "Context-aware recommender systems for learning: A survey and future challenges," *IEEE Trans. Learn. Technol.*, vol. 5, no. 4, pp. 318–335, Oct./Dec. 2012.

[48] C. Hsieh, L. Yang, Y. Cui, T.-Y. Lin, S. Belongie, and D. Estrin, "Collaborative metric learning," in *Proc. 26th Int. Conf. World Wide Web*, Apr. 2017, pp. 193–201.

[49] J. Weston, S. Bengio, and N. Usunier, "Large scale image annotation: Learning to rank with joint word-image embeddings," *Mach. Learn.*, vol. 81, no. 1, pp. 21–35, 2010.

[50] H. Gao, J. Tang, and H. Liu, "gSCorr: Modeling geo-social correlations for new check-ins on location-based social networks," in *Proc. ACM 21st Int. Conf. Inf. Knowl. Manage.*, Oct./Nov. 2012, pp. 1582–1586.

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

[52] S. Rendle, C. Freudenthaler, and L. Schmidt-Thieme, "Factorizing personalized Markov chains for next-basket recommendation," in *Proc. ACM 19th Int. Conf. World Wide Web*, Apr. 2010, pp. 811–820.

[53] M. Quadrana, A. Karatzoglou, B. Hidasi, and P. Cremonesi, "Personalizing session-based recommendations with hierarchical recurrent neural networks," in *Proc. 11th ACM Conf. Recommender Syst.*, Aug. 2017, pp. 130–137.

[54] S. Feng, X. Li, Y. Zeng, G. Cong, Y. M. Chee, and Q. Yuan, "Personalized ranking metric embedding for next new POI recommendation," in *Proc. 24th Int. Conf. Artif. Intell.*, Jul. 2015, pp. 2069–2075.

[55] J.-D. Zhang and C.-Y. Chow, "iGSLR: Personalized geo-social location recommendation: A kernel density estimation approach," in *Proc. ACM 21st SIGSPATIAL Int. Conf. Adv. Geographic Inf. Syst.*, Nov. 2013, pp. 334–343.

[56] W. Yao, J. He, G. Huang, and Y. Zhang, "Modeling dual role preferences for trust-aware recommendation," in *Proc. ACM 37th Int. SIGIR Conf. Res. Develop. Inf. Retr.*, Jul. 2014, pp. 975–978.

[57] Y. Long, P. Zhao, V. S. Sheng, G. Liu, J. Xu, J. Wu, and Z. Cui, "Social personalized ranking embedding for next POI recommendation," in *Proc. 18th Int. Conf. Web Inf. Syst. Eng.* Cham, Switzerland: Springer, 2017, pp. 91–105.

[58] S. Rendle, "Factorization machines with libFM," *ACM Trans. J. Intell. Syst. Technol. (TIST)*, vol. 3, no. 3, May 2012, Art. no. 57.

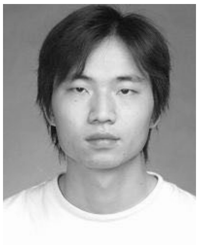
[59] H. Liu, X. He, F. Feng, L. Nie, R. Liu, and H. Zhang, "Discrete factorization machines for fast feature-based recommendation," in *Proc. 27th Int. Conf. Artif. Intell. Org. (IJCAI)*, Jul. 2018, pp. 3449–3455.

[60] C. Ma, P. Kang, B. Wu, Q. Wang, and X. Liu, "Gated attentive-autoencoder for content-aware recommendation," in *Proc. ACM 12th Int. Conf. Web Search Data Mining*, Feb. 2019, pp. 519–527.

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

[62] P. Sun, L. Wu, and M. Wang, "Attentive recurrent social recommendation," in *Proc. ACM 41st Int. SIGIR Conf. Res. Develop. Inf. Retr.*, Jul. 2018, pp. 185–194.

YANGYANG XU received the M.S. degree from Wuhan University, Wuhan, China, in 2019. He is currently with the 2012 Laboratory, Huawei Technologies Company, Ltd., Shenzhen, China. His current research interests include data mining and intelligent recommendation.



XUEFEI LI received the Ph.D. degree from Wuhan University, in 2007, Wuhan, China. He is currently an Associate Professor with the School of Computer Science, Wuhan University. His current research interests include data mining and multimedia technology.

JING LI received the Ph.D. degree from Wuhan University, Wuhan, China, in 2006. He is currently a Professor with the School of Computer Science, Wuhan University. His research interests include multimedia technology and data mining.

CHUNZHI WANG received the Ph.D. degree from the Wuhan University of Technology, Wuhan, China, in 2013. She is currently a Professor with the School of Computer Science, Hubei University of Technology, Wuhan. Her research interests include computer networks and data processing.

RONG GAO received the Ph.D. degree from Wuhan University, Wuhan, China, in 2018. He is currently an Assistant Professor with the School of Computer Science, Hubei University of Technology, Wuhan. His research interests include data mining and intelligent recommendation.

YONGHONG YU received the Ph.D. degree from Nanjing University, Nanjing, China, in 2016. He is currently an Associate Professor with the College of Tongda, Nanjing University of Posts and Telecommunications, Nanjing. His current research interests include data mining and machine learning.