Research Article

Deep CNN-Assisted Personalized Recommendation over Big Data for Mobile Wireless Networks

Yu Zheng ¹, Xiaolong Xu ¹, and Lianyong Qi ²

¹*School of Computer & Software, Jiangsu Engineering Center of Network Monitoring,
Nanjing University of Information Science & Technology, Nanjing 210044, Jiangsu, China*

²*School of Information Science and Engineering, Qufu Normal University, Qufu 273165, Shandong, China*

Correspondence should be addressed to Lianyong Qi; lianyongqi@gmail.com

Received 30 October 2018; Accepted 20 February 2019; Published 24 April 2019

Guest Editor: Huaming Wu

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

At present, to improve the accuracy and performance for personalized recommendation in mobile wireless networks, deep learning has been widely concerned and employed with social and mobile trajectory big data. However, it is still challenging to implement increasingly complex personalized recommendation applications over big data. In view of this challenge, a hybrid recommendation framework, i.e., deep CNN-assisted personalized recommendation, named DCAPR, is proposed for mobile users. Technically, DCAPR integrates multisource heterogeneous data through convolutional neural network, as well as inputs various features, including image features, text semantic features, and mobile social user trajectories, to construct a deep prediction model. Specifically, we acquire the location information and moving trajectory sequence in the mobile wireless network first. Then, the similarity of users is calculated according to the sequence of moving trajectories to pick the neighboring users. Furthermore, we recommend the potential visiting locations for mobile users through the deep learning CNN network with the social and mobile trajectory big data. Finally, a real-word large-scale dataset, collected from Gowalla, is leveraged to verify the accuracy and effectiveness of our proposed DCAPR model.

1. Introduction

At present, mobile wireless networks are moving towards the interconnection of all things, intelligent interconnection, and social production organizations to accelerate customization, decentralization, and service transformation direction. Global mobile wireless network users reach 3 billion 70 million, and smartphone penetration rate reaches 56%. The global population of 7.2 billion shows that the global mobile wireless network market as a whole has a demographic dividend and is considerable [1]. The popularity of mobile devices will take up a large number of users' time, time fragments, which have been verified in China, Europe, the United States, and other developed countries. Whether on the subway or on the bus, or even in the bathroom, mobile devices play the role of information acquisition tools, always accompanied by users. Therefore, mobile devices have become a major gateway to recommending information on mobile wireless networks.

With the explosive growth of mobile traffic data and unprecedented demand for computing power, users' behavior in the mobile network is no longer limited to accessing information, but more interacting with other users on the social network. Mobile social networking as an open public information exchange and business service platform has quickly entered people's daily work and life [2]. Famous social networks include Facebook, LinkedIn, Twitter, Sina microblog, Renren network, Tencent QQ, and WeChat. In social networking sites, users are no longer individual individuals, but have intricate relationships with many people on the network. The most important resource in a social network is the relationship data between the user and the user. Using technologies such as GPS positioning, the geographic location and movement trajectory of the mobile network user can be obtained very accurately. Due to the rapid development of GPS technology in the global positioning system, it is convenient to obtain the current location information of the user. The trajectory formed during the movement

of people during activities can also be saved by collecting GPS data. In a mobile social network, the user is not just an individual, and the behavior of the user in the social network is affected by these user relationships. By mining the original GPS data to find out the information between users, users can not only find users who are similar to their own activities to establish user social networks, but also predict other users' destinations through similar users, thus giving users some activity recommendations, such as restaurant, tourist destination, and gym recommendation. Therefore, the research and application of the recommendation system in such mobile social networking sites should consider the interaction of user social relationships.

The purpose of recommender system is to help consumers focus on products they care about and avoid over selection. According to a large number of experimental data it is found that most people in the daily work and decision-making always rely on other people's suggestions. Recommender systems are particularly important for those who lack sufficient personal experience and ability. When they cannot find the most needed information from a large amount of information data, personalized recommendation system will help them filter information. According to the user's personal preferences and requirements, different users or user groups receive different recommendations. Therefore, personalization is a basic strategy to promote user experience.

With the deepening of the era of big data, the application of deep learning in recommendation systems has been paid more and more attention by academics and industry. In a mobile wireless network, the relationship between users is different. The users who establish the relationship may be relatives, classmates, colleagues, friends in the real world, or virtual friends in the network, such as members of social networks with common interests. This information constitutes a huge dataset. So far, the combination of deep learning and social network-based recommendation systems has triggered a series of research results, and the recommendation of location-based social network sequence modeling based on deep learning is in the ascendant. Deep learning has shown outstanding performance in many research fields such as computer vision, natural language processing, and so on, which has aroused great interest. At present, how to securely implement multisource data efficient recommendation recommendation service based on big data environment has also attracted the attention of many scholars [3–8]. Obviously, the field of deep learning in recommendation system is booming.

In this paper, we deeply studied the personalized recommendation of social network based on the trajectory data of mobile wireless network users. This paper proposes a new location-based social network recommendation framework, which combines the user's location trajectory sequence, user-shared images, and text information together to recommend the location of mobile social network users more accurately. The main contributions of this paper are as follows:

- (i) Firstly, the location information of mobile network users in different time periods is analyzed, then the user trajectory in a certain time period is constructed, then the space-time similarity calculation method of

the location trajectory of the mobile network user is selected, and finally the location-based neighbor is picked out for the user.

- (ii) Study the problem of how to label user according to the extract features of pictures and text. We use the dual CNN network to extract the characteristics and semantics of the images posted by the users to judge the user's interest points, so as to find the neighbors in the mobile social network.
- (iii) Conduct experiments on a dataset collected from Gowalla to demonstrate the effectiveness of the proposed framework.

The rest of this paper is organized as follows: Section 2 describes the progress of the deep learning algorithm and the recommended algorithm. Section 3 specifies the preparations that the algorithm model needs to perform. Section 4 details the three main components of the proposed model. In Section 5, we conducted extensive experiments and case studies. Finally, in Section 6, we summarize and look forward to the related work.

2. Related Work

Recently, a lot of research has been done in the field of deep learning recommendation. Recommender system estimates user preferences for projects and recommends items that users may like [9–11]. Recommendation models are usually classified into three types: content based, collaborative filtering, and hybrid recommendation system [12]. Content based recommendation is mainly based on the comparison of project and user assistance information. Collaborative filtering provides recommendations by learning from user project history interactions, whether explicit (for example, the user's previous rating) or implicit feedback (for example, browsing history). Various auxiliary information (such as text, image, and video) can be considered. A hybrid model is a recommendation system that integrates two or more recommendation strategies [12].

Zheng et al. [13] based on deep collaborative neural networks use comment information to jointly learn project attributes and user behavior. The model uses the shared layer to couple the project characteristics with user behavior. The model was compared with five lines based on matrix decomposition, probability matrix decomposition, LDA, cooperative topic regression, hidden factor, and cooperative in-depth learning using three real-world datasets: yelp review, Amazon review, and Beer review. The model outperforms all baselines on all benchmark datasets [14].

In terms of collaborative filtering based e-commerce recommendation system, Li et al. [15] first proposed the framework of combining in-depth learning features with CF models (such as matrix decomposition) in 2015. Kriegeskorte et al. [16] developed a probabilistic rating autoencoder for unsupervised feature learning. The autoencoder generates user profiles based on user item rating data, effectively enhancing collaborative filtering methods. Hidasi first used RNN to recommend data based on short sessions, rather

FIGURE 1: User check-in distribution map from the Gowalla dataset.

than long historical data in 2016 [17]. And in 2018, Hidasi et al. used item features such as images and text to further enhance RNN based session recommendation [18]. Jannach et al. showed that the combination of RNN and KNN can effectively improve the recommendation accuracy of e-commerce applications [19]. Chatzis et al. [20] used Bayesian statistical variational reasoning model to improve recurrent neural network model based on session prediction. Bogina et al. [21] proposed a RNN model. Merge dwell time (the time users check for specific items) to improve the accuracy of session-based recommendation in e-commerce datasets, Yoochoose. Ebesu et al. [22] showed that, to solve the cold start problem of cooperative recommendation system, a neural network semantic personalized sorting method based on deep neural network and pairwise learning is proposed.

Based on hybrid recommendation system, Kim et al. [23] studied a model based on convolutional neural network, which combines the metadata information of users or projects to achieve the purpose of improving the matrix decomposition method. Wu et al. [24] proposed a denoising collaborative filtering method based on automatic encoder. This model serves as a general framework for all collaborative filtering methods, but with more flexible adjustments. The model performs better on the MovieLens, Yelp, and Netflix datasets than the baseline, such as ItemPop, ItemCF, Matrix Decomposition, BPR, and FISIM.

Since convolution neural network has powerful functions in image, text, audio, video, and other types of multisource feature representation learning, most CNNs-based recommendation models use CNNs for feature extraction. In [25], Wang et al. studied the problem of using visual content to enhance POI recommendation. In particular, [26] proposed a new framework, Visual Content Enhanced POI Recommendation (VPOI), which combines the visual content of POI recommendation and validates the effectiveness of the

proposed framework with real-world datasets. In [27], Chu et al. used pretrained deep network VGG-f from MatConvNet toolbox to extract CNN features and used support vector machine (SVM) to classify images into four categories: food, beverage, indoor, and outdoor. Different types of images may vary in restaurant recommendation. By combining content based approach and collaborative filtering method, a hybrid restaurant recommendation system is constructed.

3. Preliminaries

Nowadays, many people use mobile social networks to post, praise, share, comment, browse news, and organize offline activities through social networks, so that people with the same hobbies can gather together. If the user's preferences are learned from these behaviors, and the user is accurately portrayed, then personalized content recommendation can be made according to personal preferences, habits, and other information. For example, if we open the news class app, because there is personalized content, everyone sees that the news home page is different. In this chapter, we analyze the composition of recommendation data from three perspectives. Firstly, users' potential hobby information is obtained by extracting the pictures of users in social networks. Secondly, we judge the places that users often visit according to the users' moving track. Finally, we determine the users' interest points by the posted picture and forwarding information in social networks.

3.1. Trajectory Marking Scheme. Figure 1 depicts the distribution of users around the world, with data sources coming from the Gowalla dataset. This dataset contains 196591 nodes, 950327 edges, and 6,442,890 check-ins.

We classify social network data into three categories: one-way social network data, two-way social network data,

FIGURE 2: Social network attention. We selected the 10 users as the representatives from the Gowalla_edges dataset. The ordinate represents the ID number of the person concerned and the abscissa coordinates the ID number of fans. As can be seen from the figure, the user whose ID number is 20 received more than 176 people's attention, the highest degree of attention; and the user whose number is 10 only received the attention of 22 people. Figure 2(b) shows the trajectory of the 3 users (their user Id are 15, 16, and 20, respectively). The abscissa represents latitude, and the ordinate represents longitude.

and community-based social network data. In social network data, the amount of attention and the amount of fans of each user can be regarded as a complex directed graph. Each node represents a user, and the total number of users that each user pays attention to is recorded as the output degree of the node, and the total number of fans is recorded as the input degree of the node. The social impact of users can be

judged according to the user's output and the degree of input. User's degree reflects the social impact of users; the greater the degree, the greater the impact. User's degree indicates the number of users' fans. As you can see from Figure 2, the real-world phenomenon is that the most influential users in social networks are always in the minority, while the majority of users who pay attention to many people are in the minority,

and the vast majority of users only pay attention to a few people.

It is easy to see in Figure 2(a) that users 16 and 20 have more than 150 fans. But their trajectory in Figure 2(b) shows that there is no intersection in the place they are going. This means that, in social networks, even if many people are concerned about the same kind of people, it does not mean that there must be a common interest between these people. To mine the POI between them, some information must be added, such as the user's age, education, gender, nationality, etc. Experience shows that users from the same region tend to have the same tastes; people with the same educational experience tend to focus on the same hot news. Therefore, the user data we set is as follows: UserID, Age, Sex, Native place, and Educational background.

Place marking is an important condition for our DCAPR model. We use a potential factor to represent the location effect at a given time and then learn from the potential factor model. The site marking scheme determines how to allocate potential factors to specific locations.

To capture site features on different time scales, we represent a site with a five-tuple representation and then aggregate their contributions. Based on the empirical data analysis, we consider the characteristics of three site scales: time, longitude, and latitude. They are described by three different latent vectors. Therefore, place L_i is marked by five tuples $(m, w, lo_i, lg_i, \text{ and } IID)$, which satisfies $m \in \{1, \dots, 12\}$, $w \in \{1, \dots, 7\}$, $lo_i \in \{-180, +180\}$, and $lg_i \in \{-90, +90\}$, and IID is the place label. In addition, $L1.h8 \times W$, $L2.h16 \times W$, and $L3.h24 \times S$ are defined to represent the corresponding site potential factor matrix. $L1.h8 \times W$ represents the trajectory of user activity within 8 hours of the working day, $L2.h16 \times W$ represents the trajectory of user activity beyond 8 hours, and $L3.h24 \times S$ represents the trajectory of user activity during the Sunday and Saturday. W is the dimension of potential vector, representing the working days in a week.

After defining the location information of users, we use Cosine clustering algorithm to cluster the location information matrix in order to obtain the friends with the same interest points in the community. In this way, running Cosine clustering algorithm can get the user group, and each user belongs to only one group. In fact, users in the same group generally have the same preferences, and then they can recommend the information based on the past information of the users in the group. Then we can recommend information to users more accurately according to the information of these friends.

The Cosine clustering algorithm uses distance as similarity index to find K classes in a given dataset, and the center of each class is obtained according to all the values in the class. Each class is described by clustering center. For a given dataset X containing N d -dimensional data points and a class K to be partitioned, the Euclidean distance is chosen as the similarity index. The clustering objective is to minimize the sum of squares of all kinds of clustering, as shown in formula (1).

$$J = \sum_{k=1}^k \sum_{i=1}^n \|x_i - u_k\|^2 \quad (1)$$

In the past mobile trajectory model, data sparsity is a big problem. From Figure 2(b), we can see 6 users moving trajectories within one day. Observations show that each user is basically only active in a fixed number of places, and some users have repetitive movement paths, indicating that their behavior is similar between L_i and L_j in different places (L denotes location; i and j denote the number of different places). However, it is also easy to see that user with no. 9 is basically fixed in two places of activity, not intersected with others; similarity is zero. In addition, we find that there are other changes. User preferences vary with climate and mood.

Check-in variations at different spatial scales can describe user preferences from different perspectives: (1) Users can log on to their home system to communicate or shop with friends, or they can log on to APP in the office during the day to communicate with colleagues, or they can log in at night when they have a good time at the bar. (2) Users can visit more places in his / her home or office on weekdays. At weekends, he/she can check more information in some shopping centers or resorts. (3) Users may have different habits in different seasons. For example, he or she would ski in the cold north during the hot summer or visit the south coast in the hot summer. Therefore, it is impossible to capture all user features that need to be represented in different scales by modeling only the heterogeneity on a single scale.

3.2. Comments Scheme. Traditional machine learning methods mainly use the n -gram concept in natural language processing to extract text features and use TFIDF to adjust the weight of n -gram features and then input the extracted text features into the classifier such as Logistic regression, SVM for training. However, the above feature extraction methods have the problems of sparse data and dimension explosion, which is disastrous for the classifier, and makes the training model generalization ability limited. Therefore, it is often necessary to take some strategies to reduce dimension, such as stop word filtering, low-frequency n -gram filtering, LDA, etc.

We use CNN to classify sentences in our recommendation algorithm. A sentence is made up of many words. If a sentence has n words and the i th word is w_i and the word w_i is expressed as a vector of d -dimension after embedding, then the matrix of a sentence $w_{1:n}$ is $n \times d$ can be formalized as follows:

$$W_{1:n} = w_1 \oplus w_2 \oplus \dots \oplus w_n \quad (2)$$

A word window containing m words is represented as $W_{i:i+m-1}$, and a convolution kernel is a matrix of size $m \times d$. A feature f_i can be extracted by extracting a word window from an activation function, as follows:

$$f_i = F(M \cdot W_{i:i+m-1} + b) \quad (3)$$

where b is the corresponding intercept and F is Sigmoid activation function. A convolution kernel matrix is used to scan the whole sentence from the beginning of the clause to the end of the clause to extract the features of each word window, and a feature vector can be obtained, which is

represented as follows (where the default is not to padding the sentence):

$$X = \{x_1, x_2, \dots, x_{n-d+1}\} \quad (4)$$

If there are m filters, a vector of length m can be obtained by a layer of convolution and a layer of pooling.

$$z = [\widehat{C}_1, \widehat{C}_2, \dots, \widehat{C}_m] \quad (5)$$

where $\widehat{C}_i \in \mathbb{R}$; it is the result of Max pooling after extracting a feature map from a filter. Next, we carry out Max pooling for feature map extracted from a convolution kernel. Finally, the vector X is input to the full link layer to get the final feature extraction vector y .

$$y = W \cdot z + b \quad (6)$$

3.3. Image Feature Extract. In social networks, especially Twitter, QQ, WeChat, and other online social apps, users often share some pictures in the circle of friends. Some of these pictures were taken by the users themselves, and some were taken by other users. Some of these shared pictures have text descriptions, and some have no. Regardless of where these images come from, they represent the user's interest preferences at that moment. If we can accurately analyze and capture these points of interest from these images, we can provide relevant recommendation to users in a timely manner.

The Alexnet network structure model proposed by Alex in 2012 triggered a boom in neural network applications and won the championship of the 2012 Image Recognition Competition, making CNN the core algorithm model in image classification [28–30]. So here we use the CNN network to extract the semantic features of the image.

For CNN networks for processing user-image information, the input data of Layer 1 is represented by R, G, and B of the original image. For convolution operations, the size of convolution kernel is as follows: $11 \times 11 \times 3$, $5 \times 5 \times 96$, $3 \times 3 \times 256$, $3 \times 3 \times 384$. For example, on the first layer, if the original image size is 227×227 , then the image is convoluted by the convolution kernel of $11 \times 11 \times 3$. Each convolution of the original image generates a new pixel. The convolution kernel moves along the x-axis and y-axis directions of the original image. The moving step is 4 pixels. Therefore, the convolution kernel generates $(227 - 11) / 4 + 1 = 55$ pixels (227 pixels minus 11, exactly 54 pixels, plus 11 subtracted to generate one pixel) and 55×55 pixels of rows and columns form the pixel layer after convolution of the original image.

As ReLU deep convolution network is much faster than Tanh and sigmoid based network training, we have chosen the ReLU function in our proposed model. These pixel layers are processed by pool operation (pool operation). The scale of pool operation is 3×3 , and the step size of pool operation is 2. Then the image after pooling is normalized, and the normalized operation scale is 5×5 . The Dropout operation is more effective in preventing overfitting of neural networks. Regular methods are used to prevent overfitting of models as generally as linear models, while Dropout is implemented

in neural networks by modifying the structure of the neural network itself. For a certain layer of neurons, some neurons are randomly deleted by the defined probability, while keeping the individuals of the input layer and the output layer neurons unchanged, and then the parameters are updated according to the learning method of the neural network. In the next iteration, some neurons are rerandomly deleted until the end of the training. The fully connected layer is actually a convolution operation in which the convolution kernel size is the feature size of the upper layer output. The result of the convolution is a node, which corresponds to a point of the fully connected layer. The convolution takes local features, and the full join is to reassemble the previous local features into a complete graph through the weight matrix.

4. Deep CNN-Assisted Personalized Recommendation

4.1. DCAPR Framework. In this paper, we propose a novel deep CNN-assisted personalized recommendation DCAPR. As shown in Figure 3, DCAPR consists of three layers of progressively progressive recommendation layers: a rough recommendation layer, an enhanced recommendation layer, and an accurate recommendation layer.

The first layer is a rough recommendation layer. By comparing the user trajectory sequence of the mobile social network, the similarity of the user's moving trajectory sequence is compared, and several candidate buddy users are picked out. But, among these candidate users, there may be "fake-friends", that is, although the two people have similar movement trajectories, the points of interest are completely different and cannot be regarded as true friends. For example, user A and user B have the same trajectory within a certain period of time and are all active in a certain mall. However, User A is concerned with clothing, while User B is concerned with the e-sports game upstairs in the clothing store. Therefore, DCAPR built a second layer of recommendation framework to improve this problem.

The second layer is the enhancement layer. Based on the candidate friends selected in the previous layer, the CNN convolutional neural network is used to extract features of various image content uploaded by the candidate users on the mobile social platform. According to the visual content of the image, the interest association between the users can be further explored, so that the candidate friends can be refined and filtered.

The third layer is the accurate recommendation layer. For the text, the deep learning CNN classification method is combined with the context to extract and retrieve the semantic content of the text, and the vocabulary defined as illegal is deleted or the illegal vocabulary is occupied by the recommendation. Based on the previous two layers, the semantic comparison of the posts posted by the user is carried out to construct a deep hierarchical prediction model for more accurate recommendation.

The model integrates the location information of the user in the real world, the pictures shared by the user in the social network, and the text information published or forwarded by

FIGURE 3: The framework of DCAPR model. The framework consists of three layers: a rough recommendation layer, an enhanced recommendation layer, and an accurate recommendation layer.

the user on a platform. Therefore, in the same space, the user is recommended for images, news, and places by calculating the similarity among the semantic features of the characters, the semantic features of the images, and the auxiliary location information.

4.2. Rough Recommendation Layer. In order to recommend a location point that may be of interest to a mobile social network user, first of all, look for his neighbors in the mobile social network. Since his neighbors and the user may have similar points of interest, we can recommend the place where the friend has been to the user, and vice versa. In this layer, we temporarily do not consider the context of the user's location sequence and only calculate and analyze the user's behavior characteristics from the perspective of time and space, so as to roughly filter out several friends of the mobile social network users to prepare for future recommendation information. Since mobile social network users have different check-in times and ways for location points, we divide the rough recommendation layer into two modes: frequency position point mode and trajectory sequence matching mode.

4.2.1. Frequency Position Point Mode. The degree of interest of the user at the location point is determined based on the user's frequency of check-in at a certain point. We first calculate the frequency of each user's access to a certain location, compare it with the preset frequency threshold, and then select the users who visit the location with a frequency greater than a fixed threshold to form a user neighbor group. Since the nature of each user's work may be different, the working time may be different, and the labor intensity may be different,

such statistics may cause large errors. For example, user A and user B frequently go to a famous gym, but user A is a courier, he is a customer who delivers courier items to the gym; and user B is a member of the gym, he is going to exercise every time. Therefore, it is easy to generate misjudge whether two users are neighboring users only by the number of occurrences at a certain place. In order to avoid this defect, we have improved the statistical method by using the user's check-in frequency ratio instead of the check-in frequency. That is, we count the ratio of the number of times each user has a checkpoint l_i ($1 \leq i \leq n$) to the total number of check-ins of the user in a fixed time range (for example, 1 week), and the specific calculation is as shown as formulas (7) and (8).

$$R_{i,j} = \frac{p_{i,j}}{\sum_{j=1}^n p_{i,j}} \quad (7)$$

$$S_{i,j} = \sqrt{\frac{\sum (R_{i,j} - \bar{R}_i)^2}{n-1}} \quad (8)$$

where n represents the total number of location points and $R_{i,j}$ indicates the check-in frequency ratio of user i at the location point j . And $p_{i,j}$ is the percentage of user i who checked in at location j ; $\bar{R}_i$ is the average percentage of each user who checked in at all locations.

According to common sense of life, we know that the greater the proportion indicates that the user is more interested in the location. According to the probability of sign-in at each location point, we can list each location's interest point table for each user in order of high to low proportion and then

FIGURE 4: The check-in ratios for five users, the probability of each user's check-in at this location, and the standard deviation of the number of check-ins.

calculate the variance according to the location interest point table to calculate the similarity between users.

Table 1 lists the frequency of check-in frequencies for five randomly selected users at specific locations.

Table 1 lists the check-in frequency ratios of five users randomly selected in the Gowalla dataset at the location of the tag 420315. As can be seen from Table 1, in terms of the number of times, the user numbered UserId 66 has checked in 47 times at the place, and the number of check-ins is greater than the remaining four users. However, it is obviously wrong to judge that the user is very interested in the location point 420315 because the user has a sign-in ratio of 17.1% at that location. The other user UserId 7 has only 21 check-ins at this location. This number of check-ins is the least compared to the number of other four users. However, his/her check-in ratio at position 420315 is 28%, which clearly indicates that his/her interest in the location is very strong.

Figure 4 shows the check-in ratios for five users, the probability of each user's check-in at this location, and the standard deviation of the number of check-ins. The blue color in the figure indicates the sign-in ratio of each user at the location point 420315; red indicates the proportion of each user's ratio of the check-in at this location compared to the total check-in ratio of the five users; green indicates the calculated standard deviation. The closer the standard deviation to the sign-in ratio, the more intense his or her interest in the location.

4.2.2. Trajectory Sequence Matching Mode. According to the sequence of moving trajectories, we can analyze from two dimensions in space and time, and by comparing the motion trajectories of the users, we can find the nearest neighbors similar to the trajectory sequence of the user. And then, the location contained in the nearest neighbor's trajectory

sequence is recommended to users who are similar to their trajectory but have not been to the location. For mobile network user location recommendation, we divide it into three steps. The first step is the preprocessing stage. We obtain the movement trajectory and movement time interval of each user by preprocessing the dataset, thus forming the user's movement trajectory sequence, as shown as Figure 5. In Step 2, we regard the sequence of moving tracks as a string, each character representing a place and setting a threshold. When comparing the motion trajectories between two users, once there is a common substring whose length exceeds the threshold in their trajectory, it is considered that the two users find each other as the nearest neighbor. If the common substring's length is less than the threshold, step 3 is performed; that is, the similarity is simply considered spatially. We first count the number of times each user has been to each location, and then use the Cosine method to calculate the similarity between users.

Cosine Clustering for User Location. How to accurately extract the personalized information demand preference model of mobile users with location changes according to the change rule of users' personalized demand for information changes with location changes will become the key of location-based mobile communication network information recommendation service. In the proposed model, we learn the user's personalized demand for information according to the cyclical changes of the user's position with time and extract the user's personalized information demand preference model. The user's geographical location is constantly changing within a certain period of time (one day, one week, or one month), and the information services required in different geographical locations are also different. However, within a plurality of time periods (a few days), there is a certain regularity in the change of the geographical location of the mobile user.

TABLE 1: Check-in status of 5 users of mobile social network at location point 420315.

UserID	LId	Check-in frequency	Total Check-in Frequency	Ratio	Probability
0	420315	28	224	0.125	0.117116535
7	420315	21	75	0.28	0.262341038
31	420315	43	100	0.43	0.402880879
52	420315	21	342	0.061404	0.05753139
66	420315	47	275	0.170909	0.160130159

TABLE 2: The locations and times of three users visited in the past week.

UserID	Locations								
	Natorium	Gym	Hospital	Bank	Museum	Restaurant	Starbucks	Library	Bowling alley
U _A	2	4	1	1	1	0	0	0	0
U _B	3	0	1	0	1	4	2	1	0
U _C	0	0	0	0	1	3	3	2	1

In location-based social networks, all POIs have location attributes, and user behavior has temporal and spatial sequential patterns. At present, the social network can obtain the user's trajectory through technical means such as check-in and GPS in the social network. According to the cross information of the user's trajectory and combined with the rating of the location, the preference of the user can be found. However, the recommendation system based on location-based social network should not only focus on the user's own trajectory sequence, but also focus on the social relationship between users, so as to select the top k sites to recommend to users through the ratings of other users with high similarity. For instance, as shown in Figure 6, according to the user's trajectory, the user UA has visited Natatorium, Gym, Hospital, Bank, Museum, etc. in the past week. Also in the past week, user UB has visited Natatorium, Restaurant, Hospital, Museum, Starbucks, and Library, respectively. Another user UC went to Bowling alley, Restaurant, Museum, Library, and Starbucks.

Table 2 shows the places where the three users in Figure 6 have been visited and the number of times each place has been visited. From Table 2, we can see the social relationship and similarity between UA, UB, and UC. Therefore, we can recommend to users UA, UB, and UC the sites that they may be interested in according to the similarity.

We divide each time period into N segments based on the number of user activities. Then, the sequence of change of the geographical location of the mobile user in a time period is $l_i, i=1,2,\dots,N$, and in all M time periods, the sequence of position change sequence of each mobile user is

$$P_{i,j} = (l_{ij})_{M \times N}, \quad i = 1, 2, \dots, M, \quad j = 1, 2, \dots, N \quad (9)$$

The location-based mobile user preference model is a two-tuple $U_k=(u_i, L_j)$, where U_k represents the k th user in a mobile social network. And the two-tuple $U_k=(u_i, L_j)$ represents the i th user at a certain location L_j . Suppose there are two mobile social network users A and B. The application characteristics of all network service items in the locations L_a and L_b are $U_a=(u_a, L_a)$ and $U_b=(u_b, L_b)$, respectively; u_a and u_b which are all network service multidimensional

feature vectors used by the two mobile social network users at locations L_x and L_y are normalized such that they have the same length. The location-based user preference similarity can be defined as follows:

$$\begin{aligned} \text{sim}(U_a, U_b) &= \frac{1}{e^{\text{dis}(L_a, L_b)}} \\ &\times \frac{\sum_{i=1}^n (u_{a,i} \times u_{b,i})}{\sqrt{\sum_{i=1}^n (u_{a,i})^2} \times \sqrt{\sum_{i=1}^n (u_{b,i})^2}} \end{aligned} \quad (10)$$

Obviously, on the one hand, when two mobile users are in the same position, the distance between them is 0, $\text{dis}(L_a, L_b)=0$, at this time, $e^{\text{dis}(L_a, L_b)} = 1$. For any two different locations of mobile users, due to $\text{dis}(L_a, L_b)>0$, then $0 < e^{\text{dis}(L_a, L_b)} < 1$. If and only if $a=b$, $\text{sim}(u_a, u_b)=1$. Therefore, for any two mobile users, the similarity $\text{sim}(u_i, u_j) \in [0, 1]$. According to Table 2, we can calculate the similarity between U_A , U_B , and U_C ; the result is shown in Table 3.

On the basis of the similarity calculation results in Table 3, we can judge the user's preference from the trajectory of the place where the user has been and calculate the similarity between the trajectory of the user and the user. As can be seen from Table 3, the similarity between User B and User C is significantly higher than that between user A and user C and between user A and user B. In this way, we can recommend the places where User B has been to User C according to the interests of User C.

4.3. Enhanced Recommendation Layer. CNN network for image processing adopts seven-layer structure, and CNN network for text processing adopts three-layer frame structure. Firstly, we rescale images to 227×227 . And then we use 8-layer VGGNet to extract an image feature map.

As shown as Figure 7, semantic information is extracted from pictures which are posted by different users, and the user is tagged with various categories. For example, from the picture that user 1 has posted, we can deduce that the user may not only like to travel, but also may be a photography enthusiast. Therefore, the user 1 can be affixed with a travel-loving label or a photographer's label; similarly, the user 3

TABLE 3: Cosine formula is used to calculate the similarity of 3 users.

UserID	UserID		
	U_A	U_B	U_C
U_A	0	0.2949	0.0426
U_B	0.2949	0	0.7578
U_C	0.0426	0.7578	0

FIGURE 5: An example about three users’ trajectory.

FIGURE 6: Three users UA, UB, and UC outdoor trajectory of the past week.

FIGURE 7: An example of the extracted semantic information from posted pictures by different users and label these users.

is the same. The user 2 can not only be tagged with travel and photography, but also can derive the user's preferred sport according to the content in the figure. If the motion tag continues to be subdivided, information about the user's preference to practice yoga can be obtained. Therefore, if the user has just arrived in the city, there is no local trajectory generated; that is, when the recommendation based on the location information is a cold start, we can recommend the location that the user may be interested in according to the picture that the user has posted.

4.4. CNN Network for Comments. The third layer of our model is the extraction of text features from comments or forwarded articles from users in social networking forums. The text extraction method refers to the extraction of text features using the CNN convolution network. First, the original text is preprocessed, including word segmentation, deactivation, etc., and then the preprocessed text is vectorized using the skip-gram model in word2vec. Finally, each sentence is transformed into a matrix form. Next, the feature extraction and classification of the comment statements can be performed using the CNN network. This process is very similar to the image feature extraction using CNN. When convolving the text matrix, the text matrix is convolved using filters of different lengths. The width of the filter is equal to the length of the longest word vector in the sentence, and then the vector extracted by each filter is operated using Max pooling. Finally, each filter corresponds to a number, and the results of these filters are spliced together to obtain a vector characterizing the sentence.

5. Experiments

5.1. Dataset and Experimental Settings. Using technologies such as user check-in information and GPS positioning, the

TABLE 4: Statistics of dataset. We separated images from geographic information from 196,591 users.

Nodes	196591
Edges	950327
Nodes in largest WCC	196591 (1.000)
Edges in largest WCC	950327 (1.000)
Nodes in largest SCC	196591 (1.000)
Edges in largest SCC	950327 (1.000)
Average clustering coefficient	0.2367
Number of triangles	2273138
Fraction of closed triangles	0.007952
Diameter (longest shortest path)	14
90-percentile effective diameter	5.7
Check-ins	6442890

geographic location and movement trajectory of the mobile network user can be obtained very accurately.

We consider using a publicly available Gowalla dataset for our proposed model. Gowalla dataset is a location-based social networking website where users share their locations by checking-in. The friendship network is undirected and was collected using their public API and consists of 196,591 nodes and 950,327 edges. We have collected a total of 6,442,890 check-ins over the period of Feb. 2009-Oct. 2010.

Table 4 presents the statistics of the dataset's detail. The dataset provides information such as user identification, age, sex, occupation, time, location, image, comments, etc. Following [31], we removed all users who have less than 10 check-ins and locations which have fewer than 15 check-ins. Finally, the collection constructed contained 837,352

TABLE 5: Comparison of the evaluation results of four models on test sets.

Method	Precision	Recall	F1
PACE	0.0976	0.0913	0.094344944
VPOI	0.1283	0.1208	0.124437094
SEER	0.1462	0.1483	0.147242513
DCAPR	0.1528	0.1567	0.154725428

TABLE 6: Results of recommendation index in the case of recommendation number increase and recommendation dimension fixed.

Method	Number of Recommended	Precision	Recall	F1
PACE	10	0.0924	0.0815	0.0866
PACE	20	0.0976	0.0913	0.0943
VPOI	10	0.1064	0.1059	0.1061
VPOI	20	0.1283	0.1208	0.1244
SEER	10	0.1305	0.1297	0.1301
SEER	20	0.1362	0.1383	0.1372
DCAPR	10	0.1398	0.1387	0.1392
DCAPR	20	0.1528	0.1567	0.1547

subtrajectories with corresponding locations, comments, and images. Table 3 presents the statistics of the dataset's detail.

5.2. *Baselines.* For comparison with the proposed model, we consider the following baselines:

- (i) *Preference and Context Embedding (PACE).* Reference [31] pointed out the current POI recommendation methods are designed for specific data and problems, and a general semisupervised learning model is proposed. That is, the preference and context embedding model can utilize the information of neighboring users and locations to alleviate the data sparse problem of the recommendation system.
- (ii) *Visual Content Enhanced POI Recommendation (VPOI).* Reference [25] proposed a POI recommendation model with visual content enhancement based on CNN and probability matrix factorization. The author studied how to incorporate image content information to improve the POI recommendation. VPOI uses CNN to extract features from image content and constructs a probabilistic theme model through user-image relationship, POI-image relationship, and user-POI relationship. Finally, the image feature extraction and probability topic model are integrated into one unified. The optimization function is built in the framework, and the Negative Sampling method is used to optimize the parameters.
- (iii) *Sequential Embedding Rank (SEER).* Reference [32] made a point of interest recommendation based on the user's interest preferences and mobile mode. Specifically, SEER model uses distributed representation technology to learn the embedded representation of the user and then embed the user as a constraint into the paired sorting model to capture the sequence pattern of the user's behavior. At the same time, it also incorporates time and space information.

5.3. *Experimental Results and Analysis.* The proposed method is evaluated based on Precision, Recall, and Accuracy using a real-world dataset. We adopt the evaluation index in information retrieval to evaluate our method and contrast model method. Specifically, we used Precision and Recall two values to evaluate the two formulas. The definitions are as follows:

$$Precision = \frac{|L_{visited} \cap L_{N,R}|}{M} \quad (11)$$

$$Recall = \frac{|L_{visited} \cap L_{N,R}|}{|L_{visited}|} \quad (12)$$

$$F1 = \frac{2 * Precision * Recall}{Precision + Recall} \quad (13)$$

where $L_{visited}$ represents the set of locations contained in the Gowalla dataset and $L_{N,R}$ represents the set of places with the recommended number of M . The final values for Precision and Recall are averaged over the dataset for all users. The related experimental results are shown in Table 5.

Figure 8 shows the Precision, Recall, and F1-Score of different models. From Table 5 and Figure 8, we can see that our model DCAPR is significantly better than the other three benchmark comparison algorithms, because we incorporate multisource heterogeneous information, such as images, text, geographic location information, etc. The integration of multisource heterogeneous information helps to more accurately characterize the user's access behavior, which in turn enables more accurate modeling.

In Table 6, when the dimensions remain the same and when the number of recommendations increases from 10 to 20, the results of each model on the corresponding evaluation indicators (Precision and Recall) are also improved. This is defined by the calculation formulas of Precision and Recall. When more places are recommended to the user, it is easier to hit the already visited records of the user in the test dataset, thus causing the value to be large.

TABLE 7: Results of recommendation index in the case of dimension increase and recommendation dimension number fixed.

Method	Dimension	Precision	Recall	F1
PACE	100	0.0924	0.0815	0.0866
PACE	500	0.0965	0.0902	0.0932
VPOI	100	0.1064	0.1059	0.1061
VPOI	500	0.1279	0.1264	0.1271
SEER	100	0.1305	0.1297	0.1301
SEER	500	0.1358	0.1376	0.1367
DCAPR	100	0.1398	0.1387	0.1392
DCAPR	500	0.1525	0.1563	0.1544

FIGURE 8: Precision, Recall, and F1-Score with different number of recommendations.

Table 7 shows that when the number of recommendations is consistent and the dimension is increased from 100 to 500, the values of the respective models on the corresponding evaluation indicators are correspondingly increased. This is because more dimensions can describe the hidden feature values more carefully, which will make the model effect increase. However, as can be seen from Table 7, the increase in the dimension does not make the model continue to improve, because the oversized dimension leads to overfitting.

6. Conclusion and Future Work

The development of intelligent mobile devices has driven the rapid development of mobile social networks. Deep learning-driven algorithms and models can promote wireless network analysis and resource management and help to cope with the growth of communication and computing in emerging mobile applications. In this paper, by means of in-depth learning, the user behavior sequence pattern is integrated into the recommendation system, which is helpful to discover the dependencies between user behaviors and improve the quality of recommendation. It is for this purpose we presented a novel social network recommendation algorithm framework based on mobile wireless network. Finally, a comprehensive experiment of the DCAPR method is carried out using the user dataset from Gowalla. The results show that the baseline improvement is more significant when the user's behavior sequence is fused with the user's posted images, text, and so on through DCAPR framework.

Now, the recommendation system based on deep learning faces two main problems: one is how to better combine multisource data for recommendation; the other is how to analyze the intermediate process and the final result from a mathematical perspective. The deep learning-based recommendation system usually uses the end-to-end model to predict the user's preference for the project by using the multisource heterogeneous data as input. The recommendation system involves many auxiliary data: comments, tags, user portrait information, user socialization, and recommended situation information (time; location). It can be seen that the current recommendation system needs many modeling factors. In the future, if the multiobjective optimization [33–37] and multisource heterogeneous data can be combined to dynamically evolve user preferences and project features, the performance of the recommendation system can be improved. For the second question, we are inspired by the research of Sun et al. [38–48], and we may be able to find out the answer we want.

At present, learning algorithms in mobile wireless systems are immature and inefficient. More endeavors are needed to bridge the gap between deep learning and wireless communications and mobile computing research. Specifically for mobile wireless network recommendation system, the application of in-depth learning in location-based social network recommendation system mainly focuses on sequential pattern modeling. How to integrate a large number of implicit and explicit heterogeneous spatiotemporal data of mobile wireless network users through in-depth learning,

so as to build a unified recommendation framework, is the future direction of development.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

The work was funded by the National Natural Science Foundation of China (Grants nos. 61702277 and 61872219).

References

- [1] X. Zhang and Q. Zhu, "Hierarchical caching for statistical QoS guaranteed multimedia transmissions over 5G edge computing mobile wireless networks," *IEEE Wireless Communications Magazine*, vol. 25, no. 3, pp. 12–20, 2018.
- [2] Z. Sheng, C. Mahapatra, V. C. M. Leung, M. Chen, and P. K. Sahu, "Energy efficient cooperative computing in mobile wireless sensor networks," *IEEE Transactions on Cloud Computing*, vol. 6, no. 1, pp. 114–126, 2018.
- [3] L. Qi, R. Wang, S. Li et al., "Time-aware distributed service recommendation with privacy-preservation," *Information Sciences*, vol. 480, pp. 354–364, 2018.
- [4] Y. Xu, L. Qi, W. Dou, and J. Yu, "Privacy-preserving and scalable service recommendation based on simhash in a distributed cloud environment," *Complexity*, vol. 2017, Article ID 3437854, 9 pages, 2017.
- [5] X. Xu, Q. Liu, Y. Luo et al., "A computation offloading method over big data for IoT-enabled cloud-edge computing," *Future Generation Computer Systems*, vol. 95, pp. 522–533, 2019.
- [6] W. Gong, L. Qi, and Y. Xu, "Privacy-aware multidimensional mobile service quality prediction and recommendation in distributed fog environment," *Wireless Communications and Mobile Computing*, vol. 2018, Article ID 3075849, 8 pages, 2018.
- [7] X. Xu, S. Fu, L. Qi et al., "An IoT-Oriented data placement method with privacy preservation in cloud environment," *Journal of Network and Computer Applications*, vol. 124, pp. 148–157, 2018.
- [8] S. Zhang, L. Yao, and A. Sun, "Deep learning based recommender system: A survey and new perspectives," *ACM Computing Surveys*, vol. 1, no. 1, pp. 1–35, 2018.
- [9] M. Gruteser and D. Grunwald, "Anonymous usage of location-based services through spatial and temporal cloaking," in *Proceedings of the 1st International Conference on Mobile Systems, Applications and Services, MobiSys 2003*, pp. 31–42, May 2003.
- [10] G. Adomavicius and A. Tuzhilin, "Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions," *IEEE Transactions on Knowledge and Data Engineering*, vol. 17, no. 6, pp. 734–749, 2005.
- [11] F. Ricci, L. Rokach, and B. Shapira, "Recommender systems: introduction and challenges," in *Recommender Systems Handbook*, pp. 1–34, Springer US, 2015.

- [12] R. Burke, "Hybrid recommender systems: survey and experiments," *User Modeling and User-Adapted Interaction*, vol. 12, no. 4, pp. 331–370, 2002.
- [13] Y. Zheng, L. Zhang, Z. Ma, X. Xie, and W.-Y. Ma, "Recommending friends and locations based on individual location history," *ACM Transactions on the Web (TWEB)*, vol. 5, no. 1, article no. 5, 2011.
- [14] C. Chen, P. Zhao, L. Li, J. Zhou, X. Li, and M. Qiu, "Locally connected deep learning framework for industrial-scale recommender systems," in *Proceedings of the 26th International Conference on World Wide Web Companion, International World Wide Web Conferences Steering Committee*, pp. 769–770, Perth, Australia, 2017.
- [15] S. Li, J. Kawale, and Y. Fu, "Deep collaborative filtering via marginalized denoising auto-encoder," in *Proceedings of the 24th ACM International Conference on Information and Knowledge Management, CIKM 2015*, pp. 811–820, ACM, Australia, October 2015.
- [16] N. Kriegeskorte, "Deep neural networks: a new framework for modeling biological vision and brain information processing," *Annual Review of Vision Science*, vol. 1, no. 1, pp. 417–446, 2015.
- [17] B. Hidasi, M. Quadrana, A. Karatzoglou, and D. Tikk, "Parallel recurrent neural network architectures for feature-rich session-based recommendations," in *Proceedings of the 10th ACM Conference on Recommender Systems, RecSys 2016*, ACM, pp. 241–248, USA, September 2016.
- [18] B. Hidasi and A. Karatzoglou, "Recurrent neural networks with top-k gains for session-based recommendations," in *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, ACM, pp. 843–852, Torino, Italy, October 2018.
- [19] D. Jannach, L. Lerche, F. Gedikli, and G. Bonnin, "What recommenders recommend: analysis of accuracy, popularity, and sales diversity effects," in *Proceedings of the International Conference on User Modeling, Adaptation, and Personalization*, pp. 25–37, Springer, Berlin, Heidelberg, 2013.
- [20] S. P. Chatzis, P. Christodoulou, and A. S. Andreou, "Recurrent latent variable networks for session-based recommendation," in *Proceedings of the 2nd Workshop on Deep Learning for Recommender Systems (DLRS 2017)*, ACM, pp. 38–45, Como, Italy, August 2017.
- [21] V. Bogina and T. Kuflik, "Incorporating dwell time in session-based recommendations with recurrent neural networks," in *Proceedings of the 1st Workshop on Temporal Reasoning in Recommender Systems in CEUR Workshop*, pp. 57–59, Como, Italy, August 2017.
- [22] T. Ebesu and Y. Fang, "Neural semantic personalized ranking for item cold-start recommendation," *Information Retrieval Journal*, vol. 20, no. 2, pp. 109–131, 2017.
- [23] Y. Kim, "Convolutional neural networks for sentence classification," <https://arxiv.org/abs/1408.5882>, 2014.
- [24] Y. Wu, C. DuBois, A. X. Zheng, and M. Ester, "Collaborative denoising auto-encoders for top-N recommender systems," in *Proceedings of the 9th ACM International Conference on Web Search and Data Mining, WSDM 2016*, pp. 153–162, ACM, San Francisco, USA, February 2016.
- [25] S. Wang, Y. Wang, J. Tang, K. Shu, S. Ranganath, and H. Liu, "What your images reveal: exploiting visual contents for point-of-interest recommendation," in *Proceedings of the 26th International World Wide Web Conference, WWW 2017*, pp. 391–400, Australia, April 2017.
- [26] J. Davidson, B. Liebald, J. Liu, P. Nandy, and T. Van Vleet, "The YouTube video recommendation system," in *Proceedings of the 4th ACM Recommender Systems Conference (RecSys '10)*, pp. 293–296, Barcelona, Spain, September 2010.
- [27] W.-T. Chu and Y.-L. Tsai, "A hybrid recommendation system considering visual information for predicting favorite restaurants," *World Wide Web*, vol. 20, no. 6, pp. 1313–1331, 2017.
- [28] Y. Zheng, J. Zhu, W. Fang, and L. Chi, "Deep learning hash for wireless multimedia image content security," *Security and Communication Networks*, vol. 2018, Article ID 8172725, 13 pages, 2018.
- [29] Z. Ligang and C. Song, "Fast near-duplicate image detection in riemannian space by a novel hashing scheme," *CMC: Computers, Materials & Continua*, vol. 56, no. 3, pp. 529–539, 2018.
- [30] W. Fang, Z. Feihong, S. Victor, and D. Yewen, "A method for improving CNN-based image recognition using DCGAN," *CMC: Computers, Materials & Continua*, vol. 57, no. 1, pp. 167–178, 2018.
- [31] 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 *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2017*, ACM, pp. 1245–1254, Halifax, Canada, August 2017.
- [32] S. Zhao, T. Zhao, I. King, and M. R. Lyu, "Geo-teaser: geo-temporal sequential embedding rank for point-of-interest recommendation," in *Proceedings of the 26th International Conference on World Wide Web Companion, International World Wide Web Conferences Steering Committee*, pp. 153–162, Perth, Australia, April 2017.
- [33] Y. Yuan and W. Banzhaf, "ARJA: automated repair of java programs via multi-objective genetic programming," *IEEE Transactions on Software Engineering*, 2018.
- [34] Y. Yuan, Y.-S. Ong, A. Gupta, and H. Xu, "Objective reduction in many-objective optimization: evolutionary multiobjective approaches and comprehensive analysis," *IEEE Transactions on Evolutionary Computation*, vol. 22, no. 2, pp. 189–210, 2018.
- [35] Y. Yuan and H. Xu, "Multiobjective flexible job shop scheduling using memetic algorithms," *IEEE Transactions on Automation Science and Engineering*, vol. 12, no. 1, pp. 336–353, 2015.
- [36] Y. Yuan, H. Xu, B. Wang, B. Zhang, and X. Yao, "Balancing convergence and diversity in decomposition-based many-objective optimizers," *IEEE Transactions on Evolutionary Computation*, vol. 20, no. 2, pp. 180–198, 2016.
- [37] Y. Yuan, H. Xu, B. Wang, and X. Yao, "A new dominance relation-based evolutionary algorithm for many-objective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 20, no. 1, pp. 16–37, 2016.
- [38] W. W. Sun, "Stabilization analysis of time-delay Hamiltonian systems in the presence of saturation," *Applied Mathematics and Computation*, vol. 217, no. 23, pp. 9625–9634, 2011.
- [39] M. Han, X. Hou, L. Sheng, and C. Wang, "Theory of rotated equations and applications to a population model," *Discrete and Continuous Dynamical Systems - Series A*, vol. 38, no. 4, pp. 2171–2185, 2018.
- [40] W. Sun and L. Peng, "Observer-based robust adaptive control for uncertain stochastic Hamiltonian systems with state and input delays," *Nonlinear Analysis: Modelling and Control*, vol. 19, no. 4, pp. 626–645, 2014.

- [41] F. Li and G. Du, "General energy decay for a degenerate viscoelastic Petrovsky-type plate equation with boundary feedback," *Journal of Applied Analysis and Computation*, vol. 8, no. 1, pp. 390–401, 2018.
- [42] J. Liu and A. Qian, "Ground state solution for a Schrödinger-Poisson equation with critical growth," *Nonlinear Analysis: Real World Applications*, vol. 40, pp. 428–443, 2018.
- [43] J. Jiang, L. Liu, and Y. Wu, "Positive solutions to nonlinear fractional differential equations involving Stieltjes integrals conditions," *Journal of Nonlinear Sciences and Applications. JNSA*, vol. 10, no. 10, pp. 5360–5372, 2017.
- [44] H. Liu and H. Gao, "Global well-posedness and long time decay of the 3D Boussinesq equations," *Journal of Differential Equations*, vol. 263, no. 12, pp. 8649–8665, 2017.
- [45] K. M. Zhang, "On a sign-changing solution for some fractional differential equations," *Boundary Value Problems*, vol. 2017, no. 59, 8 pages, 2017.
- [46] Y. Guo, "Globally robust stability analysis for stochastic cohen-grossberg neural networks with impulse and time-varying delays," *Ukrainian Mathematical Journal*, vol. 69, no. 8, pp. 1220–1233, 2017.
- [47] H. Tian and M. Han, "Bifurcation of periodic orbits by perturbing high-dimensional piecewise smooth integrable systems," *Journal of Differential Equations*, vol. 263, no. 11, pp. 7448–7474, 2017.
- [48] Y. A. Amer, A. M. S. Mahdy, and E. S. M. Youssef, "Solving fractional integro-differential equations by using sumudu transform method and hermite spectral collocation method," *Computers, Materials and Continua*, vol. 54, no. 2, pp. 161–180, 2018.