Temporal-Spatial Collaborative Prediction for LTE-R Communication Quality Based on Deep Learning

JIANTAO QU[ID][1,2]**, FENG LIU**[ID][1,2]**, YUXIANG MA**[ID][3,4]**, AND JIAMING FAN**[ID][1,2]

[1]School of Computer and Information Technology, Beijing Jiaotong University, Beijing 100044, China
[2]Engineering Research Center of Network Management Technology for High Speed Railway, Ministry of Education, Beijing 100044, China
[3]School of Computer and Information Engineering, Henan University, Kaifeng 475004, China
[4]Henan Key Laboratory of Big Data Analysis and Processing, Henan University, Kaifeng 475004, China

Corresponding author: Feng Liu (16112084@bjtu.edu.cn)

ABSTRACT In recent years, long term evolution for railway (LTE-R) has been a promising technology to meet the growing demand for railway wireless communication. To realize the active maintenance of LTE-R base station, it is of great significance to precisely predict the communication quality (CQ) of LTE-R base station. Given that the existing LTE CQ prediction methods can not support the active maintenance of LTE-R base station. Furthermore, the LTE-R base station has its unique characteristics in time relationship and regional impact, one of the most challenging problems is to effectively integrate the temporal and spatial information to improve the effect of CQ prediction. To solve the above problems, we choose daily evolved radio access bearer (E-RAB) abnormal release ratio as the CQ indicator, and propose a new deep learning-based CQ prediction approach for LTE-R. Considering the influence of adjacent base stations, this method conducts temporal-spatial collaborative prediction on multivariate time series collected from the CQ data of these stations. First, to eliminate the negative effect of redundant variables, a new variable filter method based on max-relevance, and min-redundancy (MRMR) criterion and binary particle swarm optimization (BPSO) is proposed to select a variable set from the CQ data of related base stations. Second, a new recurrent convolutional neural network (RCNN) model with a self-attention mechanism is proposed to extract temporal-spatial features from the selected variable set. With these features, we build a collaborative prediction model for CQ prediction. Experimental results on real-world LTE-R CQ datasets demonstrate the superiority of the proposed method in CQ prediction.

INDEX TERMS Time series forecasting, deep learning, LTE-R, communication quality prediction.

I. INTRODUCTION

Long term evolution for railway (LTE-R) system is considered as the newest technical solution for railway wireless communications. With the rapid development of railway transportation, high-speed and heavy-haul railway become popular, which puts forward higher requirements on railway wireless communications. In this situation, it is vital to evolve the traditional global system for mobile communication for railway (GSM-R) system into LTE-R system. LTE-R can provide a safer, more reliable and higher-speed communication

The associate editor coordinating the review of this manuscript and approving it for publication was Qichun Zhang[ID].

network for railway operators. As a result, the increasing demands of railway mobile communication services (e.g., real-time control of trains, on-board video transmission, passenger Internet access [1]) can be satisfied. At present, LTE-R has been gradually applied to the high-speed railway, heavy-haul railway, and metro. Compared with the traditional railway transportation, in above railway application scenarios, the train speed is faster, the length of the train is longer, and the departure frequency is higher, which puts forward higher requirements for the communication quality (hereinafter called CQ) of LTE-R. Generally, the main method to ensure LTE-R CQ is periodic inspection, which is a passive way of maintenance. Active maintenance based on LTE CQ

prediction can determine the LTE-R base station (or Evolved Node B, eNodeB) to be maintained in advance, so as to improve the safety of LTE-R system.

Presently, there are few studies concerning CQ prediction for railway wireless communication, no matter GSM-R or LTE-R. However, in other LTE application scenarios, thanks to the detailed, frequent, high-granularity, real-time reporting of LTE, we can use data-driven technology to further analyze and process the data [2]. There are some researches on CQ prediction in other LTE application scenarios [2]–[6]. But for LTE-R CQ prediction, these researches have the following problems: 1) From the application perspective, conventional researches mainly focus on short-term CQ prediction, and the CQ indicators in these researches can not reflect the operation status of core communication services. Therefore, the prediction results of these researches can not meet the safety requirements of the railway for the core communication services and support the active maintenance of LTE-R base station. 2) From the theoretical perspective, the prediction methods of the above researches mainly use mathematics, statistics, traditional machine learning and other shallow-model methods to build the prediction model. However, shallow-model methods lack the ability of deep feature extraction and nonlinear function approximation. To solve problem 1, we choose the daily evolved radio access bearer (E-RAB) abnormal release (EAR) ratio of LTE-R core communication services as CQ indicator. EAR ratio can reflect the daily operation status of the communication service (please refer to Section III). Meanwhile, this enables our prediction method to get LTE-R CQ one day in advance, thus allowing sufficient time for active maintenance. To solve problem 2, we proposed a multivariate time series prediction method based on deep learning. This method can fully extract temporal and spatial features from historical CQ data to realize precise prediction.

In GSM-R, railway core communication service bearing is realized by circuit-switched. Therefore railway core communication services run on a dedicated line, and other services can not exist at the same time. Different from GSM-R, LTE-R is based on orthogonal frequency division multiple access (OFDMA) technology, so the railway core communication services are carried out through packet switching, which enables other services and railway core communication services can run at the same time [1]. Therefore, in LTE-R, the operation status of railway core communication services has a correlation with the operation status of other communication services. Since we adopt the EAR ratio of LTE-R core communication services as the CQ indicator, when predicting the LTE-R CQ, we should take into account the EAR ratio of other communication services. Furthermore, it has been proved that multivariate prediction can make full use of the internal relations among the variables, which can make up for the lack of prediction information and improve the prediction accuracy [7].

Generally, the time series prediction method predicts the future value by mining the temporal information contained in
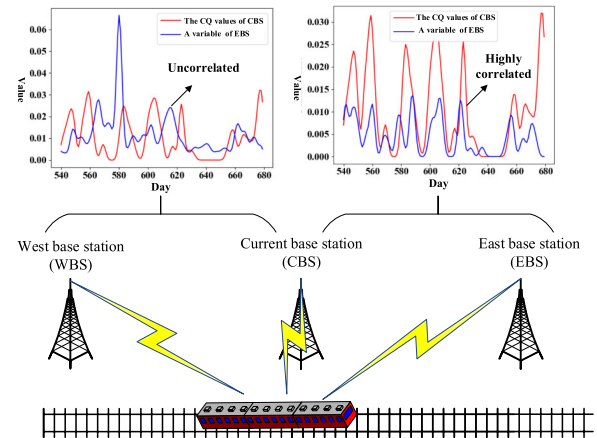

FIGURE 1. The relationship between CQ values and variables.

historical data. However, since LTE-R base stations distribute along the railway, the influence of adjacent base stations should be taken into account when making the LTE-R CQ prediction. As the CQ data of each base station has multiple variables (please refer to Section III), the historical CQ data of related base stations is actually a highly nonlinear and complex multivariate time series. However, among these variables, some may be invalid or redundant. From the theory of pattern recognition, the redundant features could be harmful to the CQ prediction [8]. For the sake of better explanation, we take a real LTE-R CQ dataset collected from Shuohuang railway in China as an example and choose three variables from three adjacent base stations. Figure 1 shows the three base stations and the three variables. Here, the values of variable are processed by the secondary moving average.

In Figure 1, it is obvious that the variable of EBS is correlated with the CQ values of CBS, while the variable of WBS looks uncorrelated with the CQ values of CBS. The variable of EBS contributes more than the variable of WBS when using them to build the CQ prediction model. In a sense, the variable of WBS will reduce, rather than improving the CQ prediction performance. Therefore, to accurately predict LTE-R CQ, we need to achieve the following goals: 1) We should pick up variables from current and adjacent base stations which are supposed to be helpful for the prediction of LTE-R CQ, and 2) Building a model between the historical sequences of the selected variables and the CQ value in the next time interval. For 1), there are many methods to evaluate the correlation between the variables of adjacent base stations and CQ values [9]–[11]. However, as mentioned in [8], we should not only focus on the correlation between selected variables and the target variable, but also minimize the redundancy existing among the selected variables. For 2), many techniques [12]–[15] can achieve this goal. However, deep learning-based methods work better than traditional shallow-model methods in extracting the deep features and approaching the complex nonlinear model infinitely. In many real-world applications, deep learning technologies perform well in dealing with multivariate time series prediction problems [16]–[19].

Given the existing LTE CQ prediction methods can not support the active maintenance of LTE-R base station, we regard LTE-R CQ prediction as a multivariate time series prediction problem, and propose a temporal-spatial collaborative prediction approach based on deep learning. In this approach, the daily E-RAB abnormal release ratio of LTE-R core communication service is selected as the CQ indicator, and to make a precise prediction, we consider not only the temporal information in the time series of CQ data, but also the regional impact of the adjacent base stations, i.e., spatial information. Firstly, the proposed approach conducts a variable filter method based on the max-relevance and min-redundancy (MRMR) criterion and binary particle swarm optimization (BPSO). Secondly, to better extract the temporal-spatial features, inspired by [20], we design a network structure based on long short-term memory (LSTM) and convolutional neural network (CNN) to extract the heterogeneous temporal-spatial features from selected variables. Moreover, we introduce an attention mechanism to weigh the extracted features and quantify the impact of different features on the results. With these features, we build a collaborative prediction model for CQ prediction. Finally, our approach is applied to the real dataset of Shuohuang railway, and the experimental results show that our approach can accurately predict the CQ of LTE-R base station and support the active maintenance of LTE-R base station.

The main contributions of this paper can be highlighted as follows.

1) We analyzed why the existing LTE CQ prediction methods can not support the active maintenance of LTE-R base station. Based on the analysis results, we adopt EAR ratio of LTE-R core communication service as CQ indicator, and put forward to utilize a deep learning-based multivariate time series prediction method to predict the LTE-R CQ.

2) We develop a temporal-spatial collaborative multivariate time series prediction approach for LTE-R CQ based on deep learning. This approach can select a variable set which can satisfy MRMR criterion among candidate variables, and efficiently mine the temporal and spatial information contained in selected variables. Moreover, we introduce the attention mechanism to improve prediction performance. Compared with the existing multivariate time series prediction methods, the experimental results demonstrate the significant benefits of our approach.

3) To illustrate the function of our prediction approach to the active maintenance of the LTE-R base station, we introduce how to decide whether to trigger the active maintenance based on the prediction results of our approach.

The rest of the paper is organized as follows. Section II investigates related work. Section III describes preliminaries, and Section IV presents our approach for the prediction of LTE-R CQ, including data preprocessing, variable filter, a temporal-spatial collaborative prediction model and model training. Experiments and discussion are described in Section V. Section VI concludes the work.

II. RELATED WORK
A. LTE COMMUNICATION QUALITY PREDICTION

Effective LTE communication quality (CQ) prediction is a fundamental building block for reliable communication. Numerous LTE CQ prediction methods have been proposed in this field. Most of these prediction methods are realized by time series prediction. Some researches use a single variable time prediction method to realize CQ prediction. Zheng et al. [4] presented a method for channel quality indicator (CQI) prediction based on the autoregressive integrated moving average (ARIMA) and Kalman filter. Simulation results show a good performance. Chen et al. [5] proposed a novel algorithm called transmission control protocol (TCP) Westwood with Holt-Winters (TCPWH) to predict the bandwidth in LTE. Lema et al. [6] analyzed the factors which can influence the CQI of LTE-A with carrier aggregation, and used the signal to interference plus noise ratio (SINR) to represent CQI. Then, they utilized cubic spline extrapolation to obtain a prediction horizon that allows extending the reliability of the channel quality evaluation along time. Some researches constructed a multivariate time series prediction model through comprehensive consideration of various factors related to the CQ indicator. Daroczy et al. [2] focused on the session drop prediction in the LTE network. They presented a novel machine learning method based on dynamic time warping (DTW) and AdaBoost. The experiment results showed that their method could predict the session drop with higher accuracy. Yue et al. [3] developed a machine learning-based prediction framework called LinkForecast. LinkForecast can select the most important features, and use the selected features to predict the link bandwidth in real-time.

The aforesaid researches performed well in LTE CQ prediction. However, the reason why those researches can not satisfy the CQ prediction and active maintenance of LTE-R is three-fold. First, they are all short-term predictions, which can not provide sufficient buffer time for active maintenance. Second, the CQ indicators of these researches can not reflect the operation status of LTE-R core communication services, which is very important for the normal operation of railway. Third, the aforesaid researches adopt shallow-model methods, which lack the ability of deep features extraction and nonlinear function approximation. In our work, we choose daily E-RAB abnormal release ratio of LTE-R core communication services as the CQ indicator, and propose a deep learning-based multivariate time series prediction method to predict the LTE-R CQ.

B. MULTIVARIATE TIME SERIES PREDICTION

In this paper, the LTE-R CQ prediction is regarded as a problem of multivariate time series prediction. Nowadays, many research works have been carried out to solve this

problem, which can be divided into two categories: shallow-model methods and deep-learning methods.

Shallow-model methods mainly use mathematics, statistics and traditional machine learning methods to establish a regression model. Kim [12] applied a support vector machine (SVM) to predict the stock price index. They selected 12 related attributes as input of SVM, and their experimental results showed that SVM could provide a promising alternative to stock market prediction. Based on the theory of maximum Lyapunov exponent, Zhang [13] selected multiple adjacent reference points to realize multivariate time series prediction. To forecast the price of stock index, Sun *et al.* [14] utilized the fuzzy C-means to preprocess the data. Then they introduced the rough set algorithm to establish a fuzzy logic relation group, which can realize multivariate time series prediction. Mao *et al.* [15] proposed a multivariate time series prediction method through outlier sequences elimination. They first divided the known sequences by using hierarchical clustering based on fuzzy entropy. Then they proposed an outlier sequence elimination method based on the principal curve. Finally, the multi-dimensional support vector regression (M-SVR) was used to construct the prediction model. The above shallow-model methods performed well in the field of multivariate time series prediction. However, there is still room for improvement in feature extraction and nonlinear approximation of these methods.

In recent years, the method based on deep neural network (DNN) has fully demonstrated its ability in generating complex non-linear models. Besides, many studies showed that RNN, especially long short-term memory (LSTM), had advantages in the sequence data processing. Hence, when dealing with multivariate time series prediction problems, many researchers have used RNN and LSTM as the main structure to construct DNN. Due to the variability of residents' activities, individual residential loads are usually too volatile to forecast accurately. Kong *et al.* [17] proposed an LSTM-based deep learning forecasting framework with appliance consumption sequences. To further improve the ability of feature extraction, Du *et al.* [18] proposed a deep neural network structure composed of CNN and LSTM. Besides, they verified the feasibility and practicability of the model to forecast the PM2.5 concentration. Li *et al.* [19] considered the attention-distracted problem. To solve this problem, they introduced the attention mechanism and proposed an evolutionary attention-based LSTM for multivariate time series prediction. Zhao *et al.* [16] proposed a bidirectional LSTM neural network which is composed of many memory units. This neural network fully considered the temporal-spatial correlation in traffic system. Compared with other representative forecast models, the proposed LSTM network can achieve better performance.

In LTE-R CQ prediction problem, we need to consider not only the historical CQ data, but also the influence of adjacent base stations. Therefore, our data is a multivariable time series data with temporal-spatial characteristics. Although the above methods can effectively extract the deep

features, there are still some problems need to be resolved. Firstly, not all variables are beneficial to the prediction of CQ value. We need to find an appropriate variable set before building the prediction model. Secondly, it has been proved that the contextual information has a great influence on the performance of the model [20]. However, the aforesaid methods lack the ability to extract contextual information, which is not conducive to predicting change trend. Thirdly, methods based on RNN and LSTM usually use the time sliding window as input. However, the features in a sliding window have different effects on the results. We need to find a way to identify this difference. In our work, we propose a variable filter method and prove that variable filter can improve the accuracy of prediction. Then, we design a DNN structure based on LSTM and CNN to obtain the contextual information contained in the eigenvectors of selected variables. Besides, attention mechanism is introduced to solve the attention-distracted problem. Finally, the superiority of our approach is demonstrated by the contrast test.

III. PRELIMINARIES
A. COMMUNICATION QUALITY RELATED DATA DESCRIPTION

In this paper, we introduce the LTE evolved radio access bearer (E-RAB) abnormal release ratio to evaluate the communication quality (CQ) of LTE-R base stations. E-RAB abnormal release (EAR) ratio is a key performance indicator of LTE, which reflects the ability of LTE base station to accept services [21]. Through the measurement report function of LTE system, we can get the EAR data of each base station. The EAR data we used in this paper is from Shuohuang railway company (a heavy haul railway in northern China). This data are classified into 6 types according to the quality of service (QoS) class identifier (QCI), and the QCI value ranges from 1 to 6. Different QCI represents different services and the value of QCI indicates the priority of those services, the lower the QCI value, the higher the priority. For example, QCI equals 1 represents this kind of EAR data comes from the most important kind of service. The QCI value and its corresponding services are shown in Table 1.

The EAR data records the daily EAR times, and the daily E-RAB request times of each kind of QCI. Therefore, the daily EAR ratio of each QCI can be calculated by EAR data. The daily EAR ratio of each QCI is as in (1):

$$E_d^q = \frac{R_d^q}{N_d^q}, \quad d = 1, 2, 3, \ldots, D, \ q \in \{1, 2, 3, 4, 5, 6\} \quad (1)$$

where D is the total number of the days or the data size. E_d^q denotes the EAR ratio of QCI q (Those services when QCI value is q) on day d. R_d^q represents the E-RAB abnormal release times of QCI q on day d. N_d^q is the E-RAB request times of QCI q on day d. Through formula (1), we can get the EAR ratio data which consists of 6 variables corresponding to QCI from 1 to 6. QCI equals 1 indicates that this type of service has the highest priority and can directly affect the normal operation of the railway. If the EAR ratio of QCI

TABLE 1. QCI value and its corresponding services.

QCI value	Type of service	Concrete content
1	Train synchronous operation command	Train control, locomotive wireless synchronic operation
2	Data transmission of controllable train tail device	Control train tail device
3	Voice communication	Dispatching communication, railway public affairs communication
4	Text data transmission of traffic control	Dispatching command, wireless train number verification information
5	Text data transmission of common application	Integrated maintenance
6	Image data information	Train-ground mobile video surveillance
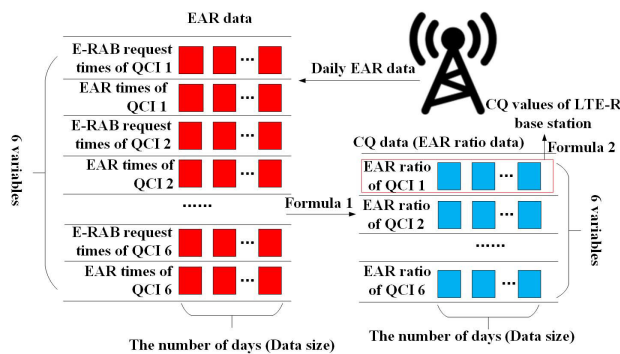
FIGURE 2. The relationship between CQ value, EAR data and CQ data.

1 consistent increase or higher than a certain value, it means this base station should be maintained. Therefore, we choose the EAR ratio of QCI 1 as the CQ value of LTE-R base station. The CQ value of LTE-R base station is as in (2):

$$Q_d = E_d^1, \quad d = 1, 2, 3, \ldots, D \tag{2}$$

In this formula, D is the total number of the days or the data size. Q_d is the CQ value of LTE-R base station on day d, and the CQ value is inversely proportional to the real communication quality. E_d^1 denotes the EAR ratio of QCI 1 on day d.

In addition to the EAR ratio of QCI 1 which directly represents the CQ of LTE-R base station, the EAR ratios of other QCI are correlated with the EAR ratio of QCI 1. Therefore, we keep the whole EAR ratio data of each base station. In this paper, the EAR ratio data is called CQ data. The data used in our approach is the CQ data. The relationship between CQ values, EAR data and CQ data (EAR ratio data) is shown in Figure 2.

B. PROBLEM STATEMENT

The LTE-R CQ prediction can be represented as a multivariate time series prediction problem. As a multivariate time series prediction problem, the CQ prediction of LTE-R base station mainly includes two tasks: one is to find an appropriate variable set, and the other is to build a prediction model between CQ value at the next time interval and the historical data of selected variables.

As mentioned in Section I, when selecting variables, we should consider both temporal and spatial characteristics. Therefore, the candidate variable set should include the variables of current base station and adjacent base stations. Since the CQ data used in this paper is from an east-west railway, each base station has two adjacent base stations: west base station and east base station. The candidate variable set U in this paper is shown in (3):

$$U = U_w \cup U_c \cup U_e \tag{3}$$

where U_w, U_c and U_e are the variable set of west, current and east base stations respectively. The variable set of each base station has 6 variables corresponding to the EAR ratio of QCI 1 to 6. The variable when QCI equals 1 of the current base station represents the CQ value, so it should be reserved as one of the selected variables. For the other 17 variables in U, we conduct a variable filter method to obtain an appropriate variable set which is helpful to predict the CQ value of current base station. The selected variable set U_s is as in (4).

$$U_s = f_s(U, q_v) \cup \{q_v\} \tag{4}$$

In (4), f_s is the variable filter method, and q_v is the variable of CQ value.

After getting U_s, we can build the multivariate time series prediction model. When building the multivariate time series prediction model, the historical sequences of U_s are used as input data. The input data is divided into feature matrix by sliding window. The input data is $X = (X_1, X_2, \ldots, X_T)$, and $X_t = (x_t^1, x_t^2, \ldots, x_t^L)$, where L is the length of time-step, X_t is the sliding-window feature matrix at time t and $X_t \in X$. The corresponding CQ values of current base station at the next time interval, $y = (y_2, y_3, \ldots, y_{T+1})$ are also given. In general, building a multivariate time series prediction model means that we need to learn a nonlinear mapping by using X and y to obtain the predicted CQ value $\widetilde{y}$ as in (5)

$$\widetilde{y} = f(X, y) \tag{5}$$

In (5), $f(\cdot)$ is the objective nonlinear mapping function which we aim to find.

C. ACTIVE MAINTENANCE BASED ON LTE-R CQ PREDICTION

Without CQ prediction, the maintenance of LTE-R base station is periodic, which is not only a passive approach, but also a waste of resources. In order to realize active maintenance, this paper mainly considers the trend of CQ and the CQ value. Once the CQ value of a base station consistent increase or the predicted CQ value approaches or exceeds the threshold, the active maintenance of this base station will be executed. Generally, active maintenance includes parameter
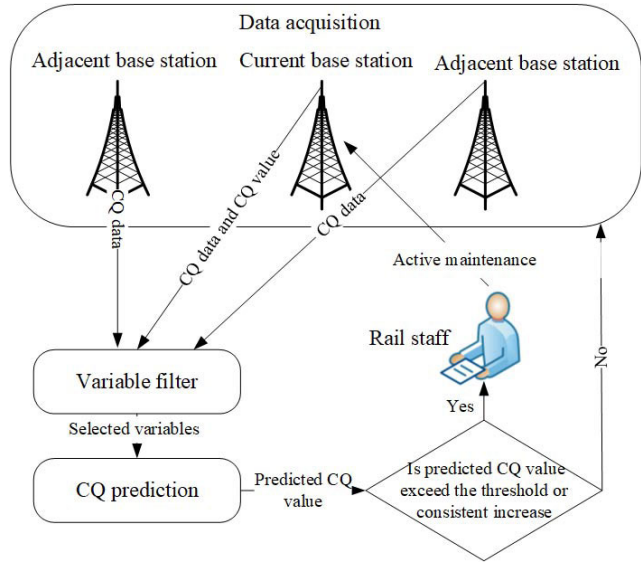
FIGURE 3. The flowchart of active maintenance based on LTE-R CQ prediction.

FIGURE 4. A framework for the prediction of LTE-R communication quality.

adjustment, alarm troubleshooting, interference analysis, etc. The flowchart of active maintenance based on LTE-R CQ prediction is shown in Figure 3.

IV. APPROACH FOR LTE-R COMMUNICATION QUALITY PREDICTION

In this section, we present an approach for LTE-R CQ prediction. Firstly, we need to preprocess the raw CQ data, including data cleaning, double moving average, and data normalization. Secondly, we propose a variable filter method to select an appropriate variable set for prediction. Thirdly, we construct a deep neural network structure composed of LSTM and CNN which can obtain the heterogeneous temporal-spatial features. With these features, we can build a collaborative prediction model. Finally, we train and evaluate the model. A graphical illustration is shown in Figure 4.

A. DATA PREPROCESSING

To get the CQ of LTE-R base station, we should calculate the EAR ratio. However, communication requests are not always existed because of the limited traffic flow of the railway, which leads to some invalid values. Meanwhile, since the CQ of the base station is affected by many factors, including human factors, weather factors, seasonal factors and so on, the raw CQ data fluctuates frequently and the trend of CQ will be difficult to identify. To solve above problems, we preprocess the raw CQ data. The main steps of data preprocessing are as follows:

1) DATA CLEANING

Data cleaning is to eliminate invalid values. For LTE-R base station, no communication request does not mean that the communication quality at this time is 0. Therefore, we need to replace those invalid values. In this paper, we adopt the
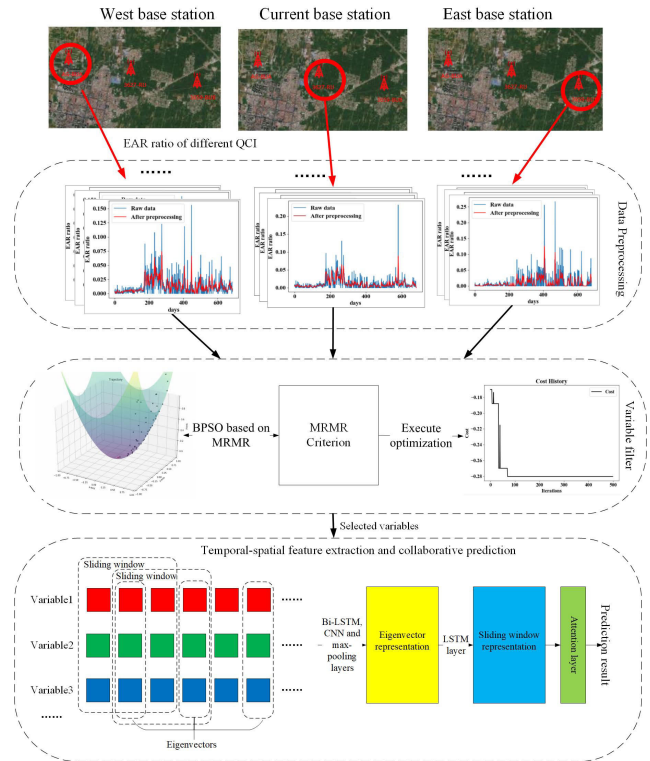
following strategy to eliminate invalid values. In general, we replace the invalid value with the average of two adjacent values. However, if the invalid value only has one adjacent value (e.g. at the beginning or end of data), we will use the adjacent value.

2) DOUBLE MOVING AVERAGE

Due to raw CQ data fluctuates frequently, the trend of communication quality is difficult to identify. Many studies have tried to use a double moving average to preprocess the raw data in time series forecasting [22]–[24]. In our approach, we apply the double moving average to make the data soother. Considering that the time series data is $X = (x_1, x_2, \ldots, x_L)$, where L is the size of data, and the moving window size is 3. The reason why we choose 3 as the window size is that we need the preprocessed data to be as close as possible to the real value while ensuring the trend of change more obvious. Therefore, we only choose the nearest data for averaging. For every x in X, the calculation is as in (6):

$$x_i = \frac{x_{i-1} + x_i + x_{i+1}}{3}, \quad i = 1, 2, 3 \ldots, L \quad (6)$$

As for boundary data, we replace it with the average of itself and its adjacent value. By executing the above process twice, the double moving average is realized. The original data of a base station of Shuohuang railway and the result of the double moving average is shown in Figure 5.
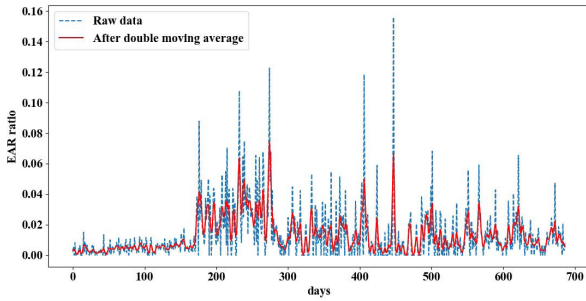
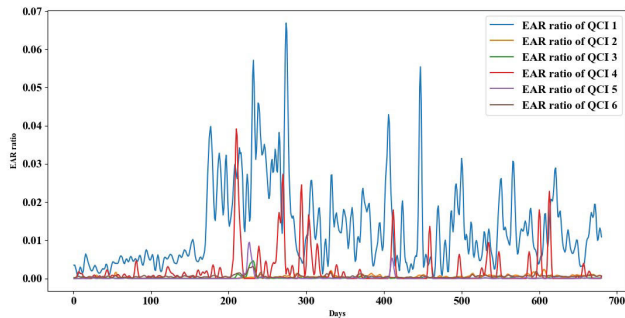**FIGURE 5.** Data after double moving average.

FIGURE 6. The CQ data of the LTE-R base station.

Figure 5 shows that the double moving average can smooth away the small scale fading influence and the trend of CQ will be easier to identify.

3) DATA NORMALIZATION

To better explain the purpose of data normalization, we take the CQ data of a base station in Shuohuang railway for example. As mentioned in Section III, the CQ data of LTE-R base station has 6 variables, and the change of each variable over time is shown in Figure 6.

From Figure 6, we can find that the data range of each variable is quite different. If these data are directly used as training data, it will cause errors while training model. Meanwhile, it is not conducive to the convergence of the model. Therefore, we perform a minimum-maximum normalization to make every variable of the CQ data distribute between 0 and 1. The minimum-maximum normalization is as in (7):

$$x_t = \frac{x_r - x_{min}}{x_{max} - x_{min}} \tag{7}$$

In this formula, x_t is the target data, x_r is the real data. x_{max} is the maximum value of the variable. x_{min} is the minimum value of the variable.

After data preprocessing, the new data will have no invalid value, less noise, and distributed between 0 and 1.

B. VARIABLE FILTER BASED ON MRMR AND BPSO

Before building the prediction model, we need to conduct a variable filter method to get an appropriate variable set as input data. As mentioned in Section III, the CQ data
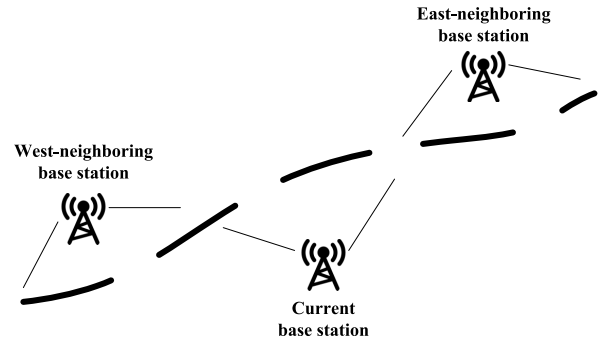

FIGURE 7. The coverage of base station.

have 6 kinds of QCI, it means that each base station has 6 variables. Meanwhile, LTE-R base stations are linearly distributed which means that each base station has two adjacent base stations. They have the overlap coverage area. Therefore, when predicting the CQ of a base station we should consider its two adjacent base stations. The distribution and coverage of the LTE-R base station are shown in Figure 7.

In Figure 7, the dotted line represents the railway, and the solid line shows the coverage of the base station. As mentioned before, each base station has 6 variables, we will have 18 alternative variables to choose from. The variable when QCI equals 1 is the CQ value of the current base station, so it should be reserved as one of the selected variables. For the other 17 variables, we propose a variable filter method based on MRMR criterion and BPSO method to select an appropriate set of variables, which is helpful for the prediction. The details of this method are as follows.

Firstly, due to the difference in traffic flow and traffic load of different base stations, some variables contain a large number of zero and invalid values. We need to remove these variables first. In this paper, if a variable has more than 50 percent invalid data and zero data or the variance is small, this variable will be removed.

Secondly, as mentioned in [8], to get an appropriate set of variables, we need to ensure not only these variables are most relevant to the target variable, but also the correlation between these variables is as small as possible. Furthermore, it has been reported that maximal information coefficient (MIC) can well measure the linear and nonlinear correlation between variables [25]. Therefore, the max-relevance and min-redundancy should satisfy (8) and (9):

$$\begin{cases} \max D(U_a, q_v) \\ D = \dfrac{1}{n} \displaystyle\sum_{i=1}^{n} MIC(u_i, q_v) \end{cases} \tag{8}$$

$$\begin{cases} \min R(U_a) \\ R = \dfrac{1}{C_n^2} \displaystyle\sum_{i=1}^{n-1} \sum_{j=i+1}^{n} MIC(u_i, u_j) \end{cases} \tag{9}$$

In (8) and (9), U_a is the a variable set selected from the remaining 17 variables, n is the number of variables in this

variable set, q_v is the variable of CQ value (target variable), $MIC(u_i, q_v)$ represents the MIC between CQ value and each variable, $MIC(u_i, u_j)$ is the MIC between two variables. Generally, we use addition to integrate formula (8) and (9). The MRMR criterion can be defined as in (10).

$$MRMR = \min[R(U_a) - D(U_a, q_v)] \qquad (10)$$

The purpose of our variable filter method is to obtain a variable set, U_a, which can satisfy (10).

Finally, we utilize the BPSO algorithm to obtain the appropriate variable set, which can satisfy (10). Presently, many other variable filter methods based on MRMR criterion use greedy algorithm to obtain the appropriate variable set [26]–[28]. Different from greedy algorithm, BPSO has a strong global search ability which can help us to find the global optimal solution [29]. In PSO algorithm, each particle (as known as an individual) is a possible solution to the optimization problem. In the search space, each particle has a velocity. The algorithm adjusts the velocity dynamically according to the experience of the particle itself and the shared information of the population [30]. The important steps of the algorithm are the update of particle position and velocity. The velocity and position of $i - th$ particle are shown in (11) and (12):

$$V_i^{t+1} = \omega \cdot V_i^t + c_1 r_1(p^t - P_i^t) + c_2 r_2(g^t - P_i^t) \quad (11)$$
$$P_i^{t+1} = P_i^t + V_i^{t+1} \qquad (12)$$

where V_i^t and P_i^t are the velocity and position of $i - th$ particle at time t, p^t and g^t are the individual optimal position and global optimal position of the whole population at time t, c_1 and c_2 are the learning factors, r_1 and r_2 are the random numbers between 0 and 1, ω is the inertia coefficient. However, for each variable, it only has two states: selected or unselected. Obviously, our solution space is discrete. BPSO is a variant of PSO in solving discrete space optimization problem which is suitable for the variable filter. The particle velocity in BPSO is calculated in the same manner as in PSO. However, particle velocity in BPSO determines the probability of particle position change. The particle position in BPSO can be calculated by (13):

$$P_i^{t+1} = \begin{cases} P_i^t, & r > F(V_i^{t+1}), \\ !P_i^t, & r \leqslant F(V_i^{t+1}), \end{cases} \qquad (13)$$

where r is a random number between 0 and 1, $F(\cdot)$ is a probability function, in this paper the function is *tanh*. Moreover, the MIC between two variables will be calculated in advance to form a MIC matrix. Then, we use the look-up table to get MIC value in each iteration which can greatly improve the speed of BPSO. Finally, formula (10) is used as the optimization objective of our BPSO. Algorithm 1 outlines the variable filter method.

After the variable filter method, we can get a variable set, U_a, which can satisfy formula (10). Based on formula (4), the selected variables which will be input to DNN are composed of U_a and the variable of CQ value.

Algorithm 1 Variable Filter Method

Input:
    The set of variables for selection, U_n;
    The maximum number of iterations, T;
    The target variable, q_v;
    The options of BPSO;

Output:
    A variable set, U_a;

1: Calculate MIC matrix M with input U_n;
2: Initialize the position and velocity of each particle
3: G ← *null*
4: **while** $t \leq T$ **do**
5:     Obtain the global best position P_t;
6:     Obtain the selected variables U_t at t-th iterative with the help of P_t;
7:     Calculate the max-relevance value D_t with the help of U_t and q_v based on formula (8);
8:     Calculate the min-redundancy value R_t with the help of U_t and M based on formula (9);
9:     $g_t \leftarrow R_t - D_t$;
10:    **if** $G \neq null$ **then**
11:      **if** $g_t \leq G$ **then**
12:        G ← g_t;
13:        $U_a \leftarrow U_t$;
14:      **end if**
15:    **else**
16:      G ← g_t;
17:      $U_a \leftarrow U_t$;
18:    **end if**
19: **end while**
20: **return** U_a

C. TEMPORAL-SPATIAL FEATURE EXTRACTION AND COLLABORATIVE PREDICTION

This part focuses on the temporal-spatial features that contained in the selected variables. Because the deep neural network has excellent performance in mining the potential features of data. In this section, inspired by [20], we first construct a deep neural network (DNN) structure composed of LSTM and CNN. This structure can obtain the heterogeneous temporal-spatial features. In fact, when conducting the multivariable time series forecasting, we need to take into account the contextual information of our sequence data, which may contain the changing trend or degree information. Therefore, to better extract the temporal-spatial features, we need to consider the contextual information of eigenvectors, and the temporal information of sliding windows. Moreover, the attention mechanism can calculate a weight for each feature so as to avoid being attention-distracted. To further improve the accuracy of the model, we introduced the self-attention mechanism. Based on the above analysis, we can construct a DNN structure to extract temporal-spatial features, and build a collaborative prediction model. The
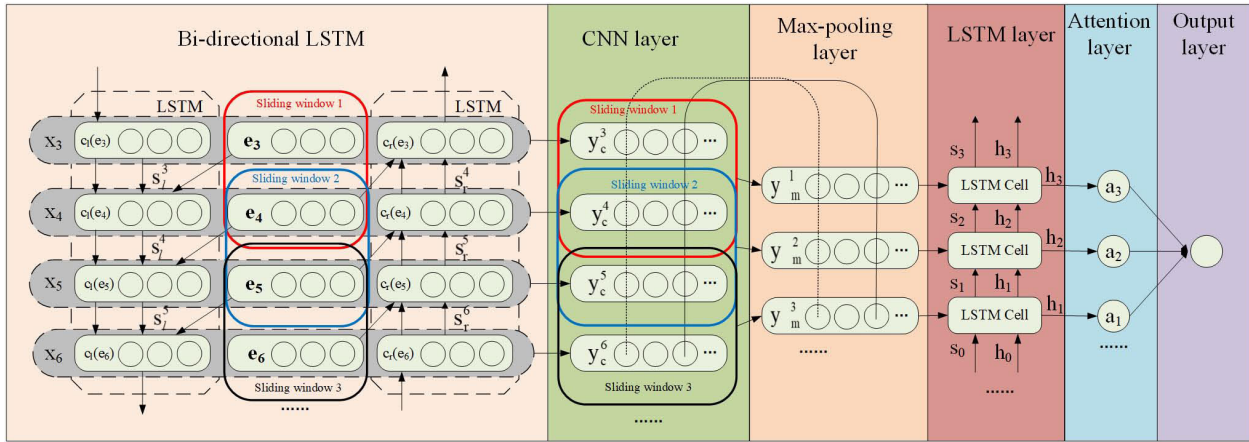
FIGURE 8. The DNN structure of our model. This figure is a partial example of the feature matrix $[e_3, e_4, e_5, e_6, \ldots]$, which is devided by three sliding windows. In this figure, the sliding window size is 2, and the subscript denotes the position of the corresponding eigenvector in the matrix.

structure of our DNN is shown in Figure 8, and the key steps are shown as follows.

1) CAPTURE THE CONTEXTUAL INFORMATION OF EIGENVECTORS

As shown in Figure 4, the selected variable sequences are divided into multiple segments by using sliding window, and each sliding window is composed of eigenvectors. The values of different variables in the same time step form an eigenvector. In order to extract the heterogeneous temporal-spatial features, the first step is to capture the contextual information of eigenvectors. To achieve this goal, we applied a bi-directional LSTM structure to obtain the eigenvector representation. This makes the model can capture the contextual information of eigenvectors.

Assuming that e_i is the current eigenvector at time step i, $c_l(e_i)$ is the contextual information captured from left eigenvectors of e_i, and $c_r(e_i)$ is the contextual information captured from right eigenvectors of e_i. $c_l(e_i)$ and $c_r(e_i)$ are calculated as shown in (14) and (15):

$$c_l(e_i) = \sigma(W_l[c_l(e_{i-1}), e_{i-1}] + b_l) * \tanh(S_l^{i-1}) \quad (14)$$
$$c_r(e_i) = \sigma(W_r[c_r(e_{i+1}), e_{i+1}] + b_r) * \tanh(S_r^{i+1}) \quad (15)$$

In (14) and (15), S is the cell state of LSTM. For more details about the calculation of S, please refer to [31]. σ is a sigmoid method.

As shown in (14) and (15), we capture the contextual information from left and right eigenvectors of e_i. Thus, we can define the representation of eigenvector x_i in (16), which is the concatenation of $c_l(e_i)$, $c_r(e_i)$ and e_i. Formula (16) is as follow:

$$x_i = [c_l(e_i), e_i, c_r(e_i)] \quad (16)$$

In this manner, our approach can capture the contextual information of eigenvectors.

2) CAPTURE TEMPORAL INFORMATION OF SLIDING WINDOWS

To capture temporal information of sliding windows, we begin by using a 1-D CNN layer with a rectified linear unit (RELU) activation function to transform the representation of eigenvectors. Let x_i denotes the representation of eigenvector at $i-th$ time step, and y_c is the output of CNN layer, which wil be send to the next layer. The transformation is as in (17):

$$y_c^i = \text{relu}(W_c x_i + b_c) \quad (17)$$

In (17), W_c and b_c are the weights and bias of the CNN layer. After that, we apply a max-pooling layer which can capture the representative features of a sliding window. Let y_m be the output of the max-pooling layer. The representative features of a sliding window are as in (18):

$$y_m = \max_{i=1}^{n} y_c \quad (18)$$

In (18), *max* function is an element-wise function. The $i-th$ element of y_m is the maximum in the $i-th$ element of y_c. n is the element number of y_c. Because there is temporal information among the sliding windows, we utilize an LSTM layer to further capture this kind of information. The process is as in (19):

$$h_i = \sigma(W_l[h_{i-1}, y_m^i] + b_l) * \tanh(S_i) \quad (19)$$

(19) is obtained by referring to the output gate of LSTM [31]. In (19), y_m^i is the the output of max-pooling layer at $i-th$ step or $i-th$ sliding window. S_i is the current cell state of LSTM. h_{i-1} is the hidden state of the $i-1$ step. W_l and b_l are the weights and bias of the LSTM output gate. h_i is the hidden state and the output of $i-th$ step. Moreover, h_i represents the temporal information of sliding windows.

3) ATTENTION MECHANISM

To solve the attention-distracted problem, we add a self-attention layer after the LSTM layer. According to the

description in [32], assuming that the current time step is t, it is necessary to obtain the output h_t of the last LSTM layer. Then, we can calculate the corresponding feature weight a_i of h_t. The computational process is shown in (20) and (21):

$$e_{ti} = f_{att}(h_t) \tag{20}$$

$$a_i = \frac{exp(e_{ti})}{\sum_{k=1}^{L} exp(e_{tk}) + \varepsilon} \tag{21}$$

In (20) and (21), f_{att} is *tanh* in this paper, L is the length of eigenvector, ε is a very small value, which can avoid dividing by zero.

After the above steps, the result will be input to a single node fully connected layer and output the result of the prediction.

D. TRAIN AND EVALUATE THE MODEL

In this paper, we design a deep neural network (DNN) to make the temporal-spatial collaborative prediction. The purpose of our DNN is to find a non-linear transformation function between the selected variables and the corresponding CQ values at the next time interval. Before training the model, the DNN needs to randomly initialize the weights of the depth network. Then, neural network optimization algorithms like gradient descent (GD), stochastic gradient descent (SGD), adaptive moment estimation (ADAM), etc. are used to adjust the model parameters. Activation functions are used in each layer of DNN to achieve non-linear mapping. In this paper, we use RELU as the activation function and ADAM as the optimization algorithm. Meanwhile, we should consider that if the number of hidden nodes of the layer in a neural network is less, the model can not fully capture the information in data. But, if there are too many hidden nodes in the neural network, it will reduce the learning rate and trap the model into a local minimum. In this paper, the number of hidden nodes of the bidirectional LSTM, which used to capture contextual information is 128, and the number of hidden nodes of the LSTM after the max-pooling layer is 256. In addition, to avoid overfitting, we introduce dropout and L2 regularization at the last LSTM layer. Their parameters are 0.01 and 0.2, respectively. The hidden nodes of the last LSTM layer are also 256.

By using the trained model, we can predict the CQ of the LTE-R base station. In order to evaluate the performance of our approach, the prediction results will be compared with RCNN [20], evolutionary attention-based LSTM (EA-LSTM) [19] and CNN-LSTM [18]. Root mean square error (RMSE) and trend accuracy ratio (TAR) are used to evaluate the performance of different prediction models. Let y_{pred}^i be the predicted CQ value on day i, y_{real}^i be the real CQ value on day i, and n is the total number of days. RMSE is as in (22):

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(y_{pred}^i - y_{real}^i)}{n}} \tag{22}$$

When training the model, (22) is the objective function. TAR is as in (23):

$$TAR = \frac{\sum_{i=1}^{n} x^i}{n} \tag{23}$$

In (23), the calculation of x^i is shown in (24)

$$x^i = \begin{cases} 1, (y_{pred}^i - y_{pred}^{i-1}) \times (y_{real}^i - y_{real}^{i-1}) \geq 0 \\ 0, (y_{pred}^i - y_{pred}^{i-1}) \times (y_{real}^i - y_{real}^{i-1}) < 0 \end{cases} \tag{24}$$

In (24), if the value of x^i equals 1, it means that the prediction of the trend is correct.

V. EXPERIMENT AND DISCUSSION

A. DATA DESCRIPTION

The experimental datasets used in our experiments come from Shuohuang railway. We choose the CQ data from three non-adjacent base stations (BS1, BS2, BS3) and their respective adjacent base stations. The time of those data range from September 2015 to August 2018, 682 days in all.

As mentioned in Section III, we joined the CQ data of each base station and its adjacent base stations, thus, we can get 3 datasets named dataset BS1, dataset BS2 and dataset BS3. Each dataset has 18 variables. For ease of exposition, the naming rules of 18 variables for each base station are as follows. For the current base station, we name the variable according to the QCI value. e.g. "qc11" represents the EAR ratio of current base station when QCI equals 1. For two adjacent base stations, considering that Shuohuang railway is an east-west railway, when naming variables, not only the QCI value but also the relative position of the base station should be considered. Thus, the variables of the west-neighboring base station all have "W-" prefix, and the variables of the east-neighboring base station all have "E-" prefix, e.g. "W-qci1" represents the EAR ratio of west-neighboring base station when QCI equals 1, and "E-qci1" represents the EAR ratio of east-neighboring base station when QCI equals 1. Furthermore, variable "qci1" is the CQ value and the CQ value at the next time interval is the prediction target.

For each dataset, 60 percent of data is training set and 40 percent of data is test set. When building and testing the model, we use the previous 6 time steps (i.e. the sliding window size is 6) to predict the CQ value at the next time interval for each base station. The sliding window size is confirmed by grid search in search space [1,20] with a step of 1.

B. EXPERIMENTAL ENVIRONMENT

Python 3.5 is used to program the method. All the experiments were carried out on a laptop, which was configured as follows: CPU is i5-8400, memory is 8G, and the graphics card is NIVIDA 1050Ti 4G.
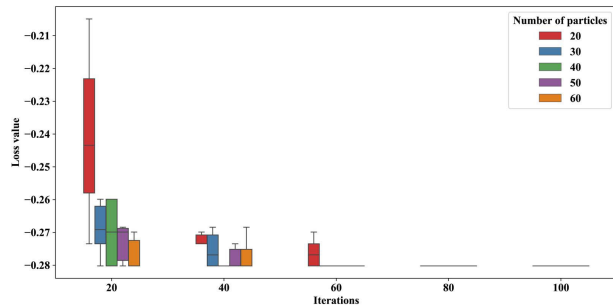
C. VARIABLE FILTER

Before predicting the CQ value, we need to select an appropriate variable set, which is helpful for CQ prediction.
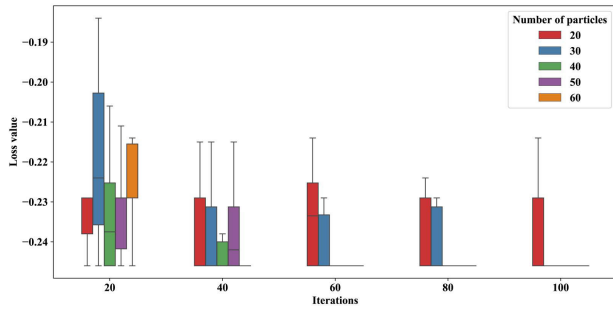
(a) Grid search results on dataset BS1
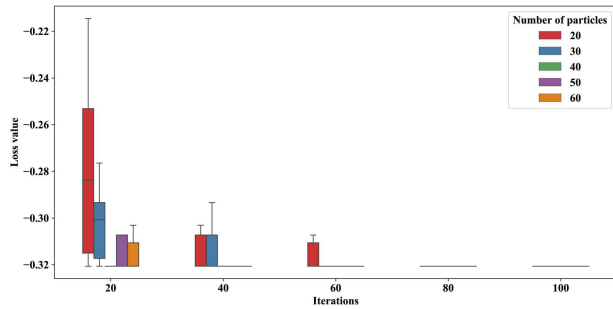
(b) Grid search results on dataset BS2

(c) Grid search results on dataset BS3

FIGURE 9. Grid search results on three datasets.

We have three datasets from three non-adjacent base stations (BS1, BS2, BS3). As mentioned in Section IV.B, for each base station, we have 17 variables to choose from. In this paper, we use BPSO to find a set of variables that can satisfy the MRMR criterion from these 17 variables. Therefore, the loss value is calculated by formula (10). The parameters of BPSO include two acceleration constants (c_1, c_2) and one inertia weight (w). As mentioned in [33], early experience indicates that the acceleration constants c_1 and c_2 each equal to 2.0 for almost all applications, and w is often range from 0.4 to 0.9. To further determine the appropriate values of these parameters, we conduct a random search based on [34]. After considering the recommended settings in [33], we set the search range for c_1 and c_2 is 0 to 4, and the search range for w is 0.3 to 1. The results of random search on three datasets are shown in Table 2.

In addition to the above three parameters, the number of particles N, and the number of iterations T directly affect the results of BPSO. Generally, the recommended number

TABLE 2. The results of random search on three datasets.

Datasets	c_1	c_2	w
Dataset BS1	1.7235	2.8981	0.7031
Dataset BS2	1.1553	3.0086	0.8031
Dataset BS3	3.2763	1.4105	0.7117

TABLE 3. The results of variable filter on three datasets.

Datasets	Result variable set	MIC	Final loss
Dataset BS1	qci5 W-qci5	0.3307 0.2931	-0.2456
Dataset BS2	W-qci5 E-qci1	0.3233 0.4798	-0.3209
Dataset BS3	qci3 E-qci5	0.3882 0.3245	-0.2802

of particles is between 20 and 50 [33]. To approximate the best performance of BPSO, we conducted a grid search over $N \in \{20, 30, 40, 50, 60\}$, $T \in \{20, 40, 60, 80, 100\}$. During grid search, to ensure the stability of BPSO, we run ten times for each combination of parameters. The grid search results on three datasets are shown in Figure 9.
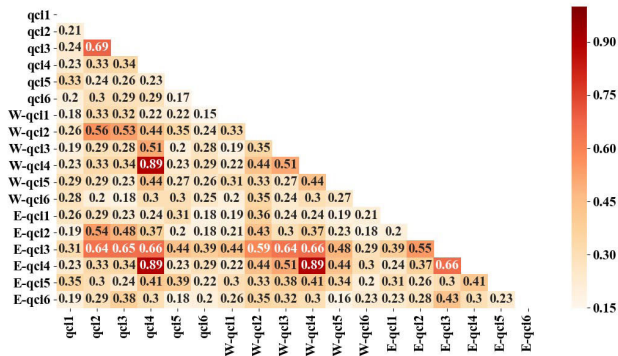
Figure 9 shows that the parameters N and T can significantly affect the results of BPSO. Moreover, with the increase of N and T, the loss values of BPSO on three datasets gradually decrease and eventually stabilize. However, the increase of N and T will lead to longer running time. Therefore, we need to find a suitable set of N and T to ensure the efficiency and stability of BPSO at the same time. As shown in Figure 9, when N equals 40 and T equals 60, the experimental results can meet the above requirements.

The above experiments show that the loss value will eventually stabilize with the iteration. This proved that our variable filter method can find an appropriate variable set for each dataset, which has the max-relevance and min-redundancy. The results of variable filter on three datasets are shown in Table 3.
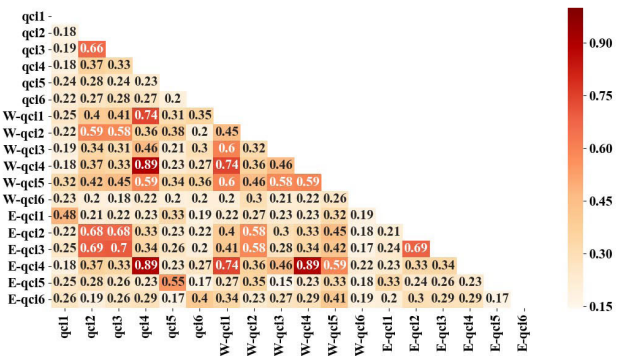
To better illustrate our variable filter method, the maximal information coefficient (MIC) between variables of each dataset are shown in Figure 10.

As can be seen from Figure 10 and Table 3, the variables in result variable set are all highly correlated with CQ value (variable "qci1"), and the MIC values between each variable are relatively small. This proved that our variable filter method can pick up an appropriate variable set for each dataset, which can satisfy the MRMR criterion.
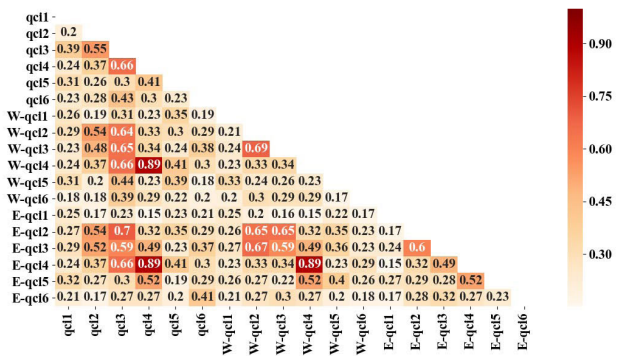
Based on formula (4) and (5), the result of variable filter will be input into DNN together with CQ value ("qci1"), while the CQ value at the next time interval is the prediction target. For example, when training the model on dataset BS1, based on Table 3, the input variables should include "qci1", "qci5" and "W-qci5". Besides, the prediction target is the value of "qci1" at the next time interval. To prove that our variable filter method can improve the accuracy of prediction,

(a) The MIC between variables of dataset BS1

(b) The MIC between variables of dataset BS2

(c) The MIC between variables of dataset BS3

FIGURE 10. The maximal information coefficient (MIC) between variables of experimental datasets.

TABLE 4. The experiment results with different input variables.

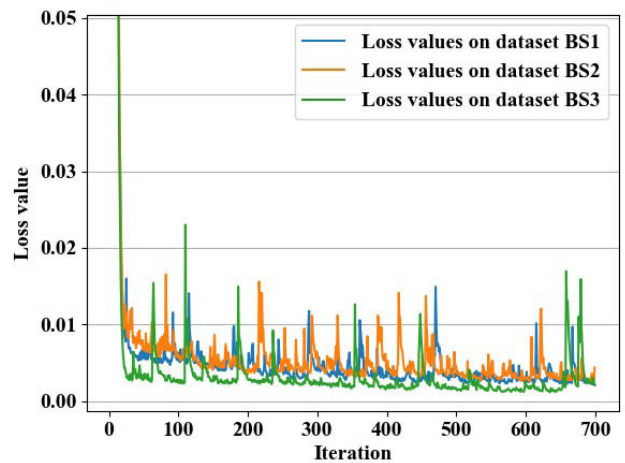|		Single variable	Selected variables	All variables
Dataset BS1	RMSE	0.0029	0.0018	0.0057
	TAR	0.7649	0.8085	0.6879
Dataset BS2	RMSE	0.0032	0.0015	0.0129
	TAR	0.7255	0.7857	0.6203
Dataset BS3	RMSE	0.0034	0.0019	0.0082
	TAR	0.7556	0.7819	0.6917

FIGURE 11. The loss values when training the model on three datasets.

selected variables and the sliding window. The output is the CQ value at the next time interval, and the loss value of the model is expressed by RMSE. Figure 11 shows the loss values when training the model on three datasets respectively.

From Figure 11, it can be seen that the loss value will not continue to decline after 600 iterations. Therefore, in this paper, we set the number of iterations to 600.
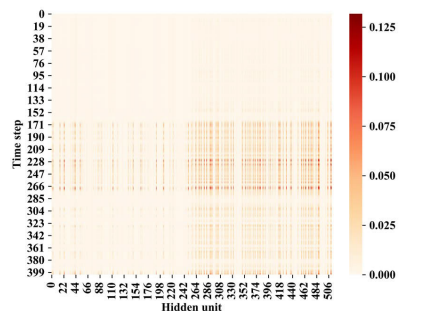
1) TRAIN THE ATTENTION LAYER
The attention mechanism can measure the effect of different features on output, and calculate the attention weights. After training, each feature will get an attention weight. The attention weights are shown in a heat map, as in Figure 12.
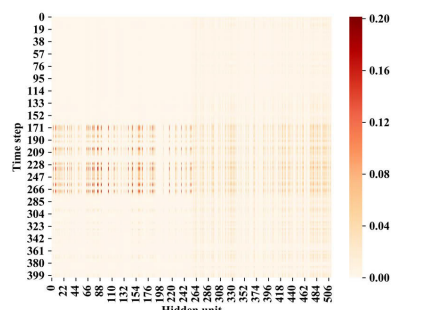
As shown in Figure 12, when training the model with BS1, BS2, and BS3, respectively, the attention weight of each feature can be obtained. Therefore, by introducing the attention mechanism, the effect of each feature on the output can be measured.

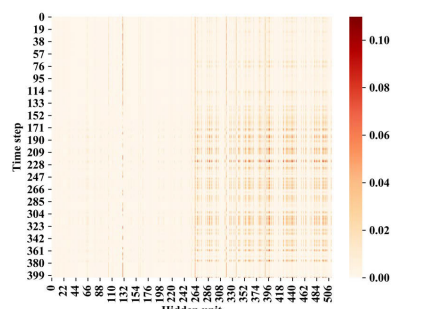### 2) COMMUNICATION QUALITY PREDICTION
Because the initialization of the network is random and the final loss value of each training is different, the prediction results of the model are different after each training. To obtain more stable results, we get ten models by training the model ten times and regard the average of the prediction results as the final result. The prediction results of our approach on experimental datasets are shown in Figure 13:

we contrast the MSE and TAR of our approach when the input data is single variable (only use CQ value, namely "qci1"), selected variables and all variables. The results of the comparison are shown in Table 4.

From Table 4, it can be seen from these experimental results that the proposed variable filter method plays an essential role in improving prediction accuracy.

D. TRAIN AND EVALUATE THE MODEL
When training the model, the size of the sliding window is 6, the batch size is 32. To obtain an appropriate iterations number, we first set the number of iterations to 700. The input of the model is feature matrices and each matrix is defined by

(a) The attentional distribution of dataset BS1.

(b) The attentional distribution of dataset BS2.

(c) The attentional distribution of dataset BS3.

FIGURE 12. The attentional distribution for each experimental dataset.

Figure 13 shows that the prediction curve of our approach is in good agreement with the actual curve. As shown in Figure 13, our approach can obtain good prediction results both in the peak value of the data and in the region where the data fluctuates sharply. This proves that our approach can effectively recognize the characteristics of data and has higher prediction accuracy.

3) COMPARISON AND DISCUSSION

To evaluate the performance of our approach in predicting communication quality, the prediction results were compared with RCNN, EA-LSTM, and CNN-LSTM. Furthermore, to verify the generality of our approach, we have conducted comparative experiments on three datasets. The CQ prediction results of each method are shown in Figure 14, Figure 15 and Figure 16.

**TABLE 5.** Comparison of experimental results.

Station name	Methods	RMSE	TAR
BS1	CNN-LSTM	0.0067	0.617
	EA-LSTM	0.0014	0.7836
	RCNN	0.0042	0.7425
	Our approach	0.0018	0.8085
BS2	CNN-LSTM	0.0068	0.5472
	EA-LSTM	0.0037	0.7481
	RCNN	0.0088	0.6879
	Our approach	0.0015	0.7857
BS3	CNN-LSTM	0.0126	0.6831
	EA-LSTM	0.0023	0.7632
	RCNN	0.0052	0.7293
	Our approach	0.0019	0.7819

Figure 14, Figure 15 and Figure 16 show the prediction results of CNN-LSTM, EA-LSTM and RCNN on BS1, BS2 and BS3 datasets, respectively. In Figure 14, Figure 15 and Figure 16, the red line represents the predicted value, and the blue dotted line represents the real value. Figure 14 shows the prediction results of CNN-LSTM. As shown in Figure 14, there is obvious deviation between the prediction curve and the real curve. Furthermore, in the peak region of the data, the prediction error is more obvious. The above phenomenon proves that CNN-LSTM can not effectively extract the features of data. As shown in Figure 16, The prediction results of RCNN are better than those of CNN-LSTM. But, the prediction error is still very obvious. In Figure 15, it can be seen that the prediction curve of EA-LSTM is also very close to the real curve. To further prove the superiority of our approach. We compared the RMSE and TAR of each method. The RMSE and TAR of each method are shown in Table 5.

In Table 5, we can figure out that the average RMSE of our approach on BS1, BS2 and BS3 is 0.0017, which is the highest of all methods. The average RMSE of CNN-LSTM, EA-LSTM and RCNN are 0.0087, 0.0025, and 0.0061, respectively. Furthermore, the average TAR of our approach on BS1, BS2 and BS3 is 0.792, which is also the highest of all methods. The average TAR of CNN-LSTM, EA-LSTM, and RCNN are 0.6158, 0.765, and 0.72. In all the experiments, CNN-LSTM has the worst performance.

From the above experiments, it can be seen that the RMSE and TAR of EA-LSTM and our approach are better than other methods. Compared with other methods, they have one thing in common, that is, they all introduced the attention mechanism. This proves that the attention mechanism can improve the accuracy of the model. Besides, when comparing the TAR of EA-LSTM and our approach, our approach is still the best one. This shows that acquiring contextual information enables our approach to better identify the changing trend of data.

From the perspective of active maintenance. TAR indicates whether the current communication quality of the LTE-R base
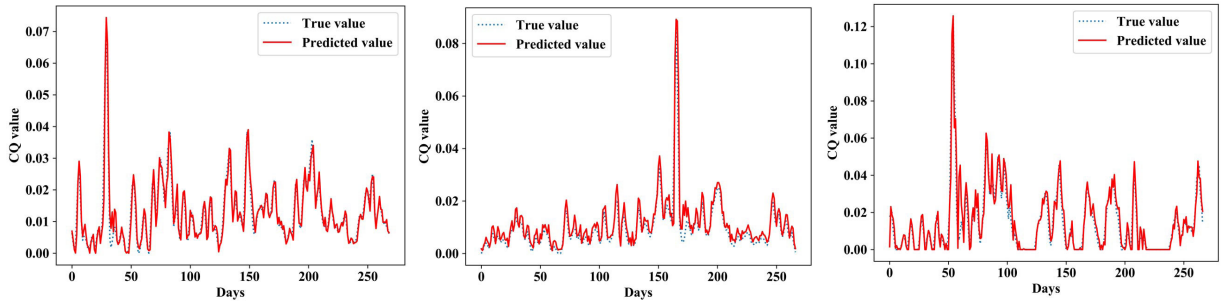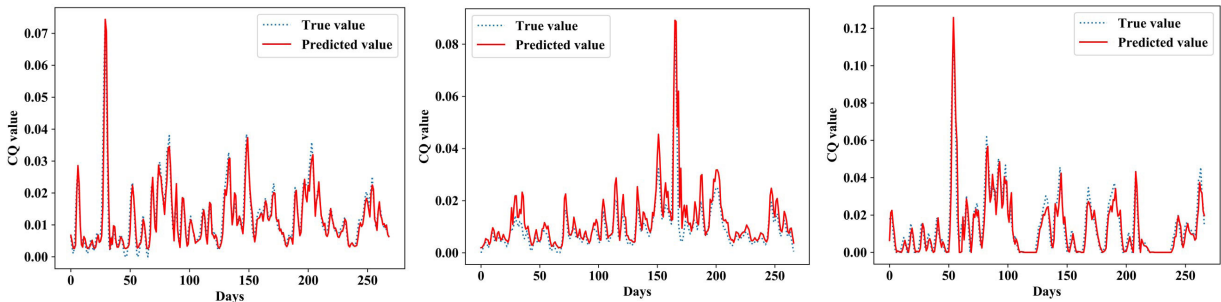
(a) The prediction results on dataset BS1.   (b) The prediction results on dataset BS2.   (c) The prediction results on dataset BS3.

FIGURE 13. The prediction results of our approach.
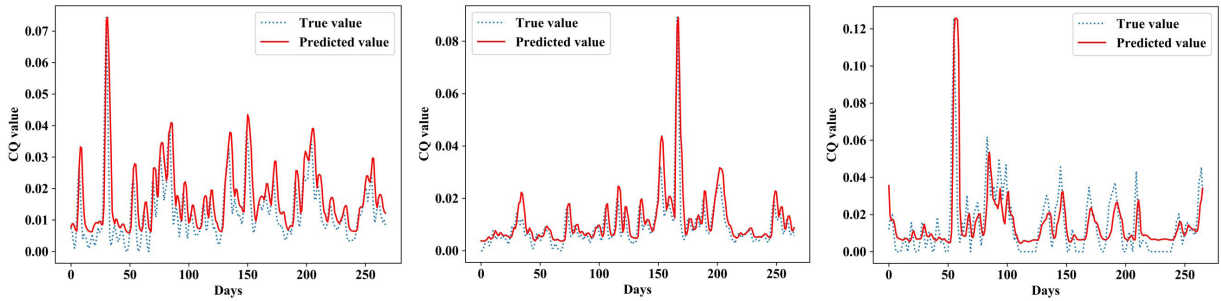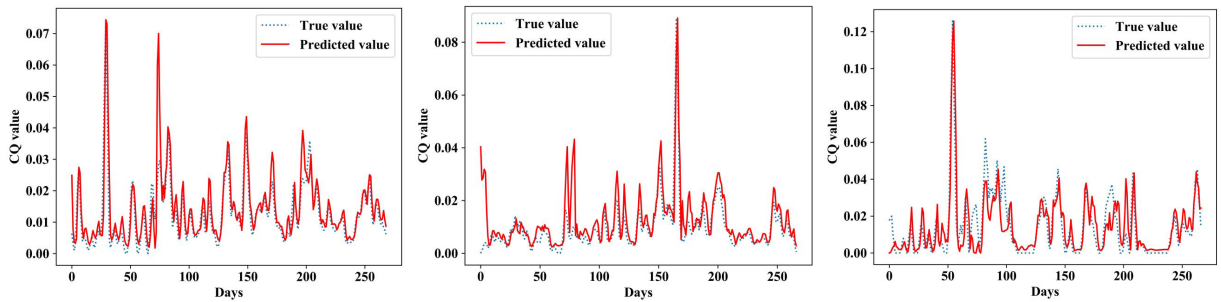

(a) The prediction results on dataset BS1.   (b) The prediction results on dataset BS2.   (c) The prediction results on dataset BS3.

FIGURE 14. The prediction results of CNN-LSTM.

(a) The prediction results on dataset BS1.   (b) The prediction results on dataset BS2.   (c) The prediction results on dataset BS3.

FIGURE 15. The prediction results of EA-LSTM.

(a) The prediction results on dataset BS1.   (b) The prediction results on dataset BS2.   (c) The prediction results on dataset BS3.

FIGURE 16. The prediction results of RCNN.

station will rise or fall, which is of great reference value to the active maintenance of LTE-R base station. Figure 17 shows a part of CQ prediction results on BS1 dataset, and illustrates how to utilize our CQ prediction approach to determine whether the active maintenance will be conducted based on this part of prediction results.

In Figure 17, the threshold is 0.02, the maximum days of CQ value continuous increase is 3, and the sliding window
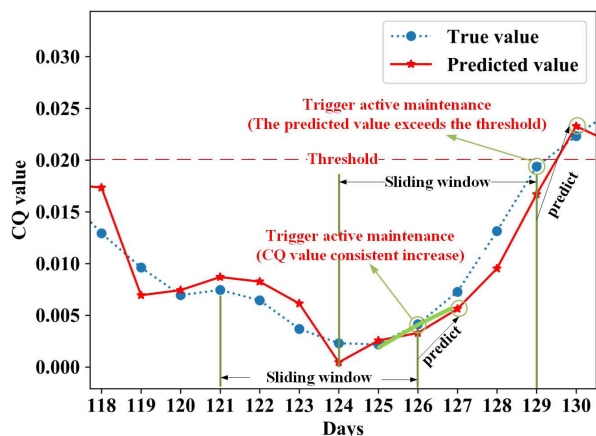
FIGURE 17. Conduct active maintenance based on CQ prediction.

size is 6. As shown in Figure 17, if the CQ value of a LTE-R base station is consistent increase or the CQ value approaches or exceeds the threshold, the active maintenance of this base station will be triggered.

In summary, our approach can accurately predict the CQ of the LTE-R base station and support the active maintenance of LTE-R base station. Although the above experiments achieved good results, our approach is more applicable to busy LTE-R base stations. Only when there are enough E-RAB requests, the CQ indicator defined in this paper can accurately reflect the CQ of the current base station.

VI. CONCLUSION

The prediction of communication quality (CQ) plays an important role in LTE-R system active maintenance, and LTE-R CQ prediction can be regarded as a multivariate time series prediction problem. In this paper, we chose the daily EAR ratio to quantify CQ and proposed a temporal-spatial collaborative prediction approach to predict the CQ of LTE-R base stations. Firstly, we preprocessed the data to make the changing trend of the data more obvious. Secondly, we proposed a variable filter method based on BPSO algorithm and MRMR criterion, which can select an appropriate variable set for CQ prediction. Thirdly, we constructed a deep neural network (DNN) structure. This structure can efficiently extract temporal-spatial features contained in selected variable sequences by considering contextual information of eigenvectors and temporal information of sliding windows. Furthermore, to obtain a more accurate prediction result, we introduced the attention mechanism to solve the attention-distracted problem. Finally, we applied the proposed approach to the real datasets of Shuohuang railway. The experimental results showed that the proposed approach achieved more competitive performance than baseline methods, and based on the LTE-R CQ prediction approach, we could conduct the active maintenance for LTE-R base station. For future work, we will further optimize the net-

work structure and improve the generalization ability of our approach.
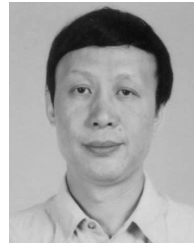
REFERENCES

[1] R. He, B. Ai, G. Wang, K. Guan, Z. Zhong, A. F. Molisch, C. Briso-Rodriguez, and C. P. Oestges, "High-speed railway communications: From GSM-R to LTE-R," *IEEE Veh. Technol. Mag.*, vol. 11, no. 3, pp. 49–58, Sep. 2016.
[2] B. Daroczy, P. Vaderna, and A. Benczur, "Machine learning based session drop prediction in LTE networks and its SON aspects," in *Proc. IEEE 81st Veh. Technol. Conf. (VTC Spring)*, May 2015, pp. 1–5.
[3] C. Yue, R. Jin, K. Suh, Y. Qin, B. Wang, and W. Wei, "LinkForecast: Cellular link bandwidth prediction in LTE networks," *IEEE Trans. Mobile Comput.*, vol. 17, no. 7, pp. 1582–1594, Jul. 2018.
[4] Y. D. Zheng, M. K. Dong, Y. Jin, and J. J. Wu, "CQI prediction for shadow fading in LTE-compatible GEO mobile satellite communications system," *Adv. Mater. Res.*, vols. 756–759, pp. 2137–2141, Sep. 2013.
[5] Z. Chen, Y. Liu, Y. Duan, H. Liu, G. Li, Y. Chen, J. Sun, and X. Zhang, "A novel bandwidth estimation algorithm of TCP westwood in typical LTE scenarios," in *Proc. IEEE/CIC Int. Conf. Commun. China (ICCC)*, Nov. 2015, pp. 1–5.
[6] M. A. Lema, M. García-Lozano, and S. Ruiz, "Cubic spline extrapolation for uplink channel quality prediction in LTE—A with carrier aggregation," *Trans. Emerg. Telecommun. Technol.*, vol. 28, no. 2, p. e2950, Feb. 2017.
[7] Z. Hong, X. Wang, C. Jie, M. Tang, and Y. Guo, "A multivariate short-term traffic flow forecasting method based on wavelet analysis and seasonal time series," *Appl. Intell.*, vol. 48, no. 3, pp. 1–12, 2018.
[8] H. Peng, F. Long, and C. Ding, "Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 8, pp. 1226–1238, Aug. 2005.
[9] J. Wu, Y. Su, Y. Cheng, X. Shao, C. Deng, and C. Liu, "Multi-sensor information fusion for remaining useful life prediction of machining tools by adaptive network based fuzzy inference system," *Appl. Soft Comput.*, vol. 68, pp. 13–23, Jul. 2018.
[10] K. Wang, C. Xu, Y. Zhang, S. Guo, and A. Y. Zomaya, "Robust big data analytics for electricity price forecasting in the smart grid," *IEEE Trans. Big Data*, vol. 5, no. 1, pp. 34–45, Mar. 2019.
[11] H. Lyu, M. Wan, J. Han, R. Liu, and C. Wang, "A filter feature selection method based on the maximal information coefficient and gram-Schmidt orthogonalization for biomedical data mining," *Comput. Biol. Med.*, vol. 89, pp. 264–274, Oct. 2017.
[12] K.-J. Kim, "Financial time series forecasting using support vector machines," *Neurocomputing*, vol. 55, nos. 1–2, pp. 307–319, Sep. 2003.
[13] Y. Zhang, "Predication of multivariable chaotic time series based on maximal Lyapunov exponent," *Acta Phys. Sinica*, vol. 58, no. 2, pp. 756–763, 2009.
[14] B. Sun, H. Guo, H. R. Karimi, Y. Ge, and S. Xiong, "Prediction of stock index futures prices based on fuzzy sets and multivariate fuzzy time series," *Neurocomputing*, vol. 151, pp. 1528–1536, Mar. 2015.
[15] W. T. Mao, M. X. Jiang, L. I. Yuan, and S. G. Zhang, "Structural prediction of multivariate time series through outlier elimination," *Zidonghua Xuebao/acta Automatica Sinica*, vol. 44, no. 4, pp. 619–634, 2018.
[16] Z. Zhao, W. Chen, X. Wu, P. C. Y. Chen, and J. Liu, "LSTM network: A deep learning approach for short-term traffic forecast," *IET Intell. Transp. Syst.*, vol. 11, no. 2, pp. 68–75, Mar. 2017.
[17] W. Kong, Z. Y. Dong, D. J. Hill, F. Luo, and Y. Xu, "Short-term residential load forecasting based on resident behaviour learning," *IEEE Trans. Power Syst.*, vol. 33, no. 1, pp. 1087–1088, Jan. 2018.
[18] S. Du, T. Li, Y. Yang, and S.-J. Horng, "Deep air quality forecasting using hybrid deep learning framework," 2018, *arXiv:1812.04783*. [Online]. Available: http://arxiv.org/abs/1812.04783
[19] Y. Li, Z. Zhu, D. Kong, H. Han, and Y. Zhao, "EA-LSTM: Evolutionary attention-based LSTM for time series prediction," *Knowl.-Based Syst.*, vol. 181, Oct. 2019, Art. no. 104785.
[20] S. Lai, L. Xu, K. Liu, and J. Zhao, "Recurrent convolutional neural networks for text classification," in *Proc. 29th AAAI Conf. Artif. Intell.*, 2015, pp. 2267–2273.
[21] J. Reunanen, J. Salo, and R. Luostari, *LTE Key Performance Indicator Optimization*. Hoboken, NJ, USA: Wiley, 2015, pp. 195–248.
[22] S. J. Taylor, "Stock index and price dynamics in the UK and the US: New evidence from a trading rule and statistical analysis," *Eur. J. Finance*, vol. 6, no. 1, pp. 39–69, Mar. 2000.

[23] W. Dbouk, I. Jamali, and K. Soufani, "The effectiveness of technical trading for Arab stocks," *Emerg. Markets Finance Trade*, vol. 50, no. 4, pp. 5–25, Jul. 2014.

[24] D. P. C. Aristaningrum and F. Nhita, "Planting calendar of potato plants based on rainfall forecasting using double moving average on fuzzy evolutionary algorithm," *Adv. Sci. Lett.*, vol. 22, no. 8, pp. 2052–2056, Aug. 2016.

[25] D. N. Reshef, Y. A. Reshef, H. K. Finucane, S. R. Grossman, G. McVean, P. J. Turnbaugh, E. S. Lander, M. Mitzenmacher, and P. C. Sabeti, "Detecting novel associations in large data sets," *Science*, vol. 334, no. 6062, pp. 1518–1524, Dec. 2011.

[26] K. Zeng, K. She, and X. Niu, "Feature selection with neighborhood entropy-based cooperative game theory," *Comput Intell Neurosci*, vol. 2014, no. 12, 2014, Art. no. 479289.

[27] Y. Liu, J. Yang, Y. Chen, K. Tan, L. Wang, and X. Yan, "Stability analysis of hyperspectral band selection algorithms based on neighborhood rough set theory for classification," *Chemometric Intell. Lab. Syst.*, vol. 169, pp. 35–44, Oct. 2017.

[28] S. Cheriguene, N. Azizi, N. Dey, A. S. Ashour, and A. Ziani, "A new hybrid classifier selection model based on mrmr method and diversity measures," *Int. J. Mach. Learn. Cybern.*, vol. 10, no. 5, pp. 1–16, 2018.

[29] J. Kennedy and R. C. Eberhart, "A discrete binary version of the particle swarm algorithm," in *Proc. IEEE Int. Conf. Syst., Man, Cybern. Comput. Cybern. Simul.*, vol. 5, Oct. 1997, pp. 4104–4108.

[30] R. Eberhart and J. Kennedy, "Particle swarm optimization," in *Proc. IEEE Int. Conf. Neural Netw.*, vol. 4, Nov. 1995, pp. 1942–1948.

[31] F. A. Gers, J. Schmidhuber, and F. Cummins, "Learning to forget: Continual prediction with LSTM," *Neural Comput.*, vol. 12, no. 10, pp. 2451–2471, Oct. 2000.

[32] C. Raffel and D. P. W. Ellis, "Feed-forward networks with attention can solve some long-term memory problems," 2015, *arXiv:1512.08756*. [Online]. Available: https://arxiv.org/abs/1512.08756

[33] Eberhart and Y. Shi, "Particle swarm optimization: Developments, applications and resources," in *Proc. Congr. Evol. Comput.*, May 2002, pp. 81–86.

[34] L. James and V. Miranda, "PySwarms: A research toolkit for particle swarm optimization in Python," *J. Open Source Softw.*, vol. 3, no. 21, p. 433, Jan. 2018, doi: 10.21105/joss.00433.

FENG LIU received the B.S. and M.S. degrees from Beijing Jiaotong University, in 1983 and 1988, respectively, and the Ph.D. degree from the Renmin University of China, in 1997.

He is currently a Professor with the School of Computer and Information Technology, Beijing Jiaotong University. He is also the Director of the Engineering Research Center of Network Management Technology for High-Speed Railway, Ministry of Education. He has been involved in research and engineering in the field of rail transit information technology more than 20 years. He has published more than 60 articles and four books. His research interests include rail transit information technology, software engineering technology, big data for railway, and prognostic and health management (PHM). He is also a Millions of Talent Projects National candidates and he got the first class of technological progress awards, in 1991.

YUXIANG MA received the B.S. degree in network engineering from Henan University, in 2013, and the Ph.D. degree from the Computer Network Information Center, Chinese Academy of Sciences, in 2019.

He is currently an Associate Professor with the School of Computer and Information Engineering, Henan University. His research interests include future internet architecture and technologies, network security, and big data for networking.
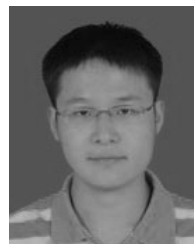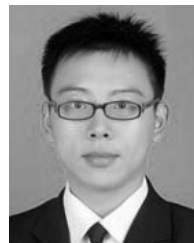
JIANTAO QU received the B.S. and M.S. degrees in computer application technology from Henan University, China, in 2016. He is currently pursuing the Ph.D. degree with the School of Computer and Information Technology, Beijing Jiaotong University.

His research interests include rail transit information technology, prognostic and health management (PHM), and deep learning.

JIAMING FAN received the M.S. degree in computer application technology from Henan University, China, in 2015. He is currently pursuing the Ph.D. degree with the School of Computer and Information Technology, Beijing Jiaotong University.

His research interests include rail transit information technology, prognostic and health management (PHM), and big data for railway.

• • •