

Research Article

Short-Term Traffic Flow Prediction: A Method of Combined Deep Learnings

Chuanxiang Ren ¹, **Chunxu Chai**,¹ **Changchang Yin** ², **Haowei Ji**,¹ **Xuezhen Cheng** ²,
Ge Gao,¹ and **Heng Zhang**¹

¹College of Transportation, Shandong University of Science and Technology, Qingdao 266590, China

²College of Electrical Engineering and Automation, Shandong University of Science and Technology, Qingdao 266590, China

Correspondence should be addressed to Changchang Yin; ycc2009@sdust.edu.cn

Received 25 March 2021; Revised 16 June 2021; Accepted 21 June 2021; Published 5 July 2021

Academic Editor: Erfan Hassannayebi

Copyright © 2021 Chuanxiang Ren 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.

Short-term traffic flow prediction can provide a basis for traffic management and support for travelers to make decisions. Accurate short-term traffic flow prediction also provides necessary conditions for the sustainable development of the traffic environment. Although the application of deep learning methods for traffic flow prediction has achieved good accuracy, the problem of combining multiple deep learning methods to improve the prediction accuracy of a single method still has a margin for in-depth research. In this article, a combined deep learning prediction (CDLP) model including two paralleled single deep learning models, CNN-LSTM-attention model and CNN-GRU-attention model, is established. In the model, a one-dimensional convolutional neural network (1DCNN) is used to extract traffic flow local trend features and RNN variants (LSTM and GRU) with attention mechanism are used to extract long temporal dependencies trend features. Moreover, a dynamic optimal weighted coefficient algorithm (DOWCA) is proposed to calculate the dynamic weights of CNN-LSTM-attention and CNN-GRU-attention with the goal of minimizing the sum of squared errors of the CDLP model. Then, the neuron number, loss function, optimization algorithm, and other parameters of the CDLP model are discussed and set through experiments. Finally, the training set and test set for the CDLP model are established through the processing of traffic flow data collected from the field. The CDLP model is trained and tested, and the prediction results of traffic flow are obtained and analyzed. It indicates that the CDLP model can fit the change trend of traffic flow very well and has better performance. Furthermore, under the same dataset, the results from the CDLP model are compared with baseline models. It is found that the CDLP model has higher prediction accuracy than baseline models.

1. Introduction

With the economic development, the number of motor vehicles in the urban area has increased rapidly, and traffic congestion and traffic accidents have become increasingly serious. In order to mitigate the urban traffic problem, intelligent transportation systems have been widely implemented [1–4]. Among them, short-term traffic flow prediction is one of the core parts of an intelligent transportation system, which provides the basis for traffic management, traffic control, and traffic guidance and also provides support for traveler's decision-making. Prediction of short-term traffic flow has always been a hot topic for scholars in the field of traffic engineering.

For short-term traffic flow forecasting, early research mainly focused on statistical learning methods based on traditional mathematical models. Under the assumption of a certain probability distribution, the parameters of the statistical forecasting model are estimated through theoretical inference, and the model's forecasting results have a better strong explanatory. The traditional methods mainly include Kalman filter models, time series models, and nonparametric regression models.

Okutani and Stephanedes [5] proposed two prediction models based on the Kalman filter theory to predict the traffic flow of streets in Nagoya. In the models, the newest prediction error and the traffic data of multiple adjacent road

sections are considered to improve the prediction accuracy. Xie et al. [6] used the discrete wavelet decomposition method to denoise the traffic flow data and then established a Kalman filter model to predict the traffic flow, which reduced the interference of local noise on the original data and obtained better prediction results. Guo et al. [7] proposed an adaptive Kalman filter model, which uses the adaptive update method of variance to improve the parameters of the model, and verified that the prediction accuracy of the model is better than the traditional Kalman filter model through a large amount of highway traffic data. Emami et al. [8] proposed a fade memory Kalman filter model based on real-time data from the Internet of vehicles and Bluetooth detectors. This model considers the influence of weights and reduces the errors caused by the measurement method. Experiments show that the model can improve the accuracy of the forecast data.

The autoregressive integrated moving average (ARIMA) model is widely used in traffic flow prediction. Ahmed and Cook [9] investigated the ARIMA model in representing freeway time series data and found ARIMA was more accurate than moving average, double-exponential smoothing models. Hamed et al. [10] applied the ARIMA model to forecast traffic volume in urban arterials, and it turned out to be the most adequate model in reproducing all original time series and is computationally tractable. In addition to the ARIMA model, the autoregressive integrated moving average model with explanatory variables, seasonal autoregressive moving average model and other variant structures ARIMA models have also been applied in the field of traffic flow forecasting [11, 12].

The K-nearest neighbors (KNN) method does not require complex prior knowledge and precise function expressions. It has the advantages of a simple algorithm and good portability and has been applied in the field of traffic flow prediction. Zhang et al. [13] used the mean KNN and weighted KNN to establish traffic flow prediction models and comparative analysis was made. Cheng et al. [14] proposed an adaptive spatiotemporal KNN model, which comprehensively considers spatiotemporal weights, time windows, and other parameters, and simulation results demonstrated that the prediction effect of traffic flow has been further improved. The core content of the KNN is to design an appropriate search mechanism, and its prediction results rely on historical data. When the historical data are large, the search efficiency of this method will have a greater impact on the real-time performance of the prediction model.

The basic idea of the support vector machine (SVM) method for traffic flow prediction is to map the original traffic flow data to the high-dimensional feature space through the kernel function and to find the linearly divided plane from the mapping space to solve nonlinear problems in traffic flow data. Yang et al. [15] proposed a short-term traffic flow prediction model based on spatiotemporal correlation and adaptive multicore SVM for the nonlinearity and randomness of traffic flow. Luo et al. [16] used the method of least square SVM to predict the traffic flow, in which a hybrid optimization algorithm is proposed to select

the optimal parameters, and the experimental results show the model can improve the prediction ability and computational efficiency. Tang et al. [17] proposed a traffic flow prediction model that combines denoising schemes and SVM algorithms to improve the prediction accuracy. Results show the model outperforms that without denoising strategy. In addition to the traditional SVM model, variant SVM algorithms, such as seasonal SVM [18], which considers traffic data seasonality, and Online-SVR [19], which deals with special events, have also been applied in traffic flow prediction and good results are obtained.

The development and wide applications of traffic information collection technology, such as inductive detector, geomagnetic detectors, radio frequency identification technology, radar detection, video detection, and floating car detection [20–24], provide a large amount of data for traffic flow prediction. At the same time, with the rapid development of artificial intelligence technology, deep learning, which has powerful data feature mining and nonlinear data fitting capabilities, has been successfully applied in many fields, such as image processing and speech recognition [25–27], and gradually used in traffic parameter forecasting [28–31].

Moreover, the key point of traffic flow forecasting research has also shifted from traditional statistical learning forecasting methods and shallow neural networks [32–34] to deep learning forecasting methods. The shallow neural networks, which only have a single hidden layer, cannot learn the deeper features of traffic flow data and their prediction accuracy is often lower than that of the deep learning network. The deep learning methods have been gradually applied to the field of traffic flow prediction.

Deep belief network (DBN) is an earlier deep learning method used for traffic flow prediction. Huang et al. [35] designed a combined prediction model with unsupervised learning DBN at the bottom layer and multitask learning layer at the top layer for supervised prediction. The multitask learning layer can make full use of the weight sharing in DBN and outperform predicted results. Koesdwiady et al. [36] incorporated weather conditions and traffic flow data into the feature space at the same time and designed a DBN network for unsupervised pretraining, and relevant data from San Francisco are used to conduct experiments to verify the effectiveness of the proposed method. Xu and Jiang [37] proposed a DBN-support vector regression model for short-term traffic flow, in which DBN is used to learn the internal characteristics of traffic flow and support vector regression to predict the traffic flow. Experiments show that the model can effectively predict traffic flow and has fine prediction accuracy. Han and Huang [38] proposed a traffic flow prediction model combining DBN and a kernel extreme learning classifier, in which the internal characteristics of traffic flow data are extracted by DBN and the kernel extreme learner is used to predict traffic flow. Experiments show that the model can improve the accuracy of traffic flow prediction and reduce simulation time.

Convolutional neural network (CNN) is also a typical structure of deep learning. It is a feedforward neural network used to solve data problems similar to a grid structure. It can

accurately extract data features while reducing the complexity of the model. This efficient local feature extraction capability is conducive to better find the spatial correlation between traffic flow data, and then it is widely used in traffic flow prediction [39]. Zhang et al. [40] proposed a short-term traffic flow prediction model based on CNN, in which a spatiotemporal feature selection algorithm determines the optimal input data time lags and amounts of traffic flow data; then, CNN learns these spatiotemporal features. The effectiveness of the model was verified by comparing the prediction results with actual traffic data. An et al. [41] proposed a fuzzy-based CNN traffic flow prediction model, in which the fuzzy approach is used to represent the features of traffic accidents. The experimental results show that the model has superior performance. Liu et al. [42] proposed a CNN-attention model to predict traffic speed. Experimental results show that the model has a great advantage in traffic flow prediction and the impact of different traffic flow temporal and spatial data on traffic flow can be found through visualizing the weights generated by the attention model. Peng et al. [43] proposed a spatial-temporal incidence dynamic graph recurrent CNN to predict urban traffic passenger flow and experiments show that the predictive performance of this network is superior to traditional predictive methods.

LSTM network is a deep learning structure and also a variant of recurrent neural network (RNN). RNN can be applied to the relevant forecasting field of time series data [44]. However, RNN has a problem of the disappearance of the gradient, which can be overcome by LSTM [45]. LSTM has been applied in the field of traffic flow prediction. Ma et al. [46] applied the LSTM to establish a traffic speed prediction model. The results show that the LSTM network effectively captures the time correlation and nonlinearity of the traffic state, and the prediction accuracy is better than most statistics methods. Zhao et al. [47] proposed a traffic forecast model based on LSTM considering temporal-spatial correlation in traffic systems. The results validate that the model can obtain better prediction performance compared with other representative forecast models. Tian et al. [48] proposed a multiscale smoothing method to fill in the missing values in traffic flow data and established an LSTM model to predict traffic flow. Experiments show that the LSTM model has better prediction performance than other prediction methods. Zhao et al. [49] established the LSTM model to predict traffic flow speed and validated that the prediction accuracy is higher than that of the support vector regression prediction method. Wang et al. [50] constructed an LSTM encoding and decoding model based on the attention mechanism for time series prediction, which includes periodic mode and recent time mode. Experiments show that the model is effective and reliable in long-term prediction of time series.

In addition, combination algorithms for traffic flow prediction, especially deep learning algorithms, have received more attention from scholars and produced a series of achievements. Zhou et al. [51] combined LSTM and SVR to build a model for short-term traffic flow prediction, in which a genetic algorithm is used to optimize the parameters of

SVR. The results indicate that the prediction model has higher accuracy than LSTM and CNN. Zhang et al. [52] proposed a model for short-term traffic forecasting, which integrates a graph convolution operator and a residual LSTM structure. The model is evaluated on a traffic speed dataset and better prediction results than six baselines are obtained. Li et al. [53] developed a deep learning-based method, including CNN and LSTM, for real-time movement-based traffic volume prediction at signalized intersections. In the model, CNN is applied to learn the spatial features of traffic volume and LSTM to learn the temporal dependencies. Xia et al. [54] proposed a distributed LSTM weighted model combined with a time window and normal distribution to enhance the prediction capability for traffic flow. Furthermore, the experimental results indicate that the model achieved accuracy improvement.

In summary, the deep learning methods have been widely applied to short-term traffic flow prediction and achieved series of results. Moreover, from the above literature researches, it can be found that the combination of multiple deep learning methods, such as a combination of CNN and LSTM, can improve the performance of the prediction model. LSTM is a variation of RNN, which can obtain the time series characteristics of traffic flow. Meanwhile, there is another variant of RNN, namely GRU, which can also obtain the time series characteristics of traffic flow and make traffic flow prediction [55, 56]. The combined model of LSTM and GRU is used to predict traffic flow parameters, which has been discussed and applied in [57, 58], and its outstanding performance in both prediction accuracy and stability has been proved. In the two works of literature, LSTM and GRU are serial structures. LSTM is firstly used to learn the spatial-temporal characteristics of data, and then GRU is used to predict traffic parameters or LSTM is firstly used to predict value and then encoder with GRUs further captures the relationship between the input sequence and the output sequence. However, the sequential combination structure of LSTM and GRU does not simultaneously use the advantages of the two to complement each other, and it also lacks CNN's guidance on the local trend of traffic flow. It is necessary to apply the combination of three deep learning methods to study the prediction of traffic flow. In addition, the attention mechanism theory [59] has the function of improving the data extraction capabilities of deep learning by imitating human vision to assign weights to data features and has been widely used in image processing and speech recognition [60–63]. Applying it to CNN, LSTM, and GRU deep learnings for traffic flow prediction is also worthy of discussion.

In this article, a DOWCA is presented, and a combined prediction model with CNN, LSTM, GRU, and attention mechanism for short-term traffic flow is proposed and discussed. The main contributions of this study are as follows:

- (1) In order to build a combined traffic flow prediction model, a dynamic optimal weighted coefficient algorithm (DOWCA), is proposed, in which the weights of each single prediction method are

calculated dynamically following new prediction results added.

- (2) A combined deep learning model for short-term traffic flow prediction, namely CDLP, is established based on the CNN, LSTM, GRU, and attention mechanism, which includes paralleled CNN-LSTM-attention model and CNN-GRU-attention model. In CDLP, the dynamic weights for the two single models are calculated by DOWCA.
- (3) After parameter setting through experiment comparison and analysis, the CDLP model is trained and tested using traffic flow data from the field. The results indicate that the CDLP model outperforms baseline models.

The rest of the article is organized as follows. In Section 2, the methodologies of CNN, LSTM, GRU, and attention mechanism are introduced. In Section 3, a DOWCA is proposed and the CDLP model is constructed. In Section 4, the experiment results and analysis are presented. Finally, a brief conclusion and recommendations for future work are presented in Section 5.

2. Methodology

2.1. CNN. CNN is a feedforward neural network with a deep structure and mainly composed of convolution layer, pooling layer, and full connection layer [64]. Among them, the convolutional layer is the most important part of CNN, which uses the convolution kernel to carry out a convolutional calculation for data from the input layer and outputs the convolutional characteristics of the data. If the CNN model contains multiple convolutional layers, then the number of output characteristic parameters by the convolutional layer is large. In order to reduce the number of parameters, the pooling layer is often used to carry out subsampling operations on the convolutional features of the data to extract part of the information and prevent the model from overfitting. The fully connected layer is usually used at the end of the CNN model to reduce unnecessary feature loss, in which all features are integrated and calculated as the final output.

2.2. LSTM Network. LSTM is a variant structure of RNN, which can solve the problem of gradient disappearance and gradient explosion in RNN and can better realize the prediction of time series sequence. The LSTM network is composed of a series of basic cells. The basic cell structure is shown in Figure 1, which includes three gate structures: input gate, output gate, and forget gate.

The orange lines in Figure 1 represent the input gate. The main function of the input gate is to control the input process of all information at time t . The information input process mainly includes two parts. One part is the process of updating the current time information through the tanh function to obtain a new state vector, and the other part is superimposing the current input and the output information of the hidden layer at the

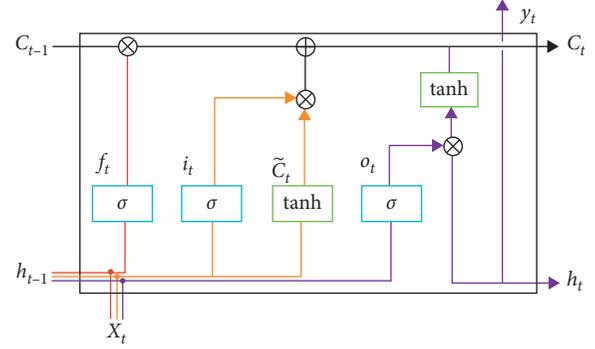


FIGURE 1: The basic unit structure of LSTM.

previous time through the sigmoid function. The specific implementation process can be expressed as follows:

$$\begin{aligned} i_t &= \sigma(W_i h_{t-1} + U_i x_t + b_i), \\ \tilde{C}_t &= \tanh(W_c h_{t-1} + U_c x_t + b_c), \end{aligned} \quad (1)$$

where W_i , W_c , U_i , and U_c are the weights of the input gates; b_i and b_c are the biases of the input gates; and σ and \tanh are activation function, and their formulas are as follows:

$$\begin{aligned} \tanh(x) &= \frac{e^x - e^{-x}}{e^x + e^{-x}}, \\ \sigma(x) &= \frac{1}{1 + e^{-x}}. \end{aligned} \quad (2)$$

The red lines in Figure 1 represent the forget gate, whose main function is to determine the redundant information to be discarded in the unit. The input of the forget gate includes input X_t and output h_{t-1} of the unit at the previous time. The output process is shown in formula (3).

$$f_t = \sigma(W_f h_{t-1} + U_f x_t + b_f), \quad (3)$$

where W_f and U_f are the weight of the forget gate and b_f represents the bias of the forget gate.

The forget gate uses the sigmoid function to superimpose the input values X_t and h_{t-1} , and the output value is limited to the range of $[0, 1]$; finally, the output value is multiplied by the output unit state C_{t-1} at the previous moment. When the output value is 0, it means that the information will be completely discarded. When the output value is 1, it means that the information will be completely retained.

The output information of the forget gate and the input gate is, respectively, multiplied and superimposed on each other to obtain the current unit output state. The specific calculation process is as follows:

$$C_t = f_t \otimes C_{t-1} + i_t \otimes \tilde{C}_t. \quad (4)$$

It can be seen from this formula that C_t represents the long-term memory of all historical information at the current moment.

The purple lines in Figure 1 represent the output gate. The output gate determines the output result of the entire basic cell, which is related to the cell output state C_t at the

current moment. First, use the sigmoid function to process part of the information of the input unit to obtain the output O_t of the output gate and then use the tanh function to process the information in C_t . After the two sets of processed information are multiplied, the final output h_t is obtained. The specific calculation formula is as follows:

$$\begin{aligned} O_t &= \sigma(W_o h_{t-1} + U_o x_t + b_o), \\ h_t &= O_t \otimes \tanh(C_t). \end{aligned} \quad (5)$$

2.3. GRU Network. Similar to LSTM, GRU is also a variant structure of the RNN algorithm, and it also has the function of dealing with the problem of gradient disappearance in RNN and ineffective long-term sequence memory. Compared with LSTM, GRU reduces the complexity of the structure by reducing the gates in the architecture. The cyclic structure of GRU consists of two gate structures, an update gate (purple lines) and a reset gate (red lines), and its cell structure is shown in Figure 2.

The update gate z_t can determine the memory information at the previous time and the remaining part of the information at the current time and continue to transfer the remaining information to the future time so as to obtain the long-term dependence in the entire network transmission process. The reset gate r_t is mainly used to obtain short-term time dependence, control the operation of the hidden state information h_{t-1} and the current input value x_t at the previous moment, and decide to forget the amount of information in the past.

Formulas (6)–(9) represent the calculation process of each state within each time step in GRU cell.

$$z_t = \sigma(W_z x_t + U_z h_{t-1} + b_z), \quad (6)$$

$$r_t = \sigma(W_r x_t + U_r h_{t-1} + b_r), \quad (7)$$

$$g_t = \tanh(W_g x_t + U_g (r_t \otimes h_{t-1}) + b_g), \quad (8)$$

$$h_t = z_t \otimes h_{t-1} + (1 - z_t) \otimes g_t, \quad (9)$$

where W_z , W_h , and W_g are input-related weight matrices; U_z , U_h , and U_g are cyclically connected weight matrices; and b_z , b_r , and b_g are related biases.

2.4. Attention Mechanism. Attention mechanism focuses on important information by assigning different weights to input features. The process of focusing on important information is shown as the calculation process of weight. The higher the importance of information is, the larger the weight is allocated. In the application of attention mechanism in deep learning model, the calculation process of context vector and weight involved is as follows.

The output hidden state of the deep learning model is supposed as $h_1, h_2, \dots, h_i, \dots, h_t$, and the context vector C_t can be calculated as follows:

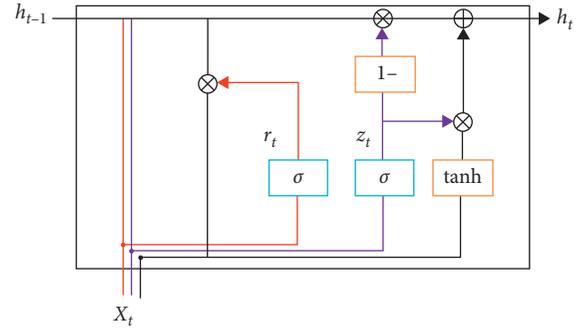


FIGURE 2: The basic unit structure of GRU.

$$C_t = \sum_{i=1}^t \alpha_{t,i} h_i. \quad (10)$$

In formula (10), $\alpha_{t,i}$ is the weight for h_i , and the sum of the weights is 1. It can be calculated as follows:

$$\alpha_{t,i} = \frac{\exp(e_{t,i})}{\sum_{i=1}^T \exp(e_{t,i})}, \quad (11)$$

where $e_{t,i}$ is an alignment model, and its calculation formula is as follows:

$$e_{t,i} = \tanh(W_a s_{t-1} + U_a h_i + b_a), \quad (12)$$

where W_a , U_a , and b_a are the network parameters of deep learning model and s_{t-1} can be calculated as follows:

$$s_{t-1} = g(s_{t-2}, y_{t-2}, c_{t-1}), \quad (13)$$

where $g(\cdot)$ denotes the deep learning network.

Based on formula (13), the output of the attention mechanism is expressed as follows:

$$y_t = \text{softmax}(s_t), \quad (14)$$

where softmax is activation function.

3. Model

3.1. Dynamic Optimal Weighted Coefficient Algorithm. Compared with a single prediction model, the combined prediction model can comprehensively utilize the advantages of multiple prediction models, improve the accuracy of prediction results, and has better robustness. In the combined prediction model, the calculation of the weighted coefficient of each single prediction model is the key. Generally, the optimal weighted coefficient algorithm (OWCA) is used, in which the weighted coefficient of each single prediction method is calculated with the goal of minimizing the sum of squared errors of the combined prediction [65–68]. The calculation principle is as follows.

Suppose there are m prediction methods; the prediction value of the i th method at time t is y_{it} , where $i = 1, 2, \dots, m$; $t = 1, 2, \dots, N$. Then, the prediction error e_{it} of the i th prediction method can be expressed by the following:

$$e_{it} = y_t - y_{it}. \quad (15)$$

Let l_1, l_2, \dots, l_m be the weighted coefficients of m prediction methods, respectively, and $l_1 + l_2 + \dots + l_m = 1$. The prediction result of the combined prediction method, labeled as \hat{y}_t , can be calculated as follows:

$$\hat{y}_t = l_1 y_{1t} + l_2 y_{2t} + \dots + l_m y_{mt}, \quad (16)$$

and the prediction error e_t for the combined prediction method at time t can be obtained:

$$e_t = \sum_{i=1}^m l_i e_{it}. \quad (17)$$

Let J represent the sum of squared errors of the combined prediction method, then the problem of solving the optimal weight at time t can be expressed as the following optimization model:

$$\text{Min } J = \sum_{t=1}^N \sum_{i=1}^m \sum_{j=1}^m l_i l_j e_{it} e_{jt}. \quad (18)$$

Formula (18) can be expressed in matrix form as follows:

$$\begin{aligned} \text{Min } J &= L^T E L \\ \text{s.t. } R^T L &= 1, \end{aligned} \quad (19)$$

where $L = (l_1, l_2, \dots, l_m)^T$ represents the weighted coefficient column vector; $R = (1, 1, \dots, 1)^T$ represents the m -dimensional column vector with all 1 elements; E is the combined prediction information error matrix, $E = (E_{ij})_{m \times m}$ and E_{ij} is expressed as follows:

$$E_{ij} = e_i^T e_j, \quad i, j = 1, 2, \dots, m, \quad (20)$$

where e_i represents the prediction error column vector of the i th single prediction method, and $e_i = (e_{i1}, e_{i2}, \dots, e_{iN})^T$.

If the prediction error vector group of m prediction methods is linearly independent, then the combined prediction information error matrix E is an invertible matrix. According to the Lagrange multiplier method [69], the optimal solution of model (18) can be obtained as follows:

$$L^* = \frac{E^{-1} R}{R^T E^{-1} R}, \quad (21)$$

where L^* is the optimal weight vector, namely, the optimal weighted coefficients of m prediction methods.

According to the OWCA and the historical prediction error of each single prediction method, the optimal weighted coefficient of each single prediction method can be obtained so as to carry out the combined prediction. In the OWCA, the weighted coefficient of each single prediction method is fixed. However, in the prediction of time data sequences, such as traffic flow, with the increase of time, the prediction results of each single prediction method also increase. More importantly, the prediction errors of each single prediction method also vary. If the weighted coefficient of each single prediction method is invariable, it cannot reflect the influence of the newly increased prediction results of each

single prediction method on the combined forecasting, which also affects the accuracy of the combined forecasting results.

Therefore, based on the optimal weighted coefficient algorithm, a dynamic optimal weighted coefficient algorithm, namely, DOWCA, is proposed. In the DOWCA, with the increase of time, the amount of historical prediction error data increases continuously, the weighted coefficient of each single prediction method, namely, the dynamic weighted coefficient, labeled as $l_{1t}, l_{2t}, \dots, l_{mt}$, is recalculated by the OWCA. The dynamic weighted coefficients are applied to each single prediction method and the combined prediction results are obtained. The whole process of the DOWCA is shown in Figure 3, and the pseudocode of DOWCA is shown in Algorithm 1.

3.2. Combined Deep Learning Prediction Model. CNN has the ability to obtain local trend features of data sequences, while LSTM and GRU have the ability to obtain long-term dependent features of data sequences. At the same time, the attention mechanism can make the deep learning model pay attention to important features. Based on this, a combined deep learning prediction model with CNN, LSTM, GRU, and DOWCA is designed for traffic flow prediction, namely, CDLP model.

In the CDLP model, CNN, LSTM, and attention are connected sequentially and become the sequential combination structure, which is named as CNN-LSTM-attention model, i.e., one single traffic flow prediction model in the CDLP model. Moreover, CNN, GRU, and attention are also designed as the sequential combination structure and named as CNN-GRU-attention model, i.e., another single traffic flow prediction model in the CDLP model. Then, the two sequential combination structures are paralleled and combined by DOWCA. From a layer standpoint, the CDLP model has three layers, input layer, hidden layer, and output layer. The hidden layer includes four layers, CNN layer, LSTM and GRU layer, attention layer, and dropout layer. The whole structure of the CDLP model is shown in Figure 4.

The input layer of the CDLP model is the processed traffic flow data sequence, including training set and test set, which is simultaneously inputted to two paralleled CNN layers in the hidden layer of the CDLP model.

The hidden layer of CDLP includes two CNN layers, LSTM and GRU layers, two attention layers, and two dropout layers in sequence. Moreover, all of them are paralleled. About the CNN layer, due to the periodicity and sequence of traffic flow data, 1DCNN is used and the output of 1DCNN is computed by the activation function ReLu. The formula of ReLu is as follows:

$$\text{ReLU}(x) = \begin{cases} x, & x > 0, \\ 0, & \text{otherwise.} \end{cases} \quad (22)$$

About LSTM and GRU layers, if too many network layers are selected, the calculation of the entire network will be large and more training time will be needed. According to [70], when both the accuracy of the prediction model and the training time are considered, the two LSTM network layers

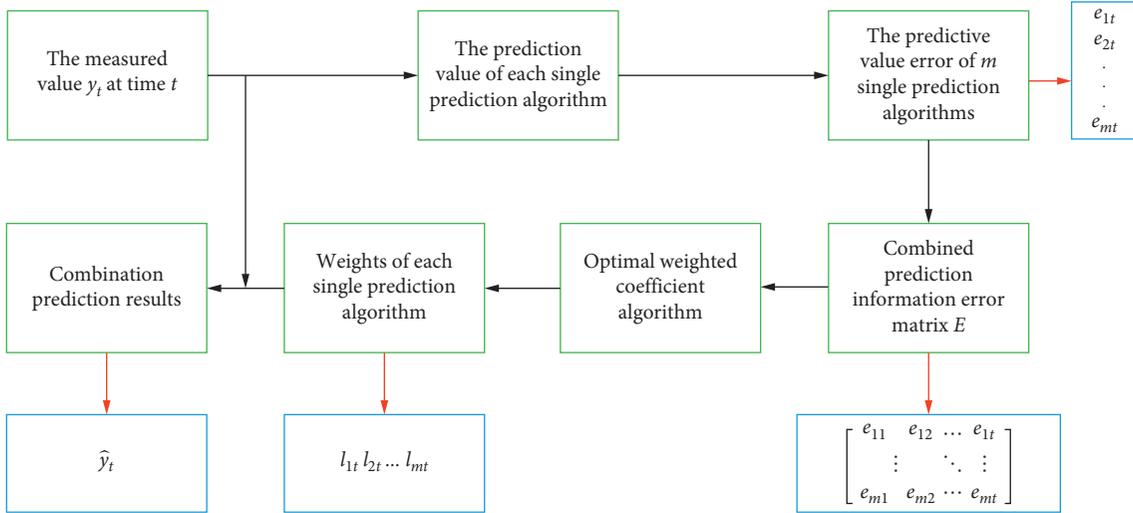
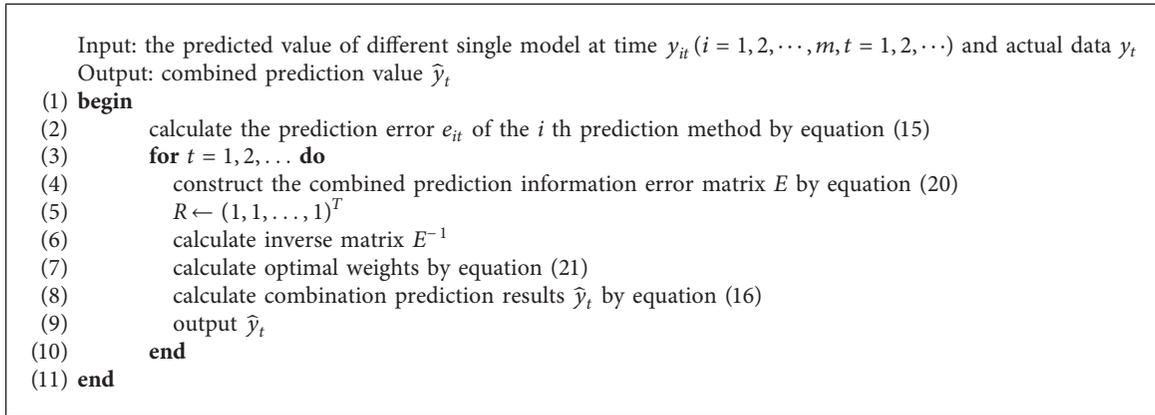


FIGURE 3: The process of DOWCA.



ALGORITHM 1: The pseudocode of DWOCA

are suitable, so two network layers in LSTM are selected. Similarly, two network layers are selected in GRU. The input of the first LSTM and GRU network layer is local trend features extracted by 1DCNN and its output is the state of the neural unit of the current LSTM and GRU layer. The second LSTM and GRU network layer mines the characteristics of the data and outputs the hidden layer state to the attention layer.

About the attention layer, the input state $h_1, h_2, \dots, h_i, \dots, h_t$ comes from LSTM and GRU. Correspondingly, $g(\cdot)$ in formula (13) denotes LSTM and GRU.

The last layer in the hidden layer, the dropout layer, is designed to prevent the occurrence of overfitting after the attention layer, which is the output from the hidden layer of CDLP to the output layer. Moreover, the input of the dropout layer is the output y_1, \dots, y_{t-1}, y_t from the attention layer.

The CDLP model is aimed to predict the traffic flow at the next moment based on the historical data. Therefore, the output layer includes two paralleled neural units, which are actually the outputs of two single models, CNN-LSTM-

attention model and CNN-GRU-attention model, respectively. The two output neural units are fully connected with the dropout layer. In addition, the output layer of the CDLP model also includes weight calculation for the outputs of the CNN-LSTM-attention model and CNN-GRU-attention model, in which the DOWCA is used. Finally, the traffic flow prediction values and the dynamic optimal weights for outputs of CNN-LSTM-attention and CNN-GRU-attention are obtained.

4. Experiment

4.1. Data Processing and Dataset. The traffic flow data at the intersection of Jiangxi Road and South Fuzhou Road in Qingdao, China, are collected through inductive detector as the dataset for the verification of the CDLP model. The original dataset contains three consecutive months of traffic flow data for each entrance road segment at the intersection from February 15, 2019, to May 15, 2019. The statistical time interval of these data is 5 minutes, and a total of 25920 pieces of data are obtained.

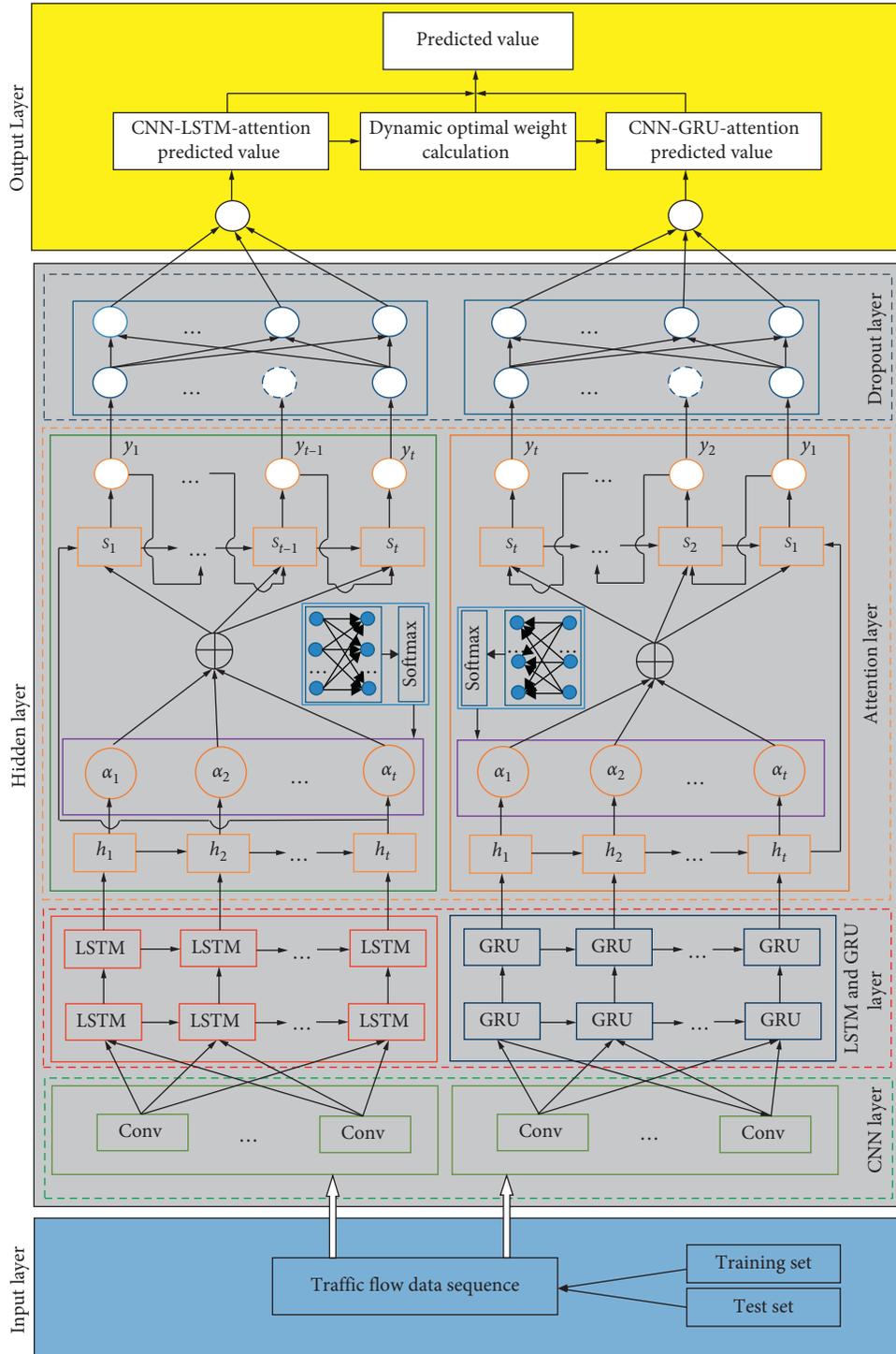


FIGURE 4: The structure of the CDLP model.

First, the abnormal and missing data in the original data are processed, in which the abnormal data are regarded as missing data. The Lagrangian interpolation method is used to process the missing data. In the process, four adjacent data before and after the missing datum are selected for interpolation to ensure the reliability of the interpolation data.

Then, the Min-Max method is used to normalize the data, and the calculation formula is as follows:

$$y' = \frac{y - y_{\min}}{y_{\max} - y_{\min}}, \quad (23)$$

where y_{\min} and y_{\max} are the minimum and maximum values of traffic flow, respectively and y and y' are the traffic flow data before and after being normalized, respectively.

The normalized data are divided into the training set and test set. The data from February 15, 2019, to May 1, 2019, are

used as the training set, and the dataset from May 2, 2019, to May 15, 2019, is the test set.

4.2. Experimental Environment and Selection of Evaluation Indicators. The hardware and software conditions in the experimental environment of this article are shown in Table 1.

In order to evaluate the traffic flow prediction performance of the CDLP model, three evaluation indicators are selected: MAPE, MAE, and RMSE. Their calculation formulas are as follows:

$$\begin{aligned} \text{MAPE} &= \frac{1}{n} \sum_{i=1}^n \left| \frac{y_i - \hat{y}_i}{y_i} \right| \times 100, \\ \text{MAE} &= \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i|, \\ \text{RMSE} &= \left(\frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{n} \right)^{1/2}, \end{aligned} \quad (24)$$

where n is the total number of samples in the test set, y_i is the i th actual value of sample, and \hat{y}_i is the predicted value of the i th sample.

4.3. CDLP Model Parameter Setting

4.3.1. Loss Function. The loss function quantifies how close a given neural network is to the ideal state it is trained on. The average absolute error function and the mean square error function are used as loss functions commonly. Because of the convenient calculation of the mean square error function, in the CDLP model, the mean square error function is selected as a loss function and the calculation formula is as follows:

$$L(y, \hat{y}) = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2, \quad (25)$$

where y_i is the actual value, n is the total number of samples, and \hat{y}_i is the prediction value.

4.3.2. The Neuron Number in the CDLP Model. The neuron numbers of the input layer and hidden layer should be set before the model is trained (the number of neurons in the output layer has been determined in Section 3.2). The following is the process of setting the number of neurons in the input layer and the hidden layer.

In order to obtain the appropriate neuron number of the input layer, 6, 12, 18, and 24 are selected, respectively, to train the model, and the optimal neuron number is obtained through error analysis of the test set. Similarly, for the setting of the neuron number of LSTM layers and GRU layers, four numbers of 16, 32, 64, and 128 are selected, respectively, to train the model. Moreover, the optimal neuron number is determined through the error analysis of the test set.

Regarding the error analysis of the test set, MAPE is selected as the main evaluation indicator, while MAE and

TABLE 1: Experimental environment.

Software and hardware configuration	Configuration parameter
CPU	Intel i5-8250U @1.60 GHz
RAM	8G
Programming language	Python 3.7.0
Deep learning framework	Tensorflow 1.14
Deep learning library	Keras 2.3.1

RMSE are used as auxiliary evaluation indicators. The evaluation indicator results of the test set under different neuron numbers in input and LSTM layers are obtained, which include the MAPE, MAE, and RMSE, as shown in Table 2.

From Table 2, it can be seen that when the neuron number of the input layer is set to 12 and the neuron numbers of the two LSTM layers are set to 128 and 128, respectively; the MAPE, MAE, and RMSE of the model test set are all the smallest. It indicates that the neuron numbers of the input layer and hidden layer are the best for the model training effect under this setting. Moreover, the neuron numbers of the two GRU layers are the same as those of LSTM, i.e., 128 and 128, respectively.

4.3.3. Optimization Algorithm. In the training process of the deep learning model, an optimization algorithm is used to iterate the model parameters to reduce the loss function value so that the training process of the model tends to be stable as the number of iterations increases. The optimization algorithms mainly include RMSprop and Adam. The two algorithms are applied to train the CDLP model and the better one is selected as an optimization algorithm according to the prediction results. After training of CDLP model under RMSProp algorithm and Adam algorithm, respectively, the results of three evaluation indicators are obtained and shown in Table 3.

It can be found from Table 3 that when the Adam algorithm is used to train the CDLP model, the MAPE, MAE, and RMSE are less than those of the RMSProp algorithm. It indicates that the Adam algorithm is more effective than the RMSProp algorithm and is selected as the optimization algorithm of the CDLP model.

4.3.4. Other Parameters. In the 1DCNN layer, the convolution operation is implemented by convolution kernels, and 64 convolution kernels with a size of 2×1 are used, i.e., filters = 64, size = 2. In the dropout layer, the loss rate of the dropout function is set as 20%. In addition, the epoch is set as 500 iterations, and the batch size is set as 128.

4.4. Results and Analysis. The CDLP model is trained and tested with a designed training set and test set after the above model parameters are determined. At the same time, in order to verify the advantages of the CDLP model, the prediction results from the single CNN-LSTM-attention model and single CNN-GRU-attention model are extracted during the process of training and testing for the CDLP

TABLE 2: Evaluation indicator results under different neuron numbers of input and LSTM layer.

Neuron number in the network	MAPE (%)	MAE	RMSE
(6,16,16,1)	25.52	24.92	35.77
(6,32,32,1)	12.99	19.85	29.54
(6,64,64,1)	10.60	12.52	19.40
(6,128,128,1)	12.07	9.68	14.03
(12,16,16,1)	12.86	20.68	31.48
(12,32,32,1)	24.06	19.93	28.90
(12,64,64,1)	9.24	8.65	13.00
(12,128,128,1)	7.33	5.12	8.25
(18,16,16,1)	15.59	20.84	31.93
(18,32,32,1)	10.78	13.60	20.48
(18,64,64,1)	14.47	9.29	13.53
(18,128,128,1)	12.86	7.10	9.92
(24,16,16,1)	13.35	20.88	32.02
(24,32,32,1)	12.59	12.98	19.65
(24,64,64,1)	22.54	11.37	15.66
(24,128,128,1)	21.84	7.81	11.42

TABLE 3: Comparison of three evaluation indicators between RMSProp and Adam.

Optimization algorithm	MAPE (%)	MAE	RMSE
Adam	7.31	5.21	9.06
RMSprop	8.21	5.60	9.73

model. Moreover, the corresponding results are obtained. Figure 5 shows the loss function curve of the training set and test set of CNN-LSTM-attention and CNN-GRU-attention. Figure 6 shows the prediction results of the CDLP model for the test set.

From Figure 5(a), it can be seen that the loss function of the training set of the CNN-LSTM-attention decreases rapidly and steadily as the number of iterations increases and finally tends to a stable state. Then, the loss function of the test set goes through initial fluctuations as the iteration progresses, quickly tends to the loss function of the training set, and is in a stable state. It can be seen from Figure 5(b) that similar to the CNN-LSTM-attention, the loss function of the training set of the CNN-GRU-attention network decreases rapidly and steadily and finally tends to a stable state and the loss function of the test set also gradually tends to the training set after initial fluctuations. Finally, the loss function is in a stable state. The loss function curves of the training set and test set of CNN-LSTM-attention and CNN-GRU-attention show that the design of CNN-LSTM-attention and CNN-GRU-attention network in the CDLP model is reasonable.

Figure 6 contains a comparison of the traffic flow predicted results of the CDLP model with the actual value (top figure) and the error of predicted traffic flow (below figure). From the top figure, it can be found that the CDLP model fits the change trend of traffic flow very well, indicating that the model learns the time change characteristics of the traffic flow series and realizes the better prediction. Moreover, from Figure 6, it can be seen that the overall errors remain stable and most of them change in a certain range of -20 and 20. Moreover, based on the error of predicted traffic flow, the MAPE curve is obtained

and shown in Figure 7. From the figure, it can be found that the trend of the MAPE curve first quickly rises to the maximum value, then quickly decreases, and gradually becomes stable. Finally, the MAPE curve tends to be 5.12%. This shows that the CDLP model has excellent robustness and obtains small error, further showing that the CDLP model can better realize the prediction of traffic flow.

Furthermore, in order to further verify the prediction effect of the CDLP model, Figure 8 shows the absolute error comparison of the traffic flow predicted values of the CDLP model, CNN-LSTM-attention model, and CNN-GRU-attention models. Figures 8(a) and 8(b) show the traffic flow prediction errors of the first week and the second week in the test set under three models. As can be seen from the figure, the fluctuation range of the prediction error curve of the CDLP model is the smallest, followed by CNN-LSTM-attention and CNN-GRU-attention. This indicates that the prediction accuracy of the CDLP model is better than that of CNN-LSTM-attention or CNN-GRU-attention and also shows the advantages of the combination model compared to a single model.

Meanwhile, some baseline models published in recent years, which are LSTM, GRU, CNN, CNN-LSTM, CNN-GRU, CNN-LSTM-attention, and CNN-GRU-attention, are used to verify the accuracy of the CDLP model. The evaluation indicators of the CDLP and baseline models, which are MAPE, MAE, and RMSE, are obtained, as shown in Table 4. Moreover, the training times of CDLP and baseline models are shown in Table 5. It can be seen from Table 4 that the evaluation indicators of the CDLP model are the smallest, followed by baseline models. This shows that the prediction accuracy of the CDLP model is the best.

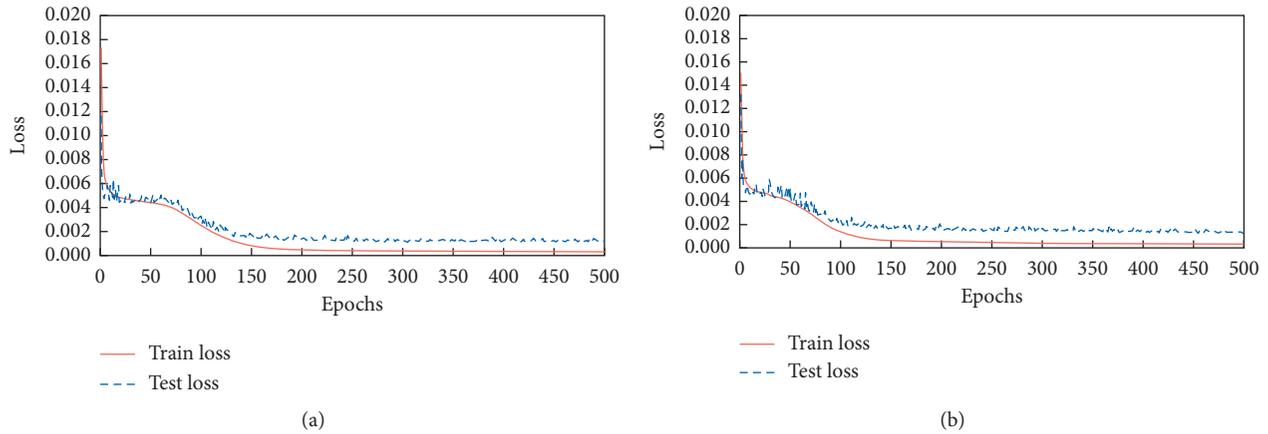


FIGURE 5: Loss function curves of the training set and test set: (a) CNN-LSTM-attention; (b) CNN-GRU-attention.

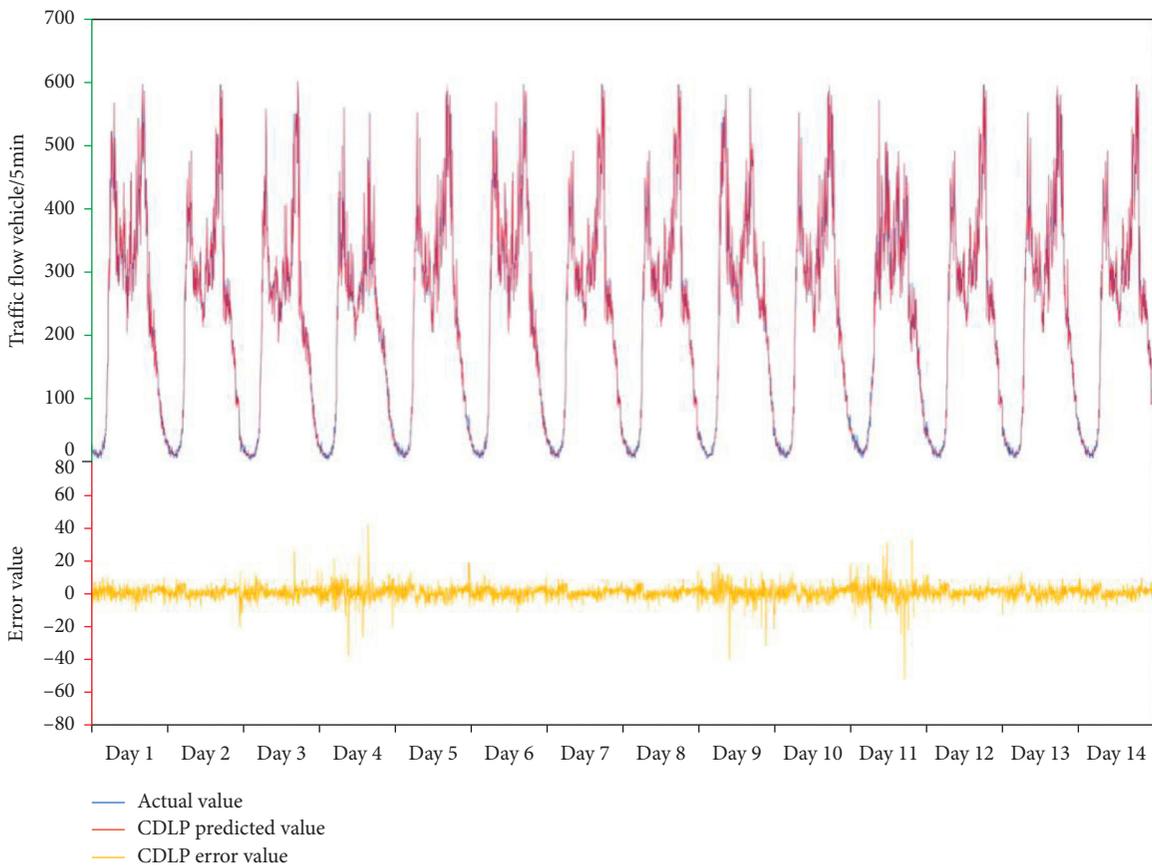


FIGURE 6: The prediction results of the CDLP model for the test set.

Moreover, it can be found from Table 5 that the training time of the CDLP model is as long as the time of the CNN-LSTM-attention model, but its prediction accuracy is higher than that of CNN-LSTM-attention model. The training time of the CNN model is the shortest, but the prediction accuracy is the lowest, so the robustness of the CDLP model is relatively high.

In addition, according to the DOWCA, the weights of CNN-LSTM-attention model and CNN-GRU-attention model in the CDLP model are calculated, as shown in Figure 9.

Figure 9 shows that the weights of CNN-LSTM-attention and CNN-GRU-attention are dynamic and constantly changing, which indicates the two methods have different

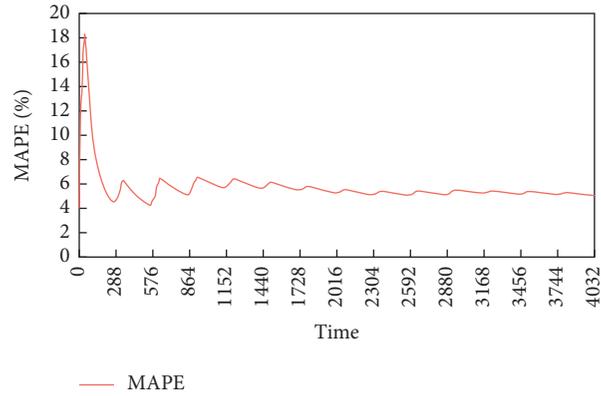


FIGURE 7: MAPE curve of CDLP model for the test set.

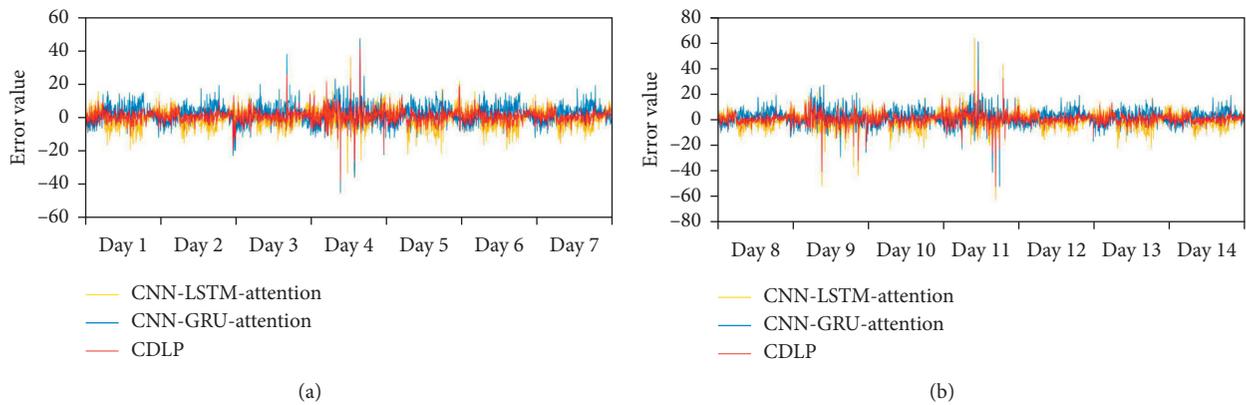


FIGURE 8: Error comparison of CDLP, CNN-LSTM-attention, and CNN-GRU-attention: (a) error of the first week in test set; (b) error of the second week in the test set.

TABLE 4: Comparison of three evaluation indexes of CDLP and baseline models.

Model	MAPE (%)	MAE	RMSE
LSTM	8.40	5.34	8.29
GRU	10.79	8.61	12.03
CNN	15.80	20.91	32.41
CNN-LSTM	7.67	4.64	6.66
CNN-GRU	7.48	4.21	6.43
CNN-LSTM-attention	6.64	4.81	7.69
CNN-GRU-attention	7.10	4.97	7.98
CDLP	5.12	3.26	6.52

TABLE 5: Comparison of model training time of CDLP and baseline models.

Model	Training time (min)
LSTM	48
GRU	42
CNN	35
CNN-LSTM	57
CNN-GRU	51
CNN-LSTM-attention	64
CNN-GRU-attention	60
CDLP	64

prediction results for the same traffic flow data. Moreover, it can be seen from Figure 8 that the weights of the two models gradually decrease from a large change at the beginning and eventually become stable, which reflects the systematic feasibility of the dynamic weighted coefficient algorithm, namely, the convergence. Furthermore, it shows that the weights of the CNN-LSTM-attention model are greater than those of the CNN-GRU-attention model, indicating that the prediction accuracy of the CNN-LSTM-attention model is higher than the CNN-GRU-attention model, which is consistent with the results in Table 4.

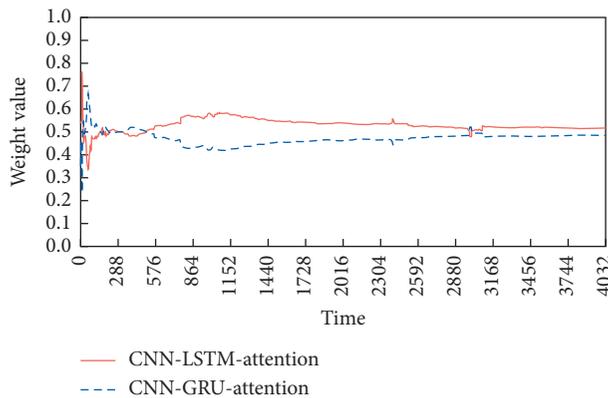


FIGURE 9: Dynamic weight value of CNN-LSTM-attention and CNN-GRU-attention model.

5. Conclusion and Future Work

Traffic flow prediction is an important part of the intelligent transportation system. In this article, a dynamic weighted coefficient algorithm for combinational prediction model is presented, namely, DOWCA. Furthermore, based on CNN, LSTM, GRU, and DOWCA, a combined deep learning model for short-term traffic flow prediction is proposed, namely, CDLP model. The structure of the CDLP model with an input layer, a hidden layer, and an output layer is designed. From the point of the combined model, the CDLP model includes two paralleled single models, i.e., CNN-LSTM-attention model and CNN-GRU-attention model. The parameters of CDLP model are determined by experiment, which includes loss function, the neuron number, and optimization algorithm.

The data from a field intersection are collected, and the dataset for the CDLP model is obtained through abnormal and missing data processing and normalization processing, which is divided into the training set and test set. The CDLP model is trained and tested. The results obtained show that the feasibility of the CDLP model can predict traffic flow with high accuracy. Moreover, in order to further verify the performance of the established model, based on the same dataset and the same parameter settings as the CDLP model, the baseline models are, respectively, used to predict the traffic flow. After analyzing the prediction results of these models, the results show that the accuracy of the CDLP model is higher than the baseline models. And DOWCA is validated to obtain the optimal weighted coefficients for CNN-LSTM-attention and CNN-GRU-attention in the CDLP model dynamically.

The structure of a CDLP model is designed and its parameters are set in this article. However, some parameters, for example, the number of nodes in the input layer and hidden layer in the model is obtained through experiments based on the selection of short-term traffic flow parameters in the past. How to optimize the parameters in a combined deep learning model needs to be further studied. Furthermore, traffic flow prediction involves several parameters; the deep learning structures based on the combinatorial algorithm can be expanded to multidimensional input variables, such as traffic speed and occupancy.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This study was supported by China Postdoctoral Science Foundation Funded Project (2019M652437), the Scientific Research Foundation of Shandong University of Science and Technology for Recruited Talents (2019RCJJ014), Shandong Postdoctoral Innovation Project (201903030), and Key Research and Development Project of Shandong Province (2019GGX101008).

References

- [1] P. Ersoy and G. Brühn, "Intelligent transportation systems and their applications in road transportation industry in Turkey," in *Proceedings of the 12th International Conference on Logistics & Sustainable Transport*, pp. 11–13, Celje, Slovenia, June 2015.
- [2] N. Cui, B. Chen, K. Zhang, Y. Zhang, X. Liu, and J. Zhou, "Effects of route guidance strategies on traffic emissions in intelligent transportation systems," *Physica A: Statistical Mechanics and its Applications*, vol. 513, pp. 32–44, 2019.
- [3] I. O. Olayode, L. k. Tartibu, M. O. Okwu, and U. F. Uchechi, "Intelligent transportation systems, un-signalized road intersections and traffic congestion in Johannesburg: a systematic review," *Procedia CIRP*, vol. 91, pp. 844–850, 2020.
- [4] A. Richter, M.-O. Löwner, R. Ebendt, and M. Scholz, "Towards an integrated urban development considering novel intelligent transportation systems: urban development considering novel transport," *Technological Forecasting and Social Change*, vol. 155, Article ID 119970, 2020.
- [5] I. Okutani and Y. J. Stephanedes, "Dynamic prediction of traffic volume through Kalman filtering theory," *Transportation Research Part B: Methodological*, vol. 18, no. 1, pp. 1–11, 1984.
- [6] Y. Xie, Y. Zhang, and Z. Ye, "Short-term traffic volume forecasting using Kalman filter with discrete wavelet decomposition," *Computer-Aided Civil and Infrastructure Engineering*, vol. 22, no. 5, pp. 326–334, 2007.
- [7] J. Guo, W. Huang, and B. M. Williams, "Adaptive Kalman filter approach for stochastic short-term traffic flow rate prediction and uncertainty quantification," *Transportation Research Part C: Emerging Technologies*, vol. 43, pp. 50–64, 2014.
- [8] A. Emami, M. Sarvi, and S. A. Bagloee, "Short-term traffic flow prediction based on faded memory Kalman Filter fusing data from connected vehicles and bluetooth sensors," *Simulation Modelling Practice and Theory*, vol. 102, Article ID 102025, 2019.
- [9] M. S. Ahmed and A. R. Cook, "Analysis of freeway traffic time-series data by using Box-Jenkins techniques," *Transportation Research Board*, vol. 773, no. 722, pp. 1–9, 1979.
- [10] M. M. Hamed, H. R. Al-Masaeid, and Z. M. B. Said, "Short-term prediction of traffic volume in urban arterials," *Journal of Transportation Engineering*, vol. 121, no. 3, pp. 249–254, 1995.

- [11] B. M. Williams and L. A. Hoel, "Modeling and forecasting vehicular traffic flow as a seasonal ARIMA process: theoretical basis and empirical results," *Journal of Transportation Engineering*, vol. 129, no. 6, pp. 664–672, 2003.
- [12] B. M. Williams, "Multivariate vehicular traffic flow prediction: evaluation of ARIMAX modeling," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 1776, no. 1, pp. 194–200, 2001.
- [13] L. Zhang, Q. Liu, W. Yang, N. Wei, and D. Dong, "An improved k-nearest neighbor model for short-term traffic flow prediction," *Procedia - Social and Behavioral Sciences*, vol. 96, pp. 653–662, 2013.
- [14] S. Cheng, F. Lu, P. Peng, and S. Wu, "Short-term traffic forecasting: an adaptive ST-KNN model that considers spatial heterogeneity," *Computers, Environment and Urban Systems*, vol. 71, pp. 186–198, 2018.
- [15] Z. Yang, Y. Wang, and Q. Guan, "Short-term traffic flow prediction method based on SVM," *Journal of Jilin University (Engineering and Technology Edition)*, vol. 36, pp. 881–884, 2006.
- [16] C. Luo, C. Huang, J. Cao et al., "Short-term traffic flow prediction based on least square support vector machine with hybrid optimization algorithm," *Neural Processing Letters*, vol. 50, no. 3, pp. 2305–2322, 2019.
- [17] J. Tang, X. Chen, Z. Hu, F. Zong, C. Han, and L. Li, "Traffic flow prediction based on combination of support vector machine and data denoising schemes," *Physica A: Statistical Mechanics and its Applications*, vol. 534, Article ID 120642, 2019.
- [18] W. Hong, "Traffic flow forecasting by seasonal SVR with chaotic simulated annealing algorithm," *Neurocomputing*, vol. 74, no. 12–13, pp. 2096–2107, 2011.
- [19] M. Castro-Neto, Y.-S. Jeong, M.-K. Jeong, and L. D. Han, "Online-SVR for short-term traffic flow prediction under typical and atypical traffic conditions," *Expert Systems with Applications*, vol. 36, no. 3, pp. 6164–6173, 2009.
- [20] H. Wang, W. Quan, and W. Y. Ochieng, "Smart road stud based two-lane traffic surveillance," *Journal of Intelligent Transportation Systems*, vol. 24, no. 5, pp. 480–493, 2020.
- [21] S. F. Wong, H. C. Mak, C. H. Ku, and W. I. Ho, "Developing advanced traffic violation detection system with RFID technology for smart city," in *Proceedings of the IEEE International Conference on Industrial Engineering and Engineering Management*, Singapore, February 2018.
- [22] A. Klein, M. Kelley, and M. Mills, "Evaluation of traffic detection technologies for IVHS," *Proceedings of SPIE*, vol. 2344, pp. 42–53, 1995.
- [23] J. Versavel and B. Boucke, "Video for traffic data and incident detection by traficon," in *Proceedings of the Fifth International Conference on Applications of Advanced Technologies in Transportation Engineering*, Newport Beach, CA, USA, April 1998.
- [24] D. Llorca, M. Sotelo, S. Sanchez, M. Ocaña, J. Rodriguez-Ascariz, and M. Garrido, "Traffic data collection for floating car data enhancement in V2I networks," *EURASIP Journal on Advances in Signal Processing*, vol. 2010, Article ID 719294, 13 pages, 2010.
- [25] L. Theis, W. Shi, A. Cunningham, and F. Huszar, "Lossy image compression with compressive autoencoders," in *Proceedings of the 5th International Conference on Learning Representations*, Toulon, France, March 2017.
- [26] L. Gao, Z. Guo, H. Zhang, X. Xu, and H. T. Shen, "Video captioning with attention-based LSTM and semantic consistency," *IEEE Transactions on Multimedia*, vol. 19, no. 9, pp. 2045–2055, 2017.
- [27] A. Graves and N. Jaitly, "Towards end-to-end speech recognition with recurrent neural networks," in *Proceedings of the 31st International Conference on Machine Learning*, vol. 5, pp. 3771–3779, Beijing, China, June 2014.
- [28] H.-F. Yang, T. S. Dillon, and Y.-P. P. Chen, "Optimized structure of the traffic flow forecasting model with a deep learning approach," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 10, pp. 2371–2381, 2017.
- [29] Y. Wu, H. Tan, L. Qin, B. Ran, and Z. Jiang, "A hybrid deep learning based traffic flow prediction method and its understanding," *Transportation Research Part C: Emerging Technologies*, vol. 90, pp. 166–180, 2018.
- [30] J. Wang, R. Chen, and Z. He, "Traffic speed prediction for urban transportation network: a path based deep learning approach," *Transportation Research Part C: Emerging Technologies*, vol. 100, pp. 372–385, 2019.
- [31] W. Wang, H. Zhang, T. Li et al., "An interpretable model for short term traffic flow prediction," *Mathematics and Computers in Simulation*, vol. 171, pp. 264–278, 2020.
- [32] X. Guo and Q. Zhu, "A traffic flow forecasting model based on BP neural network," in *Proceedings of the International Conference on Power Electronics & Intelligent Transportation System*, pp. 311–314, IEEE, Shenzhen, China, January 2010.
- [33] H. Huang, Q. Tang, and Z. Liu, "Adaptive correction forecasting approach for urban traffic flow based on Fuzzyc-mean clustering and advanced neural network," *Journal of Applied Mathematics*, vol. 2013, Article ID 195824, 7 pages, 2013.
- [34] T. Li and L. Sheng, "Prediction for short-term traffic flow based on optimized wavelet neural network model," *International Journal of Computer Science & Information Technology*, vol. 7, no. 2, 2015.
- [35] W. Huang, G. Song, H. Hong, and K. Xie, "Deep architecture for traffic flow prediction: deep belief networks with multitask learning," *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 5, pp. 2191–2201, 2014.
- [36] A. Koesdwiady, R. Soua, and F. Karray, "Improving traffic flow prediction with weather information in connected cars: a deep learning approach," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 12, pp. 9508–9517, 2016.
- [37] H. Xu and C. Jiang, "Deep belief network-based support vector regression method for traffic flow forecasting," *Neural Computing and Applications*, vol. 32, no. 7, pp. 2027–2036, 2020.
- [38] L. Han and Y. S. Huang, "Short-term traffic flow prediction of road network based on deep learning," *IET Intelligent Transport Systems*, vol. 14, no. 6, pp. 495–503, 2020.
- [39] X. Ma, Z. Dai, Z. He, J. Ma, Y. Wang, and Y. Wang, "Learning traffic as images: a deep convolutional neural network for large-scale transportation network speed prediction," *Sensors*, vol. 17, no. 4, 2017.
- [40] W. Zhang, Y. Yu, Y. Qi, F. Shu, and Y. Wang, "Short-term traffic flow prediction based on spatio-temporal analysis and CNN deep learning," *Transportmetrica A: Transport Science*, vol. 15, no. 2, pp. 1688–1711, 2019.
- [41] J. An, L. Fu, M. Hu, W. Chen, and J. Zhan, "A novel fuzzy-based convolutional neural network method to traffic flow prediction with uncertain traffic accident information," *IEEE Access*, vol. 7, pp. 20708–20722, 2019.
- [42] Q. Liu, B. Wang, and Y. Zhu, "Short-term traffic speed forecasting based on attention convolutional neural network for arterials," *Computer-aided Civil and Infrastructure Engineering*, vol. 33, no. 11, pp. 999–1016, 2018.

- [43] H. Peng, H. Wang, B. Du et al., "Spatial temporal incidence dynamic graph neural networks for traffic flow forecasting," *Information Sciences*, vol. 521, pp. 277–290, 2020.
- [44] S. Maher and P. Biswajee, "Severity prediction of traffic accidents with recurrent neural networks," *Applied Sciences*, vol. 7, no. 6, pp. 476–486, 2017.
- [45] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [46] X. Ma, Z. Tao, Y. Wang, H. Yu, and Y. Wang, "Long short-term memory neural network for traffic speed prediction using remote microwave sensor data," *Transportation Research Part C: Emerging Technologies*, vol. 54, pp. 187–197, 2015.
- [47] 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 Intelligent Transport Systems*, vol. 11, no. 2, pp. 68–75, 2017.
- [48] Y. Tian, K. Zhang, J. Li, X. Lin, and B. Yang, "LSTM-based traffic flow prediction with missing data," *Neurocomputing*, vol. 318, pp. 297–305, 2018.
- [49] J. Zhao, Y. Gao, Z. Bai, H. Wang, and S. Lu, "Traffic speed prediction under non-recurrent congestion: based on LSTM method and BeiDou navigation satellite system data," *IEEE Intelligent Transportation Systems Magazine*, vol. 11, no. 2, pp. 70–81, 2019.
- [50] Z. Wang, L. Zhang, and Z. Ding, "Hybrid time-aligned and context attention for time series prediction," *Knowledge-Based Systems*, vol. 198, Article ID 105937, 2020.
- [51] J. Zhou, H. Chang, X. Cheng, and X. Zhao, "A multiscale and high-precision LSTM-GASVR short-term traffic flow prediction model," *Complexity*, vol. 2020, Article ID 1434080, 17 pages, 2020.
- [52] Y. Zhang, T. Cheng, Y. Ren, and K. Xie, "A novel residual graph convolution deep learning model for short-term network-based traffic forecasting," *International Journal of Geographical Information Science*, vol. 34, no. 5, pp. 969–995, 2020.
- [53] W. Li, X. J. Ban, J. Zheng, H. X. Liu, and Y. Li, "Real-time movement-based traffic volume prediction at signalized intersections," *Journal of Transportation Engineering Part A: Systems*, vol. 146, no. 8, Article ID 04020081, 2020.
- [54] D. Xia, M. Zhang, X. Yan, Y. Bai, and H. Li, "A distributed WND-LSTM model on map reduce for short-term traffic flow prediction," *Neural Computing and Applications*, vol. 33, no. 7, pp. 2393–2410, 2020.
- [55] S. Du, T. Li, X. Gong, and S. J. Horng, "A hybrid method for traffic flow forecasting using multimodal deep learning," *International Journal of Computational Intelligence Systems*, vol. 13, no. 1, p. 85, 2018.
- [56] G. Dai, C. Ma, and X. Xu, "Short-term traffic flow prediction method for urban road sections based on space-time analysis and GRU," *IEEE Access*, vol. 7, pp. 143025–143035, 2019.
- [57] Y. Gu, W. Lu, L. Qin, M. Li, and Z. Shao, "Short-term prediction of lane-level traffic speeds: a fusion deep learning model," *Transportation Research Part C: Emerging Technologies*, vol. 106, pp. 1–16, 2019.
- [58] C.-M. Own, F. Sha, and W. Tao, "Triplet decoders neural network ensemble system and t-conversion for traffic speed sequence prediction," *IEEE Access*, vol. 7, pp. 162070–162082, 2019.
- [59] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate, computer science," 2021, <http://arxiv.org/abs/1409.0473>.
- [60] C. Yin, X. Cheng, X. Liu, and M. Zhao, "Identification and classification of atmospheric particles based on SEM images using convolutional neural network with attention mechanism," *Complexity*, vol. 2020, Article ID 9673724, 13 pages, 2020.
- [61] L. Ye, Z. Liu, and Y. Wang, "Dual convolutional LSTM network for referring image segmentation," *IEEE Transactions on Multimedia*, vol. 22, no. 12, pp. 3224–3235, 2020.
- [62] Z. Zhao and W.-Q. Zhang, "End-to-end keyword search system based on attention mechanism and energy scorer for low resource languages," *Neural Networks*, vol. 139, pp. 326–334, 2021.
- [63] D. Li, J. Liu, Z. Yang, L. Sun, and Z. Wang, "Speech emotion recognition using recurrent neural networks with directional self-attention," *Expert Systems with Applications*, vol. 173, Article ID 114683, 2021.
- [64] J. Gu, Z. Wang, J. Kuen et al., "Recent advances in convolutional neural networks," *Pattern Recognition*, vol. 77, pp. 354–377, 2018.
- [65] G. Yang, S. Jiao, J.-P. Liu, T. Lei, and X. Yuan, "Error diffusion method with optimized weighting coefficients for binary hologram generation," *Applied Optics*, vol. 58, no. 20, pp. 5547–5555, 2019.
- [66] D. L. Guen, S. Pateux, C. Labit, G. Moury, and D. Lebedeff, "Optimal adapted weighting of wavelet coefficients for on-board coding of satellite images," in *Proceedings of the IEEE International Geoscience & Remote Sensing Symposium*, vol. 2, Honolulu, HI, USA, February 2000.
- [67] H. Wang and J. Shen, "Research on the voltage prediction of unmanned aerial vehicle photovoltaic modules based on new combination optimization algorithm," *International Transactions on Electrical Energy Systems*, vol. 29, no. 7, 2019.
- [68] Z. Qu, S. Xie, R. Feng, N. Qu, H. Lv, and Z. Chi, "Electricity sales prediction model of electricity market of the non-linear regression combination with the optimal weight," *Journal of Computers*, vol. 30, no. 5, pp. 75–87, 2019.
- [69] L. M. Castro, N. González-Cabrera, D. Guillen, J. Tovar-Hernández, and G. Gutiérrez-Alcaraz, "Efficient method for the optimal economic operation problem in point-to-point VSC-HVDC connected AC grids based on Lagrange multipliers," *Electric Power Systems Research*, vol. 187, Article ID 106493, 2020.
- [70] Q. Wang, S. Bu, and Z. He, "Achieving predictive and proactive maintenance for high-speed railway power equipment with LSTM-RNN," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 10, pp. 6509–6517, 2020.