HIGH TECHNOLOGY LETTERS | Vol. 29 No. 1 | Mar. 2023 | pp. 41-49

doi:10.3772/j.issn.1006-6748.2023.01.005

MEEMD-DBA-based short term traffic flow prediction①

ZHANG Xijun(张玺君)②, HAO Jun, NIE Shengyuan, CUI Yong
(College of Computer and Communication, Lanzhou University of Technology, Lanzhou 730050, P. R. China)

Abstract

Aiming at the problem that ensemble empirical mode decomposition (EEMD) method can not completely neutralize the added noise in the decomposition process, which leads to poor reconstruction of decomposition results and low accuracy of traffic flow prediction, a traffic flow prediction model based on modified ensemble empirical mode decomposition (MEEMD), double-layer bidirectional long-short term memory (DBiLSTM) and attention mechanism is proposed. Firstly, the intrinsic mode functions (IMFs) and residual components (Res) are obtained by using MEEMD algorithm to decompose the original traffic data and separate the noise in the data. Secondly, the IMFs and Res are put into the DBiLSTM network for training. Finally, the attention mechanism is used to enhance the extraction of data features, then the obtained results are reconstructed and added. The experimental results show that in different scenarios, the MEEMD-DBiLSTM-attention (MEEMD-DBA) model can reduce the data reconstruction error effectively and improve the accuracy of the short-term traffic flow prediction.

Key words: modified ensemble empirical mode decomposition (MEEMD), double bidirectional-directional gated recurrent unit (DBiGRU), attention mechanism, traffic flow prediction

0 Introduction

Short-term traffic flow prediction technology is an important part of intelligent transportation system[1-2]. Accurate traffic flow prediction results can not only help traffic management departments to guide traffic better, but also help citizens to choose appropriate travel routes, and finally alleviate traffic pressure and facilitate citizens' travel[3-4].

Traffic data is affected by external factors, which will cause data loss and contain a lot of noise. In order to get a better traffic flow prediction result, a lot of researches about traffic data preprocessing technology have been done by researchers.

Ref. [5] used ensemble empirical mode decomposition (EEMD) to decompose the original sequence, then used long short-term memory (LSTM) to train the model. Ref. [6] used EEMD to decompose the original velocity sequence, and combined the bidirectional-LSTM (BiLSTM) and the attention mechanism to train the model.

In order to extract the spatio-temporal features of traffic data more fully, more and more researches have introduced machine learning and deep learning frameworks into prediction[7-9]. Ref. [10] established a model based on spatio-temporal graph convolutional network to predict traffic flow, which can better express the spatio-temporal correlation of traffic flow. In Ref. [11], an adaptive graph spatio-temporal graph neural network was proposed to capture the spatiotemporal correlations of traffic flow. Ref. [12] proposed a new data-driven method for traffic prediction. The model was based on geometric deep learning techniques, which can make full use of the topology information of the road network during the learning process. Ref. [13] proposed a parallel computing learning method to predict traffic flow, which improved the performance of the model.

EEMD used in Refs [5,6] can suppress the problem of mode mixing, but it can not completely neutralize the added white noise, and this algorithm requires a large amount of calculation. If the parameter selection is unreasonable, there will be more redundant components. The modified ensemble empirical mode decomposition (MEEMD) algorithm is used to decompose the data. In order to consider the influence of the data information before and after on traffic flow at the current

① Supported by the National Natural Science Foundation of China (No. 62162040, 61966023), the Higher Educational Innovation Foundation Project of Gansu Province of China (No. 2021A-028) and the Science and Technology Plan of Gansu Province (No. 21ZD4GA028).

② To whom correspondence should be addressed. E-mail: zhangxijun198079@sina.com.

Received on May 10, 2022

time, and extract the time features more fully, the double-layer BiLSTM (DBiLSTM) and attention mechanism are used to train the model.

The main innovations of this paper are as follows.

(1) The MEEMD algorithm is used for traffic flow data preprocessing, which reduces the error of data reconstruction and improves the data quality by eliminating data noise through noise energy analysis.

(2) DBiLSTM structure and attention mechanism are proposed to enhance the feature extraction of time series and improve the prediction accuracy of the model.

(3) The prediction results of the model during weekdays, weekends and multi-step prediction are analyzed to verify the robustness of the model, and the effectiveness of the model is verified by ablation experiments.

1 MEEMD

In MEEMD method, which is based on permutation entropy, complementary ensemble empirical mode decomposition (CEEMD) is used to decompose the time series containing noise, calculate the permutation entropy value of each component, determine the abnormal data according to the permutation entropy value, and eliminate them. Then, the empirical mode decomposition (EMD) method is used to decompose the remaining data[14]. For the time series $T(t)$ contains noise, the decomposition steps are as follows[15].

Step 1 Add positive and negative white Gaussian noise with the same amplitude to the original time series $T(t)$ in pairs to get two signals $T_i^+(t)$ and $T_i^-(t)$.

Step 2 Use EMD algorithm to decompose $T_i^+(t)$ and $T_i^-(t)$, and get the first intrinsic mode function (IMF) component sequence $\{I_{i1}^+(t)\}$ and $\{I_{i1}^-(t)\}$. Then use the ensemble average method to deal with the two-component sequence to get the first IMF component $I_1(t)$ of the sequence $T(t)$.

$$I_1(t) = \frac{1}{2L}\sum_{i=1}^{Ne}[I_{i1}^+(t) + I_{i1}^-(t)] \tag{1}$$

where, L represents the length of the time series $T(t)$, and Ne is the logarithm of the added white noise.

Step 3 Calculate the permutation entropy of $I_1(t)$, H_p, and judge whether this component is an abnormal component. If the entropy of $I_1(t)$ is greater than the threshold θ_0, the component is an abnormal component.

Step 4 If $I_1(t)$ is an abnormal component, return to Step 1 until the entropy value H_p of the qth IMF component $I_q(t)$ is less than θ_0, and then perform Step 5.

Step 5 Remove the decomposed former $q-1$ components from the original time series to obtain the time series $N(t)$ without noise.

$$N(t) = T(t) - \sum_{j=1}^{q-1} I_j(t) \tag{2}$$

Step 6 Use EMD method to decompose $N(t)$ to get the IMF component, MEEMD decomposition is completed.

2 Standard BiLSTM structure

BiLSTM is composed of a forward LSTM and a backward LSTM which are connected to the output layer. The data of the former moment and the next moment is simultaneously transmitted to the output layer, which can extract time series features more fully. The time features of the experimental data in this paper are extracted by BiLSTM, which is shown in Fig. 1.

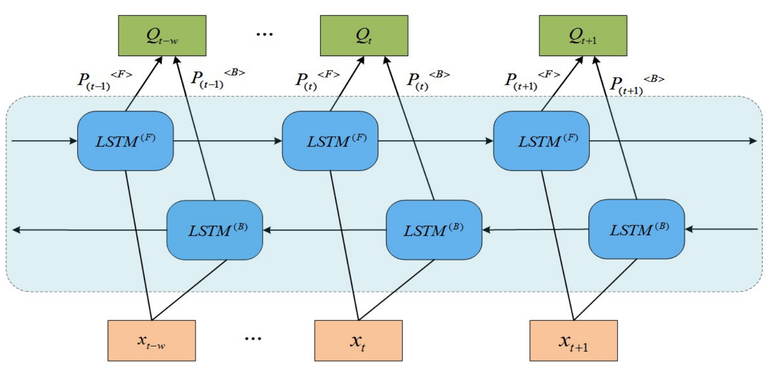


Fig. 1 BiGRU network structure

The calculation process of the BiLSTM network is described as follows.

$$P_{(t)}^{<F>} = LSTM^{<F>}(x_{t-w}, \cdots, x_t, x_{t+1}) \tag{3}$$

$$P_{(t)}^{<B>} = LSTM^{<B>}(x_{t+1}, x_t, \cdots, x_{t-w}) \tag{4}$$

$$Q_t = \boldsymbol{W}_1 \times [P_{(t)}^{<F>}, P_{(t)}^{<B>}] + b_1 \tag{5}$$

where, $LSTM^{<F>}$ represents the forward LSTM network; x is the input data; w is the size of the time sliding window, assuming there are m pieces of training data, then $1 < w < m$; $P_{(t)}^{<F>}$ is the output result of the forward LSTM network. Similarly, $LSTM^{<B>}$ represents the reverse LSTM network, and $P_{(t)}^{<B>}$ is the output result of the reverse LSTM; after matrix splicing $P_{(t)}^{<F>}$ and $P_{(t)}^{<B>}$, it is multiplied by the weight matrix $\boldsymbol{W}_1$, and added to the bias term b_1 to obtain the first BiLSTM network output Q_t at time t.

3 Combination model

3.1 Model structure

The pseudocode of the model is illustrated in Algorithm 1.

Algorithm 1 MEEMD-DBA prediction algorithm

Input: input the data set X

Output: output the prediction result R

use MEEMD algorithm to decompose the data set X to get IMFs and residual components (Res).

calculate the energy values of IMFs and eliminate the noise of the IMFs.

merge the remaining IMFs and Res into data set A.

divide the data set A into train set S and test set T.

for $t = 0$ to epoch **do**:

  put S into the DBiLSTM and attention.

  use the Eq. (14) to calculate the loss values, and use the Adam optimizer to update the parameters of the network.

end for

fine-tuning the whole network and the training initialization parameters.

input test set T in MEEMD-DBA to generate the predicted value R.

return R

The prediction structure of the model at time t is shown in Fig. 2.

The detailed description of the model is as follows.

(1) Input the original time series $T[t]$.

(2) Use MEEMD algorithm to decompose $T[t]$ into N IMF sequences $I_m[t]$ ($m = 1, 2, \cdots, N$) and a residual sequence $R[t]$.

(3) Input the noise-removed IMFs and Res into DBiLSTM network for training, set the number of neurons in the hidden layer of DBiLSTM network as H, and at time t, the output of the mth IMF is

$$h(m, t) = [h_1, h_2, \cdots, h_H] \tag{6}$$

(4) The output of DBiLSTM network is used as input of the attention layer. In the layer, the gated recurrent unit (GRU) is used as the encoder, then at the time t, the weight coefficient $\beta(m,t)$ and output $O(m,t)$ of the mth IMF are as follows.

$$Q_m(t) = [h(m,T) \cdot h(m',t)] \tag{7}$$

$$\varepsilon(m,t) = \mathrm{softmax}[Q_m(t)] \tag{8}$$

$$\beta(m,t) = [h(m,t) \cdot \varepsilon(m,t)] \tag{9}$$

$$P(m,t) = [h(m,T), \beta(m,t)] \tag{10}$$

$$O(m,t) = \tanh[\boldsymbol{W}_t P(m,t) + b_t] \tag{11}$$

where, $h(m,T)$ represents the result of the last time step of the mth IMF sequence after passing through DBiLSTM network; $h(m', t)$ represents the result of the hidden layer generated by $h(m,t)$ after passing through the attention encoder; $Q_m(t)$ represents the dot result of $h(m,T)$ and $h(m', t)$; $\varepsilon(m,t)$ is the probability distribution value of attention, $\beta(m,t)$ is the weight coefficient matrix, $P(m,t)$ is the concatenation of $h(m,T)$ and the weight coefficient matrix of the attention layer. $O(m,t)$ is the final output of attention layer, $\boldsymbol{W}_t$ is the weight matrix, b_t is the bias.

(5) The output of attention layer is predicted through the dense layer, then the output result of the mth IMF component at time t is

$$y(m,t) = \mathrm{sigmoid}[\boldsymbol{W}_m O(m, t) + b_m] \tag{12}$$

where, $\boldsymbol{W}_m$ is the weight matrix, and b_m is the bias.

(6) Output the prediction result $R(t)$ at time t:

$$R(t) = \sum_{m=1}^{N} y(m, t) \tag{13}$$

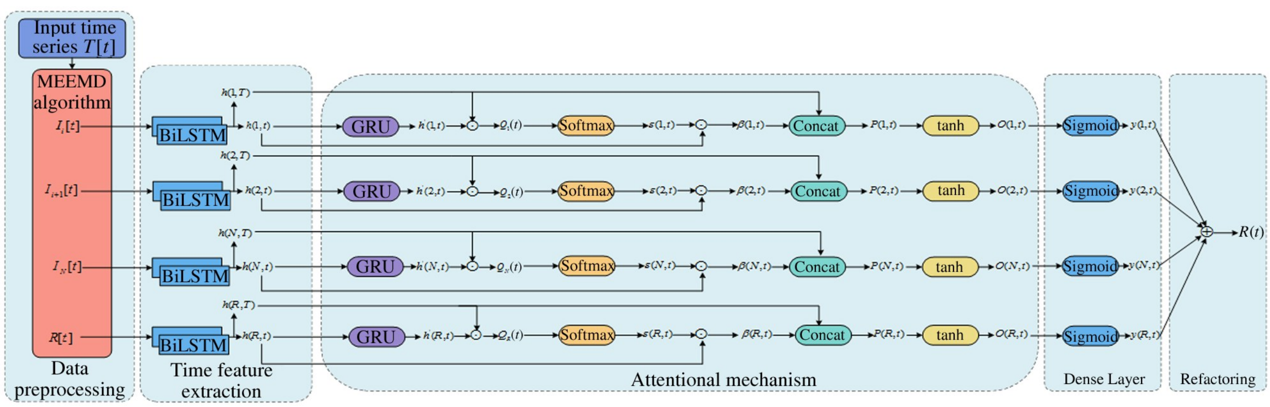


Fig. 2 Model structure

3.2 Loss function

The model selects Huber loss function as the optimization objective. The function expression is

$$L_{\delta}(y, \hat{y}) = \begin{cases} \frac{1}{2}(y - \hat{y})2 & |y - \hat{y}| \leqslant \delta \\ \delta |y - \hat{y}| - \frac{1}{2}\delta^2 & \text{otherwise} \end{cases} \quad (14)$$

where, y is the actual value, $\hat{y}$ is the predicted value, and δ is an adjustable hyperparameter. When δ is taken as 0, the loss function tends to mean absolute error (MAE), and when δ is taken as infinity, the loss function tends to in mean squared error (MSE).

3.3 Model evaluation indicators

In the experiment, goodness of fit R^2, root mean square error (RMSE), and MAE are selected as evaluation indicators. The closer the value of R^2 is to 1, the better the fitting degree of the model, and the smaller the values of RMSE and MAE, the better performance of the model. The objective functions of R^2, RMSE and MAE are as

$$R^2 = 1 - \frac{\sum_{i=1}^{n} (k_i - \hat{k}_i)^2}{\sum_{i=1}^{n} (k_i - \bar{k})^2} \quad (15)$$

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (\hat{k}_i - k_i)^2} \quad (16)$$

$$MAE = \frac{1}{n} \sum_{i=1}^{n} |\hat{k}_i - k_i| \quad (17)$$

where, $\hat{k}_i$ represents the predicted value, k_i represents the actual value, $\bar{k}$ is the average value of the actual value, and n is the number of samples.

4 Experimental analysis

4.1 The experimental data

The experimental data adopts three urban road traffic datasets in Minnesota. The dataset contains traffic volume (vol) data and lane occupancy (occ) data from January 12, 2018 to January 12, 2019. In this experiment, the traffic flow data in December 2018 was selected for the experiment, the sampling interval is 5 min, and 8928 continuous data samples are obtained. The training set and test set are divided in a ratio of 8 : 2.

4.2 Model building

The experiment is based on the Keras neural network, and the network is built and trained in Python 3.6. The parameter settings[16] of the combined model are shown in Table 1. Among them, *Epoch* is the number of times of training, and one training represents a forward propagation and back propagation calculation process completed by all training samples; *batch_size* represents the number of samples selected for each training, and the size of this value will affect the prediction accuracy of the model and convergence speed; *G_Hidden_Layer* represents the number of hidden layer neurons of DBiLSTM; *A_G_Hidden_Layer* represents the number of hidden layer neurons of the attention layer encoder GRU.

Table 1 Model parameter setting

Parameter name	Parameter value
Epoch	100
batch_size	64
G_Hidden_Layer	100
A_G_Hidden_Layer	100

4.3 MEEMD decomposition results

MEEMD is used to decompose the data, and the parameters in Table 2 should be set. Among them, *Nstd* is the amplitude of adding white noise. If the value is set too large, the noise signal will mask the information of the original data. If the value is set too small, the distribution of the maximum and minimum values of the original data can not be changed, resulting in the algorithm unable to effectively solve the mode mixing problem; *Ne* is the logarithm of the added white noise; α represents the permutation entropy embedding dimension; θ_0 is the permutation entropy threshold. According to Ref. [14], the parameter settings of MEEMD are shown in Table 2.

Table 2 MEEMD parameter settings

Parameter name	Parameter value
Nstd	0.6
Ne	25
α	6
θ_0	0.6

(1) Orthogonality analysis of IMF components

Ref. [17] points out that through modal decomposition of the original data, the components should be locally orthogonal to each other, and the orthogonality index *IO* is defined as

$$IO = \sum_{t=0}^{T} \left[\sum_{i=1}^{n+1} \sum_{j=1}^{n+1} C_i(t) C_j(t) / X^2(t) \right] (i \neq j) \quad (18)$$

where, T is the length of the sequence, $C(t)$ is the

IMF component obtained by decomposition, and $X(t)$ is the original sequence. Eq. (18) is the overall orthogonality index. The closer the IO value is to 0, the stronger the orthogonality of the components is. Ref. [18] gives the following conclusions: if there is no mode mixing phenomenon in IMF components after decomposition, the components are orthogonal; otherwise, they are not orthogonal.

According to this conclusion, this paper calculates the IO values of the components obtained after EMD, EEMD and MEEMD decomposition according to Eq. (18), which are shown in Fig. 3.

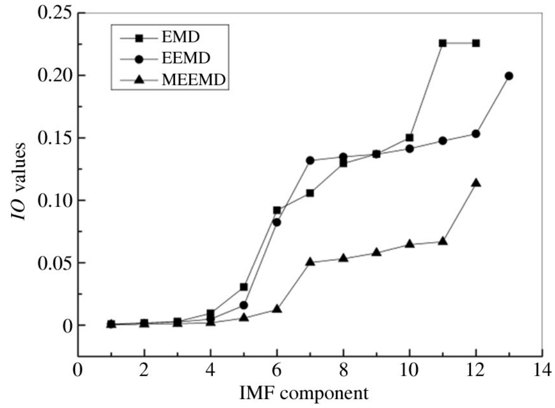


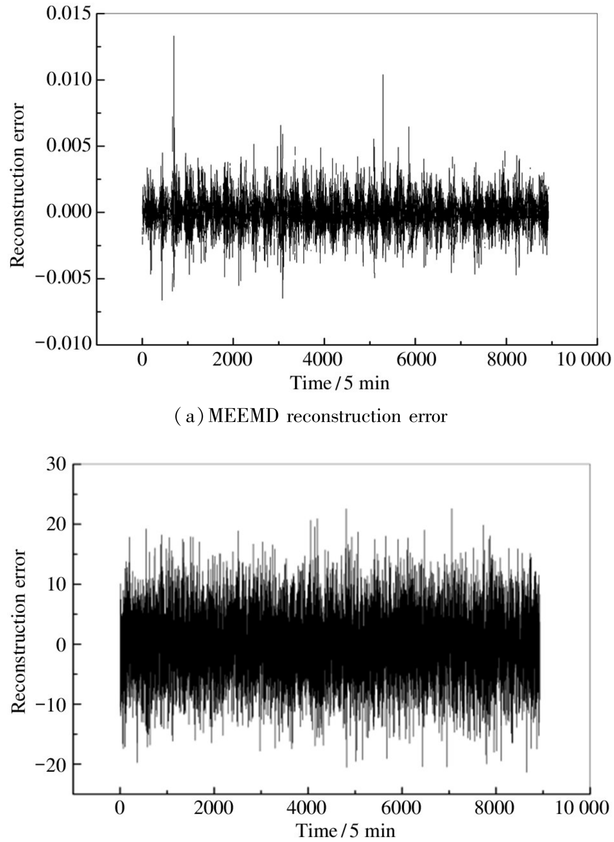
Fig. 3 Model structure

After EMD algorithm, the IO value is 0.112 86, and the mode mixing phenomenon occurs; after EEMD decomposition, the IO value is reduced to 0.099 77; after MEEMD decomposition algorithm, the IO value is 0.056 65, and the mode mixing problem is solved. From Fig. 3 and the IO values obtained by decomposing data by each mode decomposition algorithm, it can be seen that MEEMD algorithm has the best effect of suppressing mode mixing.

(2) Mode decomposition completeness analysis

The completeness of the mode decomposition, that is, the reconstruction error of the data, is the sum of the IMFs and Res obtained by the mode decomposition algorithm and compared with the original data. EEMD decomposition and reconstruction error and MEEMD decomposition and reconstruction error are shown in Fig. 4.

MEEMD algorithm calculates the permutation entropy of the obtained components after each round of decomposition, if the conditions are met, it stops adding white noise to the original sequence. At this time, the impact of white noise on the data will be offset by multiple sets of averages. The reconstruction error is small, between -0.0075 and 0.0125, which is closer to the original traffic flow data; EEMD algorithm needs to set the amplitude, logarithm of the white noise and the ensemble number. If the ensemble number is set improperly, then it can not solve the problem of mode mixing problem, and may generate redundant components which causes the poor reconstruction. After experiments, the reconstruction error of EEMD algorithm is between -20 and 20.



(a) MEEMD reconstruction error

(b) EEMD reconstruction error

Fig. 4 Reconstruction error comparison

Based on the above analysis and experimental results, EEMD is difficult to determine the hyper-parameters, and the reconstruction performance and the IMFs orthogonality of MEEMD are better than EEMD, so MEEMD is selected to preprocess the data.

4.4 Result analysis

(1) Determination of hyperparameters of Huber loss function

When the hyper-parameters in Huber loss function take different values, the experimental results are also different. As shown in Table 3, different values are chosen for comparison, the experimental model is MEEMD-BiLSTM-attention, and the model parameter settings are the same as those in Table 1.

The best result can be obtained when δ is taken as 0.5, as shown in Table 3.

Table 3 The effect of different δ on the model accuracy

δ values	Evaluation indicators		
	RMSE	*MAE*	R^2
0.1	6.58	4.97	0.984 60
0.5	6.55	5.05	0.984 76
1.0	6.58	5.08	0.984 60
1.5	6.85	5.28	0.983 33

(2) Comparison of results of different loss functions

In the experiment, MSE, MAE and Huber loss functions are chosen for comparison. The experimental results are shown in Table 4.

Table 4 Comparison of loss functions

Loss functions	MEEMD-BiLSTM-attention evaluation indicators		
	RMSE	*MAE*	R^2
Huber	6.55	5.05	0.984 76
MSE	6.64	5.17	0.984 31
MAE	6.81	4.78	0.983 48

The prediction results obtained by selecting Huber as the loss function are better, which is shown in the experimental results.

MSE loss function is more sensitive to the abnormal data because the error is squared, but the function can converge well when the error is small; as for MAE loss function, according to its function expression, it can be seen that the function is a sectional function, and the function is discontinuous, so there will be problems when back-propagation derivation is performed, and the update gradient of the loss function will not change, no matter the error is large or small, it is always updated with the same gradient; Huber loss function is combined with the advantages of MAE and MSE loss functions, suitable functions can be selected to optimize the weights for different situations.

(3) Multi-layer BiLSTM experimental results

In order to strengthen the model's extraction of temporal features of traffic flow data, a multi-layer BiLSTM network is set up in the model. The comparative experiments which is to determine the appropriate number of BiLSTM network layers can be found in Table 5.

Table 5 Multi-layer BiLSTM experimental results

Different Layers	Evaluation indicators		
	RMSE	*MAE*	R^2
1 layer	6.55	5.05	0.984 76
2 layers	6.29	4.83	0.985 95
3 layers	6.30	4.97	0.985 89

When it is set to 2 layers, the model has strong temporal feature extraction ability, as shown in Table 5. This is because with the layer of BiLSTM increasing, the feature extraction ability of the model is gradually strengthened, but when the number of layers increases to a certain level, the stability of the model decreases, the ability to extract temporal features gradually weakens, and the training time will also be affected when the number of network layers is deepened and lengthened, so this paper sets BiLSTM layers to be 2.

(4) Comparison of prediction results of different models

The comparison models selected in the experiment are shown in the following table. The parameter settings of all combined models are the same as those in Table 1, and the parameter settings of MEEMD algorithm are shown in Table 2. EEMD algorithm adds white noise amplitude and adds the same logarithm as MEEMD. Table 6 shows the prediction accuracy of different models obtained from the experiment.

MEEMD-DBA has a higher accuracy when compared with the other models, as shown in Table 6.

Table 6 Model comparison

Model	Evaluation indicators		
	RMSE	*MAE*	R^2
EMD + BiLSTM	7.16	5.55	0.981 78
EMD + BiGRU	7.38	5.89	0.980 65
EEMD + BiLSTM	7.23	5.75	0.984 05
EEMD + BiGRU	7.48	5.96	0.981 02
MEEMD + BiLSTM	6.80	5.29	0.983 57
MEEMD + BiGRU	7.01	5.35	0.982 52
MEEMD-DBA	6.29	4.83	0.985 95

In EMD algorithm, due to the mode mixing problem, the high and low frequency components can not be accurately separated, which leads to the problem of low prediction accuracy; EEMD algorithm needs to add white noise with the same amplitude and opposite positive and negative values to make the extreme values of the data approximately evenly distributed. Thereby, the problem of mode mixing problem is solved, but because the algorithm can not neutralize all the added white noise, the error is large when calculating the prediction accuracy compared with the original data. Since the traffic flow has time correlation, so the influence of the before and after traffic flow on the current traffic flow needs to be considered at the same time, and the unidirectional network structure can not make an accurate prediction of the traffic flow. In this paper, the temporal features are extracted by using DBiLSTM and attention mechanism.

Fig. 5 shows the optimal ratio of the prediction performance, which is MEEMD-DBA compared with the other comparative models. The calculation formula is

$$P = \frac{\hat{R} - R}{\hat{R}} \times 100\% \tag{19}$$

where, P is the optimization ratio. If P is a positive value, it indicates that MEEMD-DBA is improved compared with the comparison model's predictive performance. If P is negative, it indicates that the model's prediction performance in this article is poor. R is the performance evaluation index of MEEMD-DBA, and $\hat{R}$ is the performance evaluation index of the comparison model.

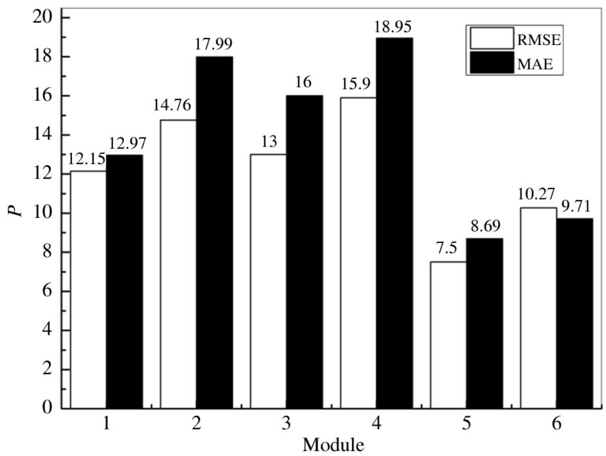


Fig. 5 Comparison of optimization results

In Fig. 5, the abscissas 1 to 6 correspond to the comparison models in Table 6. The p value indicates that MEEMD-DBA model has a better predict performance.

(5) Noise energy analysis

Through the mode decomposition algorithm, noise and high, medium and low frequencies are separated. If the noise component is removed during model training, the performance of the model will be improved. The experiment is based on the minimum energy criterion[19]. When the signal contains noise, its frequency band is higher. When a usable signal appears, the law of decreasing energy of IMF component will be broken, and a local energy pole will be generated. After the minimum point, the useful signal will dominate the energy of each IMF component instead of noise. The energy E of IMF component is

$$E_m = \sum_{i=1}^{P} [I_m(i)]^2 \tag{20}$$

where, E_m represents the energy value of the mth IMF component, and $I_m(i)$ represents the ith data in the mth IMF component.

According to Eq. (20), the square root energy value of each IMF is obtained as shown in Fig. 6.

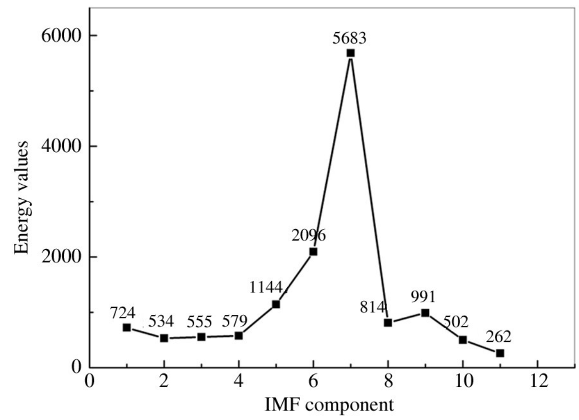


Fig. 6 IMF energy diagram

The energy values from IMF1 to IMF2 are decreasing, and after IMF2, the energy value begins to increase. Therefore, the components before IMF3 are considered to be noise components and are not considered during model training. In this paper, the following comparative experiments are carried out, as shown in Table 7.

Table 7 Remove different IMFs

Model	Evaluation indicators		
	RMSE	*MAE*	R^2
MEEMD-DBA	6.29	4.83	0.985 95
MEEMD-DBA (remove IMF1)	2.54	1.82	0.997 70
MEEMD-DBA (remove IMF1, IMF2)	1.21	0.84	0.999 48

As shown in Table 7, after removing IMF1 and IMF2, the prediction results are reconstructed, and the prediction accuracy reaches the best. Compared with the reconstruction of all the results, RMSE is optimized by 80.76%.

(6) Multi-step prediction results

The following experiments are done to verify the accuracy of the model's long-term prediction. Using the data sets 1 h to predict the 15 min, 30 min, 45 min and 1 h data, the results are shown in Table 8.

Table 8 Comparison of multi-step forecast results

Step	MEEMD-DBA evaluation indicators		
	RMSE	*MAE*	R^2
15 min	3.82	2.95	0.994 81
30 min	5.29	3.75	0.990 08
45 min	8.16	5.47	0.976 52
1 h	13.65	10.76	0.934 44

As shown in Table 8, with the prediction step increasing, the performence of the model gradually decreases. Since the training set data is divided by the sliding window strategy, when the step size is small, the distance from the historical data is closer, more information can be used, and the prediction accuracy is also higher; when the step size is increased, the information which can be used is reduced, so the prediction accuracy is lower.

(7) Prediction results by time period

The characteristics of traffic flow data are quite different between weekdays and weekends. This paper separates weekday data from weekend data in the data used in the experiment. There are 6048 pieces of processed data on weekdays and 2880 pieces of data on weekends. The same number of weekdays and weekends are selected to show the difference between them more directly, as shown in Fig. 7.

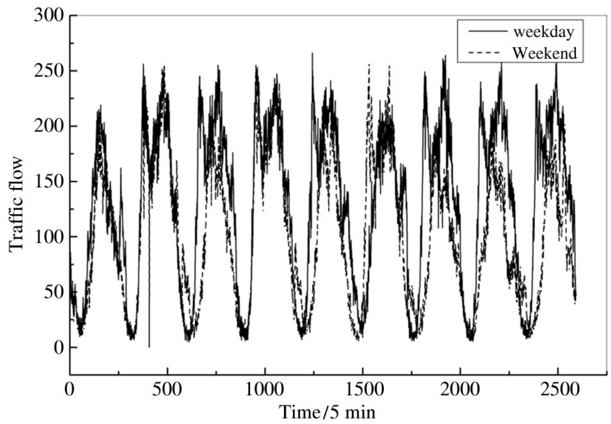


Fig. 7 Weekday and weekend data

The following experiments are done to verify the effectiveness of the model proposed in this paper.

MEEMD decomposition algorithm is used to decompose the data, and the parameter settings of the algorithm are the same as those in Table 2. After the decomposition is completed, the energy value of each IMF component is calculated according to Eq. (20). After calculation, IMF1 and IMF2 are noises in the data. After removing the noise, the model is trained. The experimental results are shown in Table 9.

MEEMD-DBA model has the best prediction effect on weekdays and weekends as shown in Table 9. Due to the lack of data in the weekend period, the single-layer BiLSTM network can not sufficiently extract temporal features. The multi-layer BiLSTM network proposed in this paper can fully extract temporal features when the amount of data is small. Combined with the attention mechanism, the features that have an impact on the prediction results are given greater weights, so the prediction accuracy is significantly improved.

During weekdays, traffic participants increase significantly, and there will be double peaks in a specific time period, that is, morning peak and evening peak. Therefore, within the same time period, the peak value of traffic flow on weekdays is greater. Because the traffic flow data on weekdays is more, the model training is more sufficient, so the accuracy of traffic flow prediction on weekdays is better.

Table 9 Experimental results

Model	*RMSE*		*MAE*		R^2	
	Weekday	Weekend	Weekday	Weekend	Weekday	Weekend
MEEMD-DBA	1.24	1.78	1.00	1.45	0.999 53	0.998 87
EEMD + BiLSTM + attention	6.20	6.77	4.92	5.41	0.988 18	0.983 69
EMD + BiLSTM + attention	1.91	2.62	1.63	2.25	0.998 87	0.997 57
MEEMD + BiLSTM + attention	1.51	2.21	1.26	1.80	0.999 30	0.998 27

(8) Ablation experiment

The ablation experiment for analyzing the influence of MEEMD, the attention mechanism and DBiLSTM on the prediction effect of the combined model is done to verify the effectiveness of the model proposed in this paper, as shown in Table 10.

Table 10 Ablation experiment

Model	Evaluation indicators		
	RMSE	*MAE*	R^2
MEEMD-DBA	1.21	0.84	0.999 48
BiLSTM + attention	12.13	8.99	0.947 65
MEEMD + BiLSTM	2.97	2.94	0.996 85
MEEMD + attention	4.42	3.46	0.993 06

The MEEMD-DBA combined model proposed in this paper has the best effect, as shown in Table 10, indicating that each algorithm has a greater impact on the prediction effect of the combined model. MEEMD algorithm can effectively remove the noise existing in the traffic flow data, separate the data with different trends, and eliminate the influence of the non-stationary sequence on the model stability. Therefore, the prediction effect of MEEMD-DBA is better than BiLSTM + attention, and its RMSE is optimized by 75.51%. The attention mechanism can adaptively strengthen the attention to the traffic flow sequence, assign a larger weight to the sequence that affects the prediction result of the model, and strengthen its training. Therefore, MEEMD-DBA has better prediction effect than MEEMD + BiLSTM, and its RMSE is optimized by 59.25%; BiLSTM model can combine the traffic flow data before and after at the same time to extract temporal features, so MEEMD-DBA has better prediction effect than MEEMD + attention, and its RMSE is optimized by 72.62%.

Based on the above experiments, the MEEMD-DBA combination model proposed in this paper achieves good results in short-term traffic flow prediction by combining MEEMD, DBiLSTM and attention mechanism.

5 Conclusion

A combined MEEMD-DBA model is proposed to predict short-term traffic flow. MEEMD algorithm effectively separates the noise in the data, and solves the problem of poor reconstruction of EEMD. DBiLSTM network strengthens the model's extraction of time series features, especially when the amount of data is small. The attention mechanism finds out the factors that have a greater impact on the prediction results through calculation and assigns greater weights to them. The experimental results show that MEEMD-DBA model has higher accuracy compared with other models under the same dataset, and can well predict weekdays and weekends. In the future, the influence of various factors on traffic flow forecast will be further explored.

References

[1] BOUKERCHE A, TAO Y, SUN P. Artificial intelligence-based vehicular traffic flow prediction methods for supporting intelligent transportation systems [J]. Computer Networks, 2020,182:1-21.

[2] BOUKERCHE A, WANG J. Machine learning-based traffic prediction models for intelligent transportation systems [J]. Computer Networks, 2020, 181: 1-15.

[3] YAN H, FU L, QI Y, et al. Learning a robust classifier for short-term traffic state prediction [J]. Knowledge-Based Systems, 2022, 242: 1-15.

[4] REN Y, JIANG H, JI N, et al. TBSM: a traffic burst-sensitive model for short-term prediction under special events[J]. Knowledge-Based Systems, 2022,240: 1-16.

[5] CHEN X Q, CHEN H X, YANG Y S, et al. Traffic flow prediction by an ensemble framework with data denoising and deep learning model[J]. Physica A: Statistical Mechanics and its Applications, 2021, 565: 1-11.

[6] LU W Q, RUI Y K, RAN B, et al. Traffic flow prediction based on hybrid deep learning under connected and automated vehicle environment[J]. Journal of Transportation Systems Engineering and Information Technology, 2020,20(3):47-53. (In Chinese)

[7] ZHANG Y P, WANG S H, XIA K J, et al. Alzheimer's disease multiclass diagnosis via multimodal neuroimaging embedding feature selection and fusion[J]. Information Fusion, 2021, 66:170-183.

[8] JIANG Y, ZHANG Y, LIN C, et al. EEG-based driver drowsiness estimation using an online multi-view and transfer TSK fuzzy system[J]. IEEE Transactions on Intelligent Transportation Systems, 2020, 22 (3): 1752-1764.

[9] JIANG Y, GU X, WU D, et al. A novel negative-transfer-resistant fuzzy clustering model with a shared cross-domain transfer latent space and its application to brain CT image segmentation [J]. IEEE/ACM Transactions on Computational Biology and Bioinformatics, 2020,18(1): 40-52.

[10] WANG J J, CHEN Q K. Traffic prediction model based on spatiotemporal attention network[J]. Journal of Chinese Computer Systems, 2021,42(2): 303-307. (In Chinese)

[11] TA X, LIU Z, HU X, et al. Adaptive spatio-temporal graph neural network for traffic forecasting[J]. Knowledge-Based Systems, 2022, 242: 1-13.

[12] JAMES J Q. Citywide traffic speed prediction: a geometric deep learning approach[J]. Knowledge-Based Systems, 2021, 212: 1-13.

[13] ZHAO L, ZHOU Y, LU H, et al. Parallel computing method of deep belief networks and its application to traffic flow prediction[J]. Knowledge-Based Systems, 2019, 163: 972-987.

[14] ZHEN J D, CHEN J S, YANG Y. Modified EEMD algorithm and its applications[J]. Journal of Vibratin and Shock, 2013(21):21-26.

[15] CHEN Z J, DING Y F. A hybrid MEEMD-GAELM based model for short-term wind power prediction[J]. Water Resources and Power, 2020,38(8):207-210.

[16] GUI Z M, LI Z Z, GUO L M. Short-term traffic flow prediction based on ACGRU model[J]. Computer Engineering and Applications, 2020,56(21):260-265.

[17] HUANG N E, SHEN Z, LONG S R, et al. The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis[J]. Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences, 1998, 454 (1971): 903-995.

[18] LI X B. Research on orthogonality of EMD method in HHT[D]. Kunming: Kunming University of Science and Technology, 2010. (In Chinese)

[19] MA X, HAO Y N. Resarch on empirical mode decomposition ans denoise method[J]. Science and Technology Vision, 2018(23):72-73.

ZHANG Xijun, born in 1980. He received his Ph. D, M. S. and B. E. degrees in Lanzhou University of Technology in 2021, 2009 and 2003 respectively. His research interests include big data processing, data mining and analysis, embedded systems, machine learning and intelligent transportation system.