



A New Time Series Forecasting Model Based on Complete Ensemble Empirical Mode Decomposition with Adaptive Noise and Temporal Convolutional Network

Chen Guo¹ · Xumin Kang¹ · Jianping Xiong² · Jianhua Wu¹ 

Accepted: 19 September 2022

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2022

Abstract

In this paper, a new hybrid time series forecasting model based on the complete ensemble empirical mode decomposition with adaptive noise (CEEMDAN) and a temporal convolutional network (TCN) (CEEMDAN-TCN) is proposed. The CEEMDAN is used to decompose the time series data and the TCN is used to obtain a good prediction accuracy. The effectiveness of the model is verified in univariate and multivariate time series forecasting tasks. The experimental results indicate that compared with the long short-term memory model and other hybrid models, the proposed CEEMDAN-TCN model shows a better performance in both univariate and multivariate prediction tasks.

Keywords Time series forecasting · Temporal convolutional network · Ensemble empirical mode · Hybrid model

1 Introduction

Time series (or time sequence) is a significant type of data, which is defined as a series of numerical values of an occurrence in time order. It is widely distributed in many fields, such as clinical medicine [1], power science [2], meteorology [3], finance [4], etc. Time series affects every aspect of our life, and its analysis and prediction has become a research hotspot.

Time series prediction is to model the regular pattern between historical data and future values through the prediction model on the premise of the principle of continuity assumption, so as to achieve the quantitative estimation of the predicted values. Time series forecasting is of great importance in industrial application and daily life. For example, the accurate prediction of power load data can not only facilitate the rational arrangement of energy production, but also avoid the waste caused by the overproduction of electricity [5]. Short-term traffic flow prediction can be used to optimize the urban traffic system and reduce the likelihood of traffic accidents [6]. Therefore, an exact time sequence prediction is of great

✉ Jianhua Wu
jhwu@ncu.edu.cn

¹ School of Information Engineering, Nanchang University, Nanchang 330031, China

² Industrial Center, Shenzhen Polytechnic, Shenzhen 518055, China

value in many application scenarios. The current prediction models are facing two challenges: first, there are time correlations and dependencies between variables in the input sequence. How to fully extract and map such complex sequence dependencies with a high-precision prediction model needs to be solved. Second, with the development of the era of big data, the time series is becoming larger in scale and more diverse, and the data presents complex characteristics such as nonlinear, non-stationary and irregularity. This makes it harder for the model to extract dependencies between data.

The basic prediction models can be roughly divided into three types: traditional statistical methods, machine learning methods and deep learning methods. In the statistical methods, the Autoregressive Integral Moving Average (ARIMA) is one of the most classic models. Although the ARIMA is applicable to the prediction of relatively stable and linear data, it cannot effectively forecast the actual field data accurately. After that, machine learning methods, such as Bayesian method, Support Vector Machine (SVM) and Support Vector Regression (SVR), were used for the prediction. Machine learning methods can learn mapping rules from a large amount of historical data to build nonlinear prediction models. Therefore, through iterative training and learning approximations, the machine learning methods can achieve a higher prediction accuracy than the statistical methods. However, the prediction performance of machine learning methods is very dependent on delicate and tedious feature engineering, which makes it unable to obtain a relatively ideal prediction accuracy in many application scenarios. In recent years, deep learning has become more and more popular in time series prediction task because of its powerful feature extraction ability and excellent generalization performance. As we know, the recurrent neural network (RNN) in deep learning methods is considered to be one of the most classical sequence processing networks. RNN is a kind of neural network with feedback structure; it obtains the characteristic information of sequence data in the time direction by passing down the state information of hidden layer, that is, the dependence information between data. Because of this special structure, RNN model has better performance in processing text, speech and other sequence data. However, when the length of the input sequence increases, RNN needs to add more hidden layers, that is, increase the number of network layers, which makes RNN prone to problems such as gradient disappearance and affects the prediction accuracy of the model. Aiming at the deficiency of RNN, an improved model, long short-term memory (LSTM) network, was proposed. LSTM utilizes three gate structures to process the information and introduces Sigmoid activation function to limit the range of output values, so as to alleviate the problem of gradient disappearance and improve the prediction accuracy. In general, LSTM is considered to be one of the state-of-the-art methods for dealing with sequence problems. However, the defects of LSTM are obvious. Although the gate structure in LSTM can alleviate the gradient vanishing and explosion problems, these problems still exist when dealing with very long sequences. What's more, each cell in LSTM is equivalent to a fully connected (FC) layer, which means that each LSTM unit needs four FC layers. When the time span is large, the quantity of computation is enormous, which raises the difficulty of model training. Recently, a special CNN, temporal convolutional network (TCN), overcomes these shortcomings and performs better than LSTM in temporal sequence prediction tasks.

However, the aforementioned methods only emphasize the prediction module of the model, thereby ignoring the influence of data and lacking an effective data preprocessing. In order to alleviate the effect of data fluctuation, an appropriate data preprocessing is necessary.

In view of the above mentioned challenges, we propose a new hybrid model based on CEEMDAN algorithm and multiple TCNs, called CEEMDAN-TCN model, to complete the time series prediction task. The CEEMDAN algorithm can process nonlinear and non-stationary time sequences adaptively. The original time sequence is decomposed into a certain

number of components with various frequencies by means of CEEMDAN algorithm, and then these components are respectively used as the inputs of the multiple TCNs. Finally, all the outputs from the multiple TCNs are integrated to form the ultimate forecast outcome. The simulation results indicate that in contrast with the LSTM model and other hybrid models, the proposed CEEMDAN-TCN model shows a better performance in both univariate and multivariate forecasting tasks.

2 Related Work

In the field of time sequence forecasting, researchers have done a lot of research work, proposed many different prediction schemes, and conducted experimental verification on various time series data such as wind speed data, traffic flow data etc. Finally, with continuous research and practice, the prediction accuracy has been greatly improved. Generally speaking, the research can be roughly divided into four aspects: statistical methods, machine learning methods, deep learning methods and hybrid methods.

In the statistical methods, the ARIMA and its variants are widely studied. For instance, Haneen et al. [7] proposed a forecasting model based on ARIMA to predict the COVID-19 spread. In this method, the accuracy of ARIMA model was investigated and validated over a relatively long period of time using Kuwait as a case study and the experimental results showed that the actual values were within bounds of prediction at 95% confidence interval. Mao et al. [8] applied a seasonal auto-regressive integrated moving average (SARIMA) model to forecast the incidence of tuberculosis in China and the result showed that the SARIMA is a useful tool for monitoring epidemics. However, the ability of statistical methods to deal with complex and nonlinear data is limited.

Compared with the statistical methods, the machine learning methods can process and model more complex data, among which SVR and SVM are commonly used models. For example, Nieto et al. [9] utilized four different models, including SVM, ARIMA, vector autoregressive moving-average (VARMA) and multilayer perceptron (MLP), to predict the value of PM10 in northern Spanish city. The simulations showed that the SVM model performs better than the other models when forecasting one month ahead and also for the following seven months. Liu et al. [10] proposed the network traffic forecast model of SVR algorithm optimized by global artificial fish swarm algorithm (GAFSA). In this method, the optimum training parameters used for SVR could be calculated by GAFSA, which made the forecast more accurate and the prediction results of the proposed model is more stable with an improved precision of more than 89%. Although the machine learning method has a good effect in some prediction tasks and has a solid mathematical theoretical foundation as support, its effect depends on expert experience and manual feature selection, which affect its practical values.

In recent years, models on the basis of deep learning are gaining popularity. Among them, long short-term memory (LSTM) and its related models are of prominent importance. Sun et al. [11] adopted an LSTM model to predict the ionospheric vertical total electron content (TEC) sequence over Beijing and the obtained root mean square error (RMSE) between the predicted values and the ground-truths was 30% lower than a multi-layer perceptron (MLP) network. Hu et al. [12] proposed a model based on a variant of LSTM, in which the forget and input gates were combined into an update gate, and a Sigmoid layer was utilized to control the information update. In this way, the parameters of the model were reduced and the memory module was simpler. Li et al. [13] applied a model which utilizes an

evolutionary attention-based LSTM (EA-LSTM) training with competitive random search in multivariate prediction task. This model introduces an evolutionary attention mechanism to transfer the shared parameters, which can confirm the pattern for importance-based attention sampling during temporal relationship mining. The experimental results have illustrated that the proposed model can achieve competitive prediction performance compared with other baseline methods. Lin et al. [14] proposed a novel end-to-end hybrid neural network, called TreNet, to learn local and global contextual features for predicting the trend of time series. TreNet consists of a convolutional neural network (CNN) and an LSTM network. This model exploited the CNN to obtain local information and the LSTM to get the long-term dependence of data. In [15], Ding et al. focused on the need of flood prediction and utilized an interpretable Spatio-Temporal Attention LSTM (STA-LSTM) model. This model was able to enhance the ability of LSTM to capture data features. After the LSTM model, Bai et al. [16] proposed the TCN model has received an extensive attention due to its excellent performance. Yin et al. [17] proposed a multi-temporal-spatial-scale convolutional network in power load forecasting tasks. The experimental results proved that the model was able to obtain a higher accuracy than other short-term power load prediction models. Zhu et al. [18] explored the TCN model to predict the wind power data sequences. The designed method solved the problems of long-term dependencies and reduced the performance degradation of deep convolutional model in sequence prediction by dilated causal convolutions and residual connections. Lin et al. [19] investigated the application of TCN for solar power forecasting and the simulation results showed that the performance of the proposed model is enhanced in contrast with the LSTM and gated recurrent unit (GRU) models. Wang et al. [20] applied a TCN model for short-term load forecasting. The experimental results showed that the TCN model achieved a higher forecast precision with less training time and computational memory. The model's evaluation index, RMSE, decreased by 51 and 28% compared with SVM and LSTM, respectively.

The study found that using the primary time series data directly to establish prediction model is subject to substantial errors and the characteristics of original data must be fully analyzed and preprocessed. Therefore, the ensemble empirical mode decomposition (EEMD) and its advanced version, the complete ensemble empirical mode decomposition with adaptive noise (CEEMDAN) [21, 22], were successively proposed, and further, these algorithms were combined with the deep learning methods to form a hybrid model. Chen et al. [23] applied a combination model to enhance the forecast accuracy of financial time sequences. This model consisted of an empirical mode decomposition (EMD) and an attention-based LSTM model, obtaining an RMSE decrease of 36% compared with the original LSTM. Xuan et al. [24] established a novel prediction model based on EMD, LSTM and Cubic Spline Interpolation (CSI) to improve the accuracy and efficiency of the short-term stock price trend prediction. Cao et al. [25] utilized the CEEMDAN algorithm and LSTM to forecast the stock market prices. Compared with support vector machine (SVM) [26–28] and MLP network [29, 30], the experimental results indicated that Cao's algorithm obtained a better prediction in one-step-ahead forecasting tasks.

3 Methodology

3.1 CEEMDAN Algorithm

The fluctuation and complexity of the original time sequence will affect the function-fitting and subsequent convergence of the TCN model, thus limiting the prediction accuracy of the

TCN model. In response to this difficulty, the original nonlinear and non-stationary time sequence is preprocessed by the CEEMDAN algorithm.

Early in 1998, Huang et al. [31] proposed an adaptive signal time–frequency processing method, called EMD algorithm, to decompose a time sequence according to the time scale information without pre-setting of any basis function. In EMD, the original time series can be broken up into a certain amount of intrinsic mode functions (IMFs) and a residual (Res). Different IMF components represent different features of the original data at different time scales [32]. The IMF is regarded as a specific factor that affects the time series, while the Res determines the overall trend of the signal. Although EMD algorithm has an edge over dealing with time series, the “mode mixing” is its defect [33]. Mode mixing refers to the existence of very analogous oscillations in different IMF components or very different amplitudes in an IMF. It can cause the adjacent IMF waveforms to be aliasing and not able to represent the real physical process. By adding a white Gaussian noise to the signal, the EEMD algorithm is able to eliminate the mode mixing in EMD [34]. However, eliminating the noise needs a repeated calculation of average, which spends a considerable amount of computation time. Moreover, the reconstructed signal including the residual noise will cause reconstruction errors and the noisy signal may produce a number of additional IMF components.

To solve the aforementioned problems, P. Flandrin et al. put forward the CEEMDAN algorithm [35]. The CEEMDAN algorithm is a modified form of EEMD with several advantages. It can eliminate mode mixing more efficiently, with a reconstruction error being almost zero and a considerable reduction of computing time [36]. The flowchart of CEEMDAN is shown in Fig. 1 and the decomposition procedure can be given as follows:

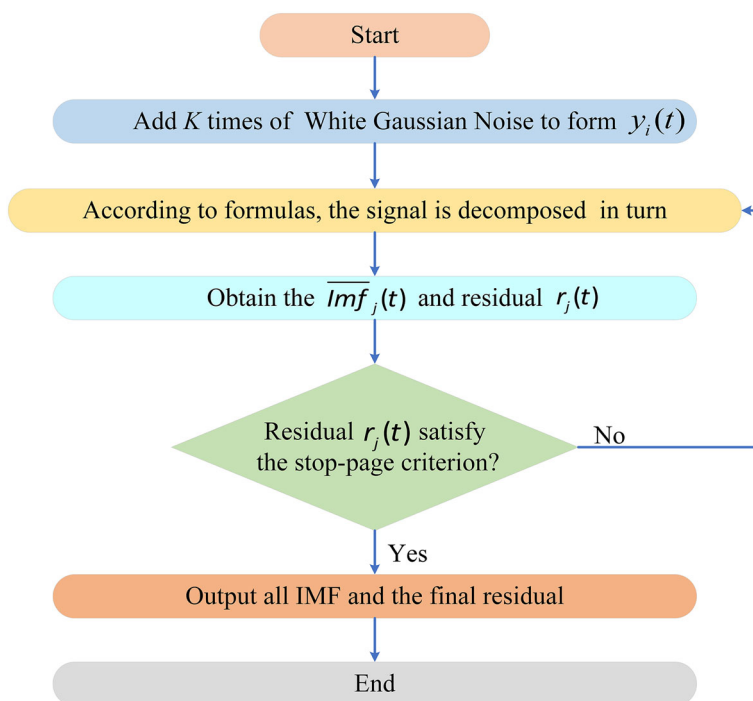


Fig. 1 Flow chat of the CEEMDAN algorithm

- (1) Suppose one has a signal $x(t)$, $t = 0, 1, \dots, T - 1$, into which a white Gaussian noise with zero mean and unit variance, $N(0, 1)$, is added:

$$y_i(t) = x(t) + \varepsilon_0 \delta_i(t), i = 1, 2, \dots, K \quad (1)$$

where K is the number of white Gaussian noises, ε_0 is a coefficient of intensity, $\delta_i(t)$ represents the i th realization of a random Gaussian process. From (1), K noisy versions of signals are obtained.

- (2) The first IMF, $\overline{Imf}_1(t)$, is calculated by averaging all the first decomposition components of EMD:

$$\overline{Imf}_1(t) = \frac{1}{K} \sum_{i=1}^K EMD_1(y_i(t)) \quad (2)$$

$$r_1(t) = x(t) - \overline{Imf}_1(t) \quad (3)$$

where $EMD_1(\cdot)$ represents the first IMF component generated by EMD algorithm, $r_1(t)$ is the residual for the first stage.

- (3) The signal $r_1(t) + \varepsilon_1 EMD_1(\delta_i(t))$ can be further decomposed by EMD and combined to obtain the second IMF component:

$$\begin{cases} \overline{Imf}_2(t) = \frac{1}{K} \sum_{i=1}^K EMD_1(r_1(t) + \varepsilon_1 EMD_1(\delta_i(t))) \\ r_2(t) = r_1(t) - \overline{Imf}_2(t) \end{cases} \quad (4)$$

where $r_2(t)$ is the residual of the second stage.

The j th IMF component and j th residual can be computed as:

$$\begin{cases} \overline{Imf}_j(t) = \frac{1}{K} \sum_{i=1}^K EMD_1(r_{j-1}(t) + \varepsilon_{j-1} EMD_{j-1}(\delta_i(t))) \\ r_j(t) = r_{j-1}(t) - \overline{Imf}_j(t) \end{cases} \quad (5)$$

where $EMD_{j-1}(\cdot)$ means the $(j - 1)$ th IMF component obtained by EMD algorithm, $r_j(t)$ is the residual after the j th decomposition.

- (5) $j \rightarrow j + 1$, Eq. (5) is repeated until the residual $r_j(t)$ satisfies a preset end standard:

$$\sum_{t=0}^{T-1} \frac{|r_{j-1}(t) - r_j(t)|^2}{r_{j-1}^2(t)} \leq D_s \quad (6)$$

where D_s is an empirical threshold value [37]. Suppose $j = N$ when the decomposition procedure stops.

Finally, the time sequence $x(t)$ can be expressed as:

$$x(t) = \sum_{n=1}^N \overline{Imf}_n(t) + r_N(t) \quad (7)$$

3.2 Temporal Convolutional Network

In time sequence prediction, the objective of a mapping function, F , is to learn a mapping from input series x_1, x_2, \dots, x_t to output values y_t . Formally, in univariate prediction task, there is:

$$\hat{y}_t = F(x_1, x_2, \dots, x_t) \quad (8)$$

where $x_t, y_t \in \mathbb{R}$, and \hat{y}_t is an estimate of the real target value y_t .

Similarly, in multivariate prediction task, there is:

$$\hat{y}_t = F(x_1, x_2, \dots, x_t) \quad (9)$$

where $x_t, y_t \in \mathbb{R}^n$, and \hat{y}_t is an estimate of the real target value y_t .

The mapping function F is obtained by a supervised learning. Additionally, the mapping function must conform to the causal constraint that \hat{y}_t relies only on x_1, x_2, \dots, x_t rather than any “future” inputs $x_{t+1}, x_{t+2}, \dots, x_T$.

Traditional convolutional neural network does not conform to the causal constraints. Moreover, it is not suitable for time series modeling because the receptive field of the network is very small, which is not conducive to capturing long-term dependent information. In order to solve these two problems, a special convolutional neural network, namely TCN, was proposed. A TCN model consists of three parts: causal convolution, dilated convolution and residual connection.

3.2.1 Causal Convolution

When handling with the sequential difficulty, the temporal order of data that is the causality between the data needs to be considered. Therefore, an improvement is made on the ordinary CNN, and the causal convolution is explored. Causal convolution indicates that the present output of the upper network layer only relies on the historical and present input of the lower network layer. The visualization of causal convolution is described in Fig. 2. The causal convolution also uses padding to ensure the shape of the output to be the same as that of the input. But compared with the traditional convolution network, causal convolution uses one-sided padding because of the causal constraints [38]. Although causal convolution meets

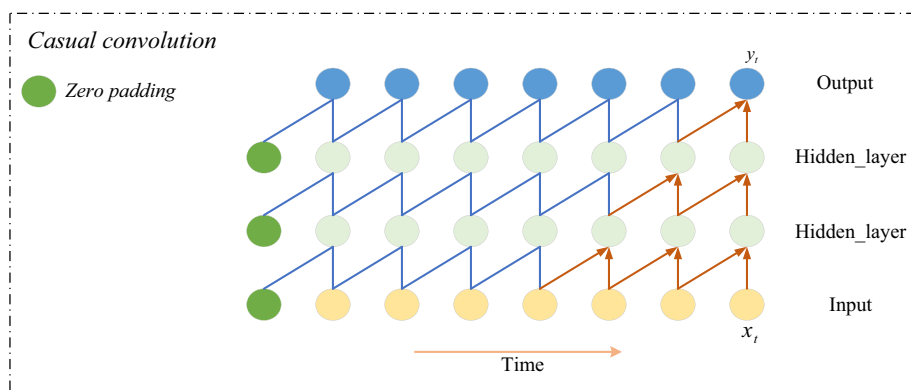


Fig. 2 Visualization of causal convolution with filter size $k = 2$

the condition of causal limitation, it has a serious problem, that is, the receptive field is greatly limited. In Fig. 2, when the convolution filter size is 2 and the number of network layers is 4, the receptive field of causal convolution is only 4. If we want to have a larger receptive field, we need to increase the convolution filter size or add more layers, which results in a huge computation load. The dilated convolution can effectively overcome the difficulty of insufficient receptive field of causal convolution.

3.2.2 Dilated Convolution

The solution to the problem of causal convolution is to utilize dilated convolution. Oord et al. [39] proposed the dilated convolution, which increases the receptive field of the network by introducing the dilation factor as a hyperparameter. More formally, for a time sequence $x \in \mathbb{R}^n$ and a filter $f : \{0, \dots, k-1\}$, the operation F on element s of the sequence is formulated as follows:

$$F(s) = (x * f)(s) = \sum_{i=0}^{k-1} f(i) \cdot x_{s-d \cdot i} \quad (10)$$

where $*$ is the convolutional operation, d is the dilation factor, k is the filter size and $s - d \cdot i$ is the direction of the past.

In the dilated convolution, the dilation means to use a specific step size between each two adjacent filters, which is controlled by the dilation factor d . The receptive field of each layer can be calculated by the following formula:

$$(k-1) \cdot \frac{b^n - 1}{b - 1} + 1 \quad (11)$$

where b is the base dilation factor, n is the number of layers, and $b = d^{\frac{1}{n}}$.

In conclusion, there are two ways in which we can enlarge the receptive field: increasing the dilation factor d and using a larger filter size k . Moreover, d increases exponentially with the increasing depth of the network, which ensures that some filters can hit each input within the effective history. The construction of dilated causal convolution is shown in Fig. 3.

3.3 Proposed Model

The proposed model consists of two parts: decomposition algorithm CEEMDAN and TCN residual block. The original time sequence can be decomposed into several IMF components and a residual by CEEMDAN. With the increase of the order of the components, the frequency of IMF decreases gradually, resulting in the component flattening. This makes the impact of sequence fluctuations be effectively suppressed, so as to further reduce the complexity of temporal data. A reduction of data complexity is conducive to the training of prediction model. On the basis of dilated causal convolution, the method of dropout is used. Moreover, the layer normalization and skip connection are added into the block which are recognized as effective methods for training deep networks. The TCN residual block applied in the proposed model is illustrated in Fig. 4.

In order to be suitable for supervised learning, the input time sequence is divided into several overlapped samples and marked with corresponding labels, as shown in Fig. 5. When the length of time series is L , the size of time window is W , the length of output is O ($O \geq 1$), the input of the t th sample is $x_t, x_{t+1}, \dots, x_{t+W-1}$ and its label is $x_{t+W}, x_{t+W+1}, \dots, x_{t+W+O-1}$.

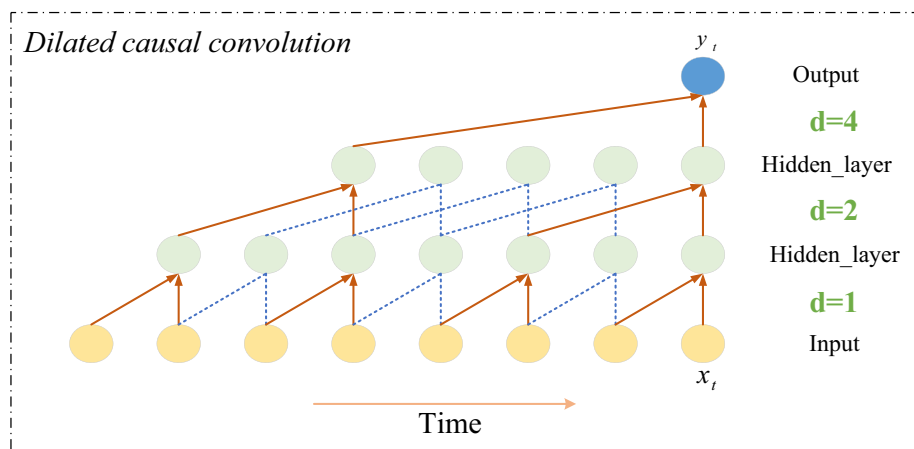


Fig. 3 Dilated causal convolution with a dilation factor of $d = 1, 2, 4$ and a filter size of $k = 2$

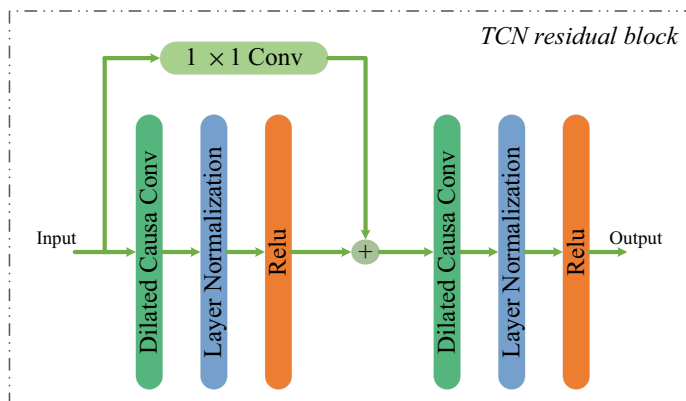


Fig. 4 TCN residual block

After the TCN block, an FC layer is added, with its number of neurons identical to the length of the output, and a linear function is used as the activation function. Mean absolute error (MAE) is chosen as the loss function and the Adam optimizer is used in back propagation. The specific process of the proposed model is shown in Fig. 6.

4 Experimental Results and Analysis

4.1 Datasets

The performance of the model was tested in two prediction tasks on three datasets. One prediction task is the univariate sequence prediction and the other is the multivariate sequence prediction. Among the three datasets, one is the ionospheric TECs over Nanchang in Jiangxi Province, P. R. China, the other is the air quality (AQ) data of Nanchang city. The third

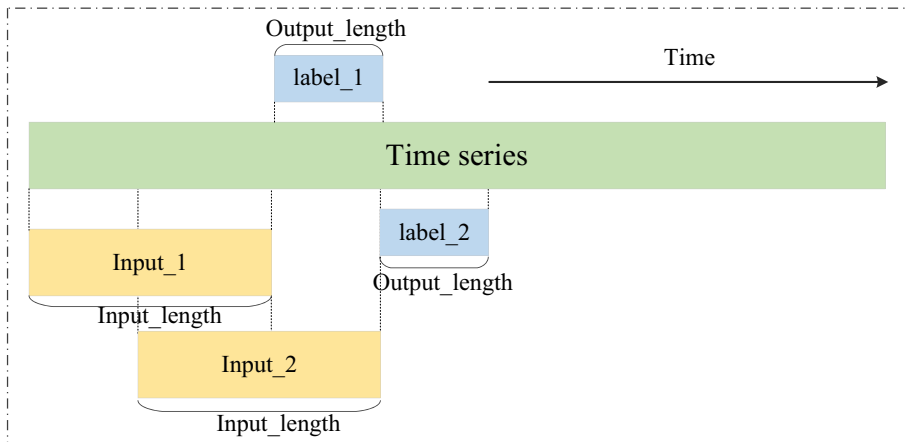


Fig. 5 Supervised learning in time series prediction

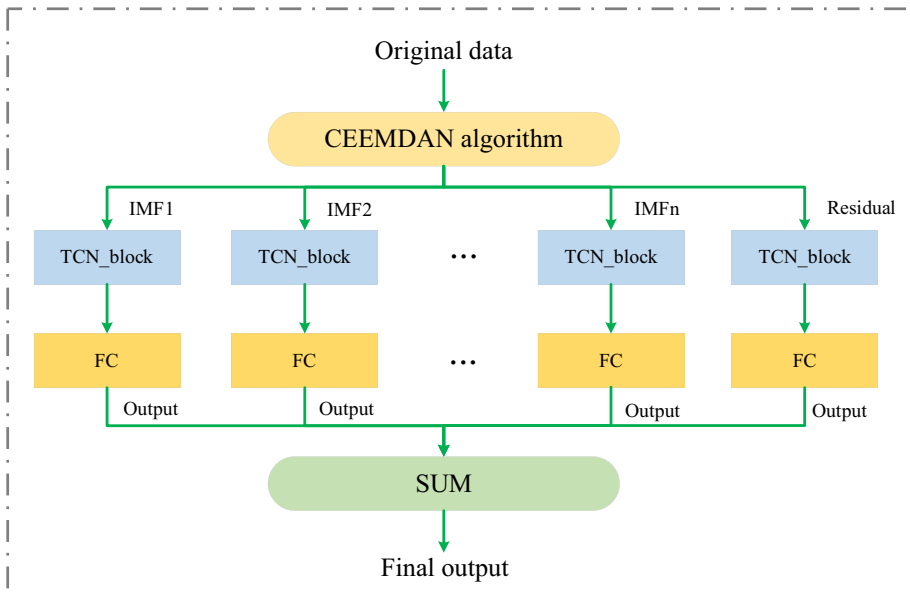


Fig. 6 Flow chat of proposed model

is of an artificial data. The type of TEC data is univariate sequence, and the type of AQ data is multivariate sequences, corresponding to our two tasks of univariate prediction and multivariate prediction, respectively, as shown in Figs. 7 and 8. The global TEC grid data was downloaded from the International GNSS Services (IGS) center and the data duration spans from January 2003 to August 2017, with a time resolution of 2 h and a spatial resolution of 5° (latitude) \times 2.5° (longitude). The air AQ data of Nanchang was downloaded from the Air Quality Historical Data Platform [40] and the time coverage is from January 2015 to March 2021, with a time resolution of 1 day and a total of 6 interdependent variables: PM_{2.5},

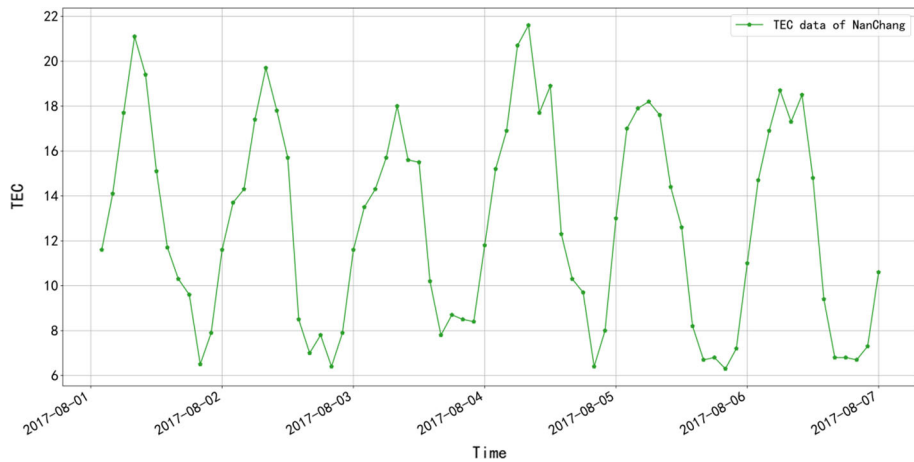


Fig. 7 Samples of TEC data of Nanchang

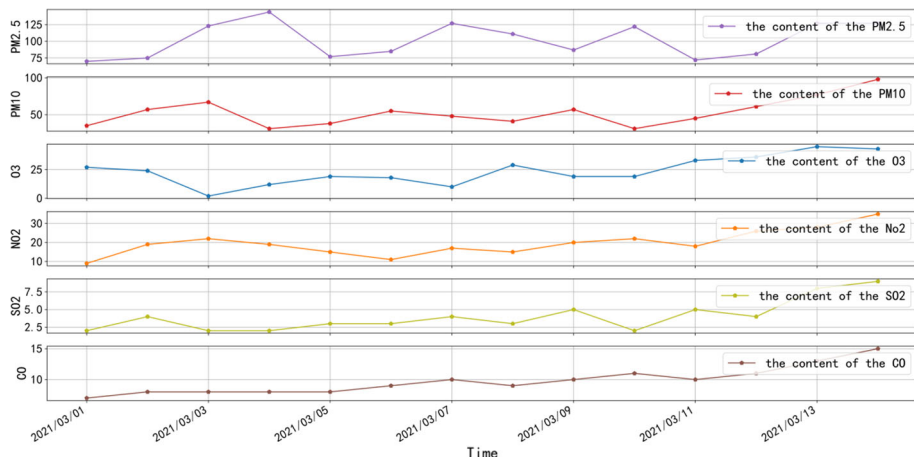


Fig. 8 Samples of air quality data of Nanchang

PM10, CO, NO₂, O₃ and SO₂. A short-term prediction of the ionospheric TEC data can, to a certain extent, improve the navigation and positioning accuracy based on electromagnetic wave propagation and the working performance of the radio communication system. Research of air quality is of great significance to local and global public health security. In addition, both the datasets are historical data, with which the future sequential data is to be predicted. Therefore, the choice of the datasets is appropriate to test the proposed CEEMDAN-TCN model.

In addition, a dataset of artificial data was chosen. The artificial data is deterministic and facilitates an exact evaluation of the model. Specifically, four sine signals, respectively at frequencies of 25, 30, 40, 50 Hz, initial phases of 0 , $\frac{\pi}{8}$, $\frac{\pi}{5}$, $\frac{\pi}{3}$ and the same sampling rate of 2000 Hz, were generated, superimposed, and mixed with a Gaussian noise sequence $N(n)$

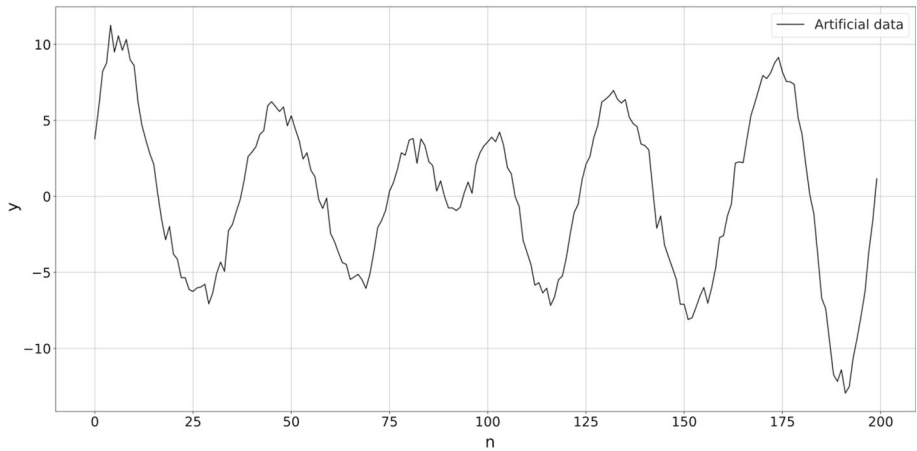


Fig. 9 Samples of the artificial data

of zero mean and unit variance:

$$y_1(t) = 6\sin(50\pi t) \quad (12)$$

$$y_2(t) = 4\sin\left(60\pi t + \frac{\pi}{8}\right) \quad (13)$$

$$y_3(t) = 2\sin\left(80\pi t + \frac{\pi}{5}\right) \quad (14)$$

$$y_4(t) = \sin\left(100\pi t + \frac{\pi}{3}\right) \quad (15)$$

resulting in an artificial sequence of $y(n) = y_1(n) + y_2(n) + y_3(n) + y_4(n) + N(n)$, as shown in Fig. 9.

4.2 Data Preprocessing

Before the original time sequence is decomposed by the CEEMDAN algorithm, a sequential preprocessing was conducted, including data sorting, data cleaning (removing the duplicate data), data normalization (alleviating the impact of different data scales) and data partitioning. In data partitioning, the dataset is divided into three parts: the training dataset (for model learning), the validation dataset (for model tuning) and the test dataset (for model evaluation), as shown in Table 1. The artificial dataset is divided into the training dataset, validation dataset and test dataset according to a ratio of 7:2:1.

Table 1 Dataset partitioning details (Date format: yyyy/mm/dd)

Dataset	Training data period	Validation data period	Test data period
TEC	2003/01/01–2015/12/31	2016/01/01–2016/12/31	2017/01/01–2017/08/11
AQ	2015/01/01–2019/12/31	2020/01/01–2020/12/31	2021/01/01–2021/03/27

Table 2 Hyperparameters for LSTM and TCN

Dataset	Model	Parameters						
		W	n	b	f	k	d	r
TEC	LSTM	36	120	36	–	–	–	0.2
	TCN	36	–	36	32	2	(1, 2, 4, 8, 16)	0.2
AQ	LSTM	3	64	36	–	–	–	0.3
	TCN	3	–	36	32	2	(1, 2)	0.3

4.3 Parameter Setting of the Model

There are three parameters in the basic LSTM model and five parameters in the basic TCN model. These parameters are the length of time window W , the size of hidden units n in LSTM, the batchsize b , the number of filters f , the filter size k , the dilation factor d and the rate of dropout r , as shown in Table 2.

4.4 Univariate Experiment

After a series of preprocessing, the CEEMDAN algorithm is applied to decompose the artificial data and TEC data. In the decomposition process, the operation of adding white Gaussian noise was performed 30 times to construct 30 noisy signals, and the decomposition results of artificial data and TEC data are presented in Figs. 10 and 11. As can be seen in Figs. 10 and 11, the frequency of the IMFs gradually decreases with the increasing order. Therefore, the prediction of an IMF and the residual can enhance the forecast precision. Then, the forecasting model is constructed for each component. Eventually, the prediction results are obtained by summing up the outputs of all components.

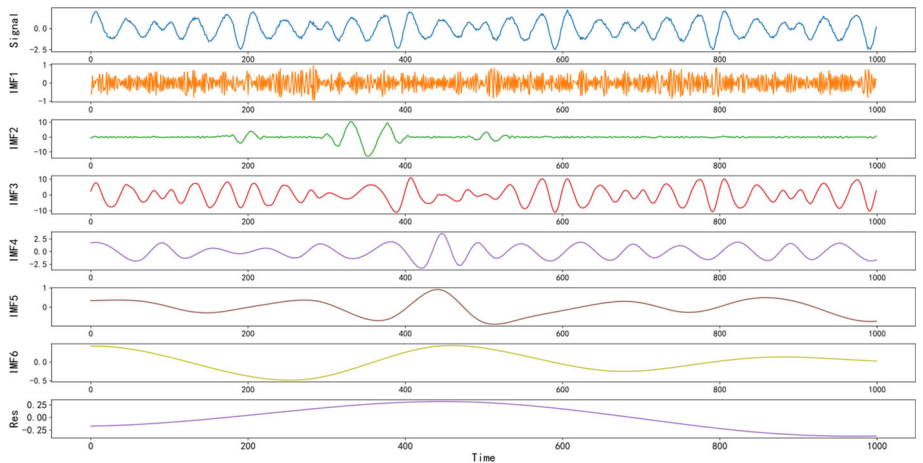


Fig. 10 Results of CEEMDAN decomposition of the artificial data

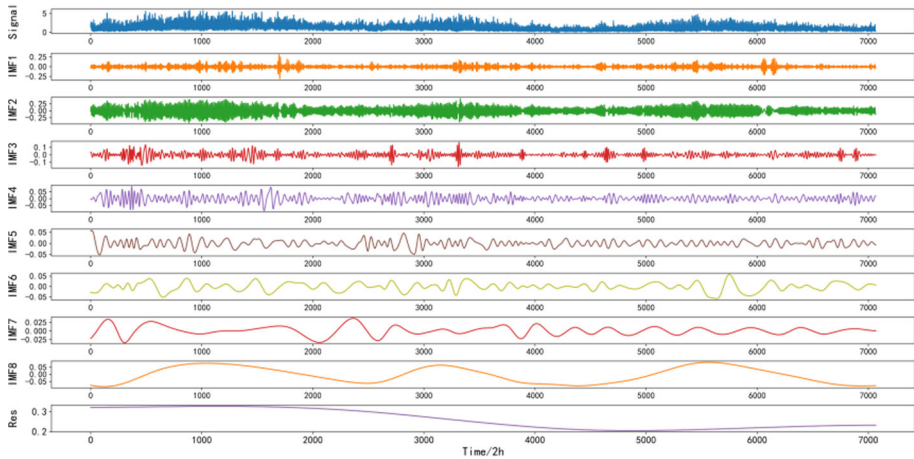


Fig. 11 Results of CEEMDAN decomposition of TEC data of Nanchang

To evaluate the performance of the proposed model numerically, four evaluation metrics are adopted, including root mean square error (RMSE), mean absolute error (MAE), mean absolute percentage error (MAPE) and determination coefficient R^2 . They are calculated as follows:

$$RMSE = \sqrt{\frac{1}{N} \sum_{t=1}^N (y_t - \hat{y}_t)^2} \quad (16)$$

$$MAE = \frac{1}{N} \sum_{t=1}^N |y_t - \hat{y}_t| \quad (17)$$

$$MAPE = 100 \times \frac{1}{N} \sum_{t=1}^N \left| \frac{y_t - \hat{y}_t}{y_t} \right| \quad (18)$$

$$R^2 = 1 - \frac{\sum_{t=1}^N (y_t - \hat{y}_t)^2}{\sum_{t=1}^N (y_t - \bar{y})^2} \quad (19)$$

where \bar{y} is the average of the sequence values of y_t . Smaller values of RMSE, MAE and MAPE indicate a smaller deviation for the predicted value from the ground-truth. The determination coefficient $R^2 \in (0, 1)$ represents the degree of model fitting to the data. An R^2 value close to 1 means prediction value very close to the ground-truth.

In the univariate experiment, 9 models are selected for comparison, namely ARIMA, EA-LSTM, TreNet, BPNN, LSTM, TCN, EMD-LSTM, EMD-TCN, CEEMDAN-LSTM and CEEMDAN-TCN, among which EA-LSTM and TreNet are the state-of-the-arts.

The evaluation metrics of different models on artificial data and TEC data are shown in Tables 3 and 4, respectively. Focusing on the RMSE, MAE and MAPE in Tables 3 and 4, it can be seen that the ARIMA and BPNN were the poor performers and all combination models (the last four) achieved better prediction performance. Furthermore, the proposed CEEMDAN-TCN obtained the best forecast results in all the ten prediction models in terms of all the evaluation metrics. As a result, it can be concluded that CEEMDAN-TCN has a more stable and accurate prediction ability.

Table 3 Evaluation metrics of different models on artificial data

Model	Evaluation metrics			
	RMSE	MAE	MAPE(%)	R ²
ARIMA	0.8033	0.6248	13.8972	0.9733
EA-LSTM	0.6830	0.5543	9.7748	0.9821
TreNet	0.7832	0.6173	13.7154	0.9762
BPNN	0.8306	0.6412	15.4487	0.9713
LSTM	0.7474	0.5859	12.8791	0.9785
TCN	0.7058	0.5712	10.9157	0.9804
EMD-LSTM	0.5097	0.4034	6.3489	0.9902
EMD-TCN	0.4615	0.3258	5.4321	0.9919
CEEMDAN-LSTM	0.4842	0.3690	5.8347	0.9910
CEEMDAN-TCN	0.4404	0.3298	5.0024	0.9924

Bold represents the proposed model

Table 4 Evaluation metrics of different models on TEC data

Model	Evaluation metrics			
	RMSE	MAE	MAPE(%)	R ²
ARIMA	3.41	2.95	20.10	0.80
EA-LSTM	2.07	1.47	10.23	0.93
TreNet	2.13	1.51	10.43	0.92
BPNN	3.04	2.23	16.72	0.83
LSTM	2.55	1.75	14.00	0.88
TCN	2.38	1.68	13.19	0.89
EMD-LSTM	2.25	1.63	12.99	0.91
EMD-TCN	1.97	1.45	11.67	0.92
CEEMDAN-LSTM	2.17	1.56	10.90	0.92
CEEMDAN-TCN	1.84	1.38	9.52	0.94

Bold represents the proposed model

In order to make a cost analysis for the proposed model, the numbers of model parameters are used for comparison, as shown in Table 5. From Table 5, it can be seen that the parameters of TCN is smaller than that of LSTM which indicates the TCN is simpler in structure than the LSTM network. The number of parameters of CEEMDAN-TCN is 9 times that of TCN in that the original data is decomposed into 9 sub-sequences in CEEMDAN, and the nine sub-sequences need to be modeled separately. Considering its significant performance gain, the proposed CEEMDAN-TCN has an acceptable complexity in network structure.

Figure 12 shows the prediction curves and the histogram of evaluation metrics on the artificial data. From Fig. 12, it can be seen that the prediction curves of all models can fit the original data well, and the superiority of the proposed CEEMDAN-TCN can be seen more apparently.

Table 5 Parameters of different models on TEC data

Model	ARIMA	EA-LSTM	TreNet	BPNN	LSTM
# parameters	–	14,941	146,028	5892	15,612

Model	TCN	EMD-LSTM	EMD-TCN	CEEMDAN-LSTM	CEEMDAN-TCN
# parameters	11,340	140,508	102,060	140,508	102,060

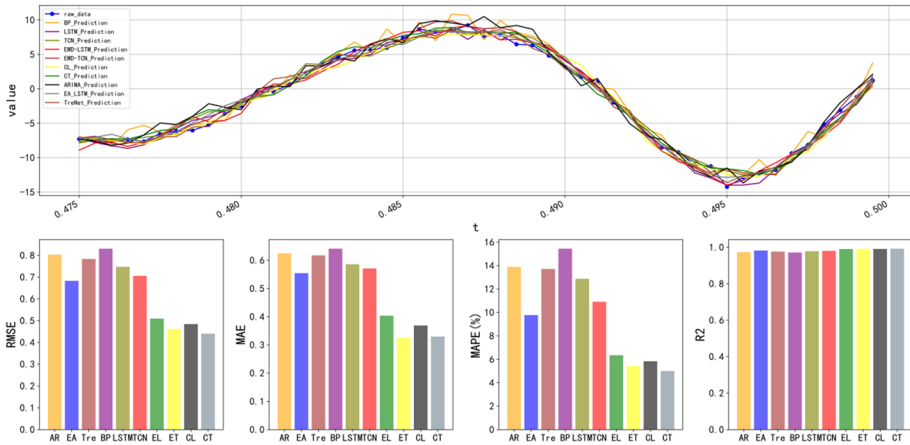
**Fig. 12** Prediction curves (upper) and bar chart (lower) of evaluation metrics on the artificial data. (AR: ARIMA; EA: EA-LSTM; Tre: TreNet; EL: EMD-LSTM; ET: EMD-TCN; CL: CEEMDAN-LSTM; CT: CEEMDAN-TCN)

Figure 13 shows the short-term prediction curves of all models on TEC data to provide a clearer assessment of the performance. Figure 14 shows the prediction curves and bar chart of evaluation metrics on TEC data in order to visually demonstrate the accuracy of the CEEMDAN-TCN. It can be seen from Figs. 13 and 14 that the prediction error of BPNN is the largest and its prediction curve deviates from the actual data. Moreover, the prediction curve of combination models, especially CEEMDAN-TCN, are closer to the actual data than that of single models. Therefore, it can be concluded that the combination models performed better than single models, and among all combination models, the proposed CEEMDAN-TCN performed the best.

4.5 Multivariate Experiment

There are 6 sequences in AQ dataset and the PM2.5 sequence is chosen as the target series for prediction. Then the CEEMDAN operation is carried out on the 6 sequences respectively, and several corresponding IMF components and residuals are obtained. In this paper, only

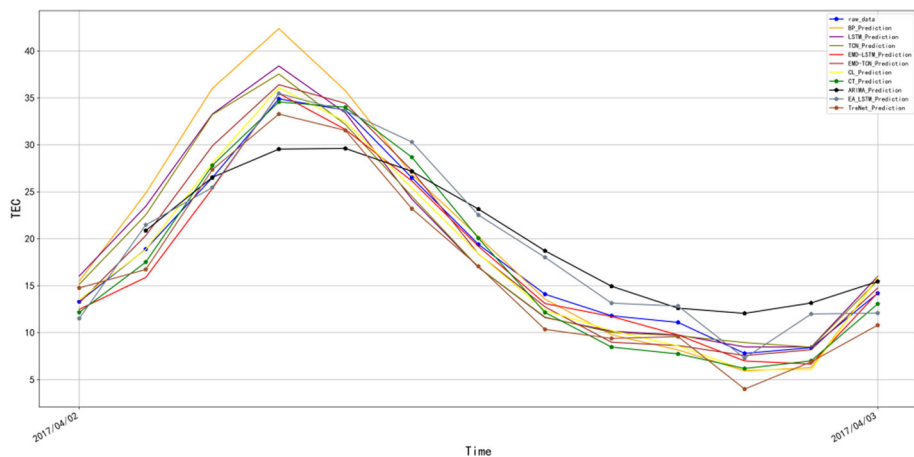


Fig. 13 Short-term prediction curves on the TEC data

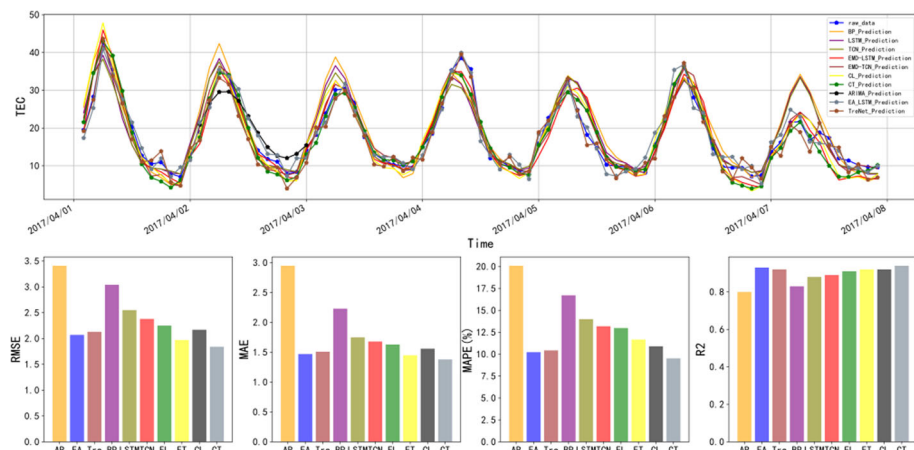


Fig. 14 Prediction curves (upper) and bar chart (lower) of evaluation metrics on the TEC data

the decomposition results of PM2.5 are shown in Fig. 15. The IMF components of the same order decomposed from different sequences are combined into six 6-dimensional inputs, which are fed respectively into the TCN model for training, and the final prediction result is obtained by summing up the training results of all inputs. Figure 16 shows the first-order IMF components of the 6 sequences of Nanchang.

In the experiment, RMSE, MAE, MAPE and R^2 are again selected as the performance evaluation metrics of the models and 7 comparison models are selected, namely LSTM, TCN, EMD-LSTM, EMD-TCN, CEEMDAN-LSTM, CEEMDAN-TCN and EA-LSTM. As can be seen in Table 6, the RMSE, MAE and MAPE of the proposed model are the smallest

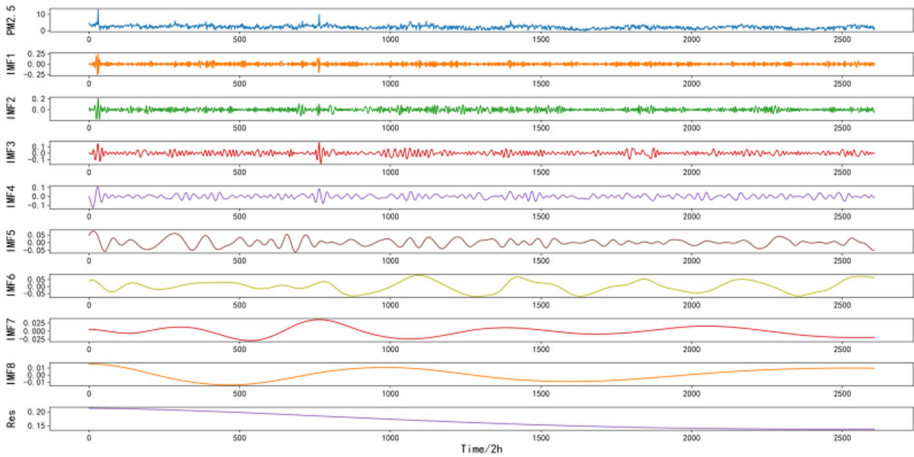


Fig. 15 Results of CEEMDAN decomposition of AQ data of Nanchang

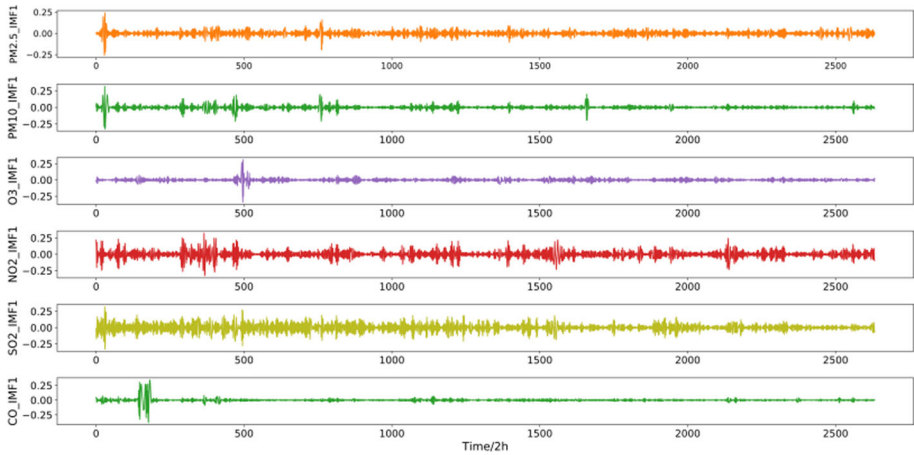


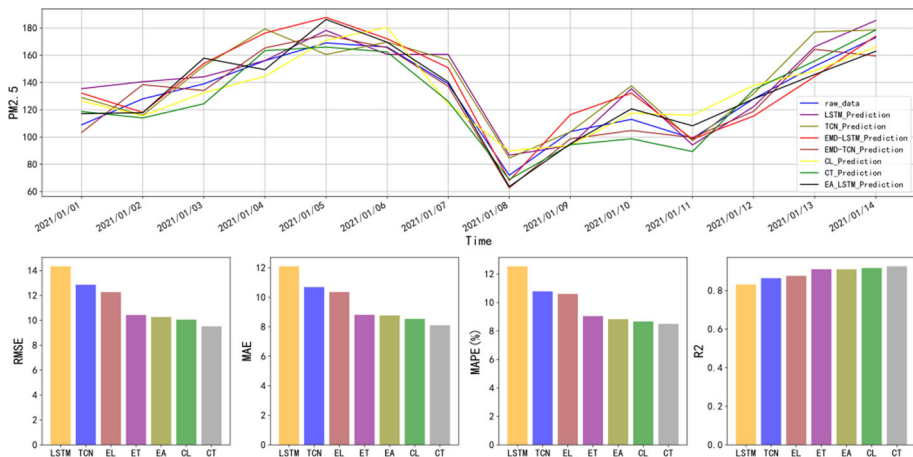
Fig. 16 First-order IMF components of 6 AQ sequences of Nanchang

among these models, and the R^2 is the closest to 1, which indicates that the proposed model performs better than all other models. In addition, the fitting performances of the prediction results of the models with the real data are drawn in Fig. 17. As can be seen more intuitively in the figures, the fitting effect of the proposed CEEMDAN-TCN model is superior to all other models.

Table 6 Evaluation indexes of different models on AQ data

Model	Evaluation metrics			
	RMSE	MAE	MAPE(%)	R ²
LSTM	14.34	12.10	12.55	0.83
TCN	12.86	10.69	10.79	0.86
EMD-LSTM	12.26	10.35	10.60	0.88
EMD-TCN	10.43	8.81	9.05	0.91
EA-LSTM	10.27	8.77	8.83	0.91
CEEMDAN-LSTM	10.06	8.54	8.67	0.92
CEEMDAN-TCN	9.52	8.10	8.50	0.93

Bold represents the proposed model

**Fig. 17** Prediction curves (upper) and bar chart (lower) of evaluation metrics on the AQ data

5 Conclusion

In this paper, a novel hybrid model is proposed which combines the CEEMDAN algorithm and TCN. The CEEMDAN algorithm, as an advanced data-adaptive time series processing technology, is applied to decompose the complicated time series into several relatively simple IMF components for facilitation of deliberate analysis and accurate prediction. In the meantime, considering the characteristics of time series, such as causal constraints, TCN model, as a powerful deep learning network, is introduced. Compared the CEEMDAN-LSTM and CEEMDAN-TCN with ARIMA, BPNN, LSTM, TCN, EMD-LSTM, EMD-TCN, EA-LSTM and TreNet, it can be concluded that the time series decomposed by CEEMDAN is more favorable to function-fitting and subsequent model convergence. By Comparing the CEEMDAN-TCN with CEEMDAN-LSTM, it can be concluded that TCN can better preserve the temporal features than LSTM. Besides, the proposed model performs well in univariate and multivariable prediction, and the parameters of the model are acceptable, which reflects its flexibility and expansibility. Finally, in prospect, the proposed model could be able to

employ in other time series prediction, such as prediction of finance and weather sequence data.

Acknowledgements This work was partly supported by the National Natural Science Foundation of China (Grant Nos. 62041106 and 61662047), and the Special Projects on Key Areas of Guangdong Provincial Department of Education (Grant no. 2021ZDZX4077).

References

1. Liu L, Shen J, Zhang M, Wang ZC, Tang J (2018) Learning the joint representation of heterogeneous temporal events for clinical endpoint prediction. In: 30th innovative applications of artificial intelligence conference, pp 109–116, 2–7 Feb 2018, New Orleans, LA. [arXiv:1803.04837](https://arxiv.org/abs/1803.04837)
2. Heydari A, Nezhad MM, Pirshayan E, Garcia DA, Keynia F, De Santoli L (2020) Short-term electricity price and load forecasting in isolated power grids based on composite neural network and gravitational search optimization algorithm. *Appl Energy* 277:115503
3. Khaniani AS, Motieyan H, Mohammadi A (2021) Rainfall forecast based on GPS PWV together with meteorological parameters using neural network models. *J Atmos Solar Terr Phys* 214:105533
4. Cao W, Hu L, Cao L (2015) Deep modeling complex couplings within financial markets. In: 29th association-for-the advancement-of-artificial-intelligence (AAAI) conference on artificial intelligence, pp 2518–2524, 25–30 Jan 2015, Austin, TX
5. Nie Y, Jiang P, Zhang HP (2020) A novel hybrid model based on combined preprocessing method and advanced optimization algorithm for power load forecasting. *Appl Soft Comput* 97:106809
6. Li W, Wang X, Zhang YW, Wu QL (2021) Traffic flow prediction over multi-sensor data correlation with graph convolution network. *Neurocomputing* 427(28):50–63
7. Haneen A, Alenezi MN, Rawajfih Y et al (2021) On the accuracy of ARIMA based prediction of COVID-19 spread. *Res Phys* 27:104509
8. Mao Q, Zhang K, Yan W et al (2018) Forecasting the incidence of tuberculosis in China using the seasonal auto-regressive integrated moving average (SARIMA) model. *J Infect Public Health* 11:707–712
9. Nieto PJG, Lasheras FS, Garcia-Gonzalo E et al (2018) PM10 concentration forecasting in the metropolitan area of Oviedo (Northern Spain) using models based on SVM, MLP, VARMA and ARIMA: a case study. *Sci Total Environ* 621:753–761
10. Liu Y, Wang RX (2016) Study on network traffic forecast model of SVR optimized by GAFA. *Chaos Solitons Fractals* 89:153–159
11. Sun WQ, Xu L, Huang X, Zhang WQ, Yuan TJ, Chen Z, Yan YH (2017) Forecasting of ionospheric vertical total electron content using LSTM networks. In: 2017 international conference on machine learning and cybernetics (ICMLC), pp 340–344, 9–12 Jul 2017, Ningbo, China. <https://doi.org/10.1109/ICMLC.2017.8108945>
12. Hu JJ, Wang XF, Zhang Y, Zhang DP, Zhang M, Xue JR (2020) Time series prediction method based on variant LSTM recurrent neural network. *Neural Process Lett* 52(2):1485–1500
13. Li YR, Zhu ZF, Kong DQ et al (2019) EA-LSTM: evolutionary attention-based LSTM for time series prediction. *Knowl Based Syst* 181:104785
14. Tao L, Tian G, Aberer K (2017) TreNet: hybrid neural networks for learning the local trend in time series. In: Proceedings of the twenty-sixth international joint conference on artificial intelligence (IJCAI-17), 19–25 Aug 2017, Melbourne, Australia
15. Ding YK, Zhu YL, Feng J, Zhang PC, Cheng ZR (2020) Interpretable spatio-temporal attention LSTM model for flood forecasting. *Neurocomputing* 403(25):348–359
16. Bai SJ, Kolter Z, Koltum V An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. [arXiv:1803.01271](https://arxiv.org/abs/1803.01271)
17. Yin LF, Xie JX (2021) Multi-temporal-spatial-scale temporal convolution network for short-term load forecasting of power systems. *Appl Energy* 283:116328
18. Zhu RJ, Liao WL, Wang YS (2020) Short-term prediction for wind power based on temporal convolutional network. *Energy Rep* 6(9):424–429
19. Lin Y, Koprinska I, Rana M (2020) Temporal convolutional neural networks for solar power forecasting. In: 2020 international joint conference on neural networks, 19–24 July 2020, Glasgow, UK. <https://doi.org/10.1109/IJCNN48605.2020.9206991>
20. Wang H, Zhao Y, Tan S (2019) Short-term load forecasting of power system based on time convolutional network. In: 2019 8th international symposium on next generation electronics (ISNE), 9–10 Oct 2019, Zhengzhou, China. <https://doi.org/10.1109/ISNE.2019.8896684>

21. Gao BX, Huang XQ, Shi JS, Tai YH, Zhang J (2020) Hourly forecasting of solar irradiance based on CEEMDAN and multi-strategy CNN-LSTM neural networks. *Renew Energy* 162:1665–1683
22. Zhou ZB, Lin L, Li SX (2018) International stock market contagion: a CEEMDAN wavelet analysis. *Econ Model* 72:333–352
23. Chen L, Chi YG, Guan YY, Fan JL (2019) A hybrid attention-based EMD-LSTM model for financial time series prediction. In: 2019 2nd international conference on artificial intelligence and big data (ICAIBD), pp 113–118, 25–28 May 2019, Chengdu, China. <https://doi.org/10.1109/ICAIBD.2019.8837038>
24. Xuan YZ, Yu Y, Wu KS (2020) Prediction of short-term stock prices based on EMD-LSTM-CSI neural network method. In: 2020 5th IEEE international conference on big data analytics (ICBDA), pp 135–139, 8–11 May 2020, Xiamen, China. <https://doi.org/10.1109/ICBDA49040.2020.9101194>
25. Cao J, Li Z, Li J (2019) Financial time series forecasting model based on CEEMDAN and LSTM. *Phys A* 519(1):127–139
26. Francis EH, Cao LJ (2017) Application of support vector machines in financial time series forecasting. *J Univ Electron Sci Technol China* 48(1–4):847–861
27. Ding M, Zhou H, Xie H, Wu M, Liu KZ, Nakanishi Y, Yokoyama R (2021) A time series model based on hybrid-kernel least-squares support vector machine for short-term wind power forecasting. *ISA Trans* 108:58–68
28. Ayeleru O, Fajimi LI, Oboirien BO, Olubambi PA (2021) Forecasting municipal solid waste quantity using artificial neural network and supported vector machine techniques: a case study of Johannesburg South Africa. *J Clean Prod* 289:125671
29. Feng X, Ma GF, Su SF, Huang CF, Boswell MK, Xue PF (2020) A multi-layer perceptron approach for accelerated wave forecasting in Lake Michigan. *Ocean Eng* 211:107526
30. Uysal G, Sensoy A, Sorman AA (2016) Improving daily streamflow in mountainous Upper Euphrates basin by multi-layer perceptron model with satellite snow products. *J Hydrol* 543:630–650
31. Huang NE, Shen Z, Long SR, Wu MLC, Shih HH, Zheng QN, Yen NC, Tung CC, Liu HH (1971) The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis. *Proc R Soc Math Phys Eng Sci* 1998(454):903–995
32. Meng EH, Huang SZ, Huang Q, Fang W, Wu LZ, Wang L (2019) A robust method for non-stationary streamflow prediction based on improved EMD-SVM model. *J Hydrol* 568:462–478
33. Li SQ, Liu XM, Lin AJ (2020) Fractional frequency hybrid model based on EEMD for financial time series forecasting. *Commun Nonlinear Sci Numer Simul* 89:105281
34. Wu YX, Wu QB, Zhu JQ (2019) Improved EEMD-based crude oil price forecasting using LSTM networks. *Phys A* 516(15):114–124
35. Torres ME, Colominas MA, Schlotthauer G, Flandrin P (2011) A complete ensemble empirical mode decomposition with adaptive noise. In: 2011 IEEE international conference on acoustics, speech and signal processing, pp 4144–4147, 22–27 May 2011, Prague, Czech Republic. <https://doi.org/10.1109/ICASSP.2011.5947265>
36. Zhang WY, Qu ZX, Zhang KQ, Mao WQ, Ma YN, Fan X (2017) A combined model based on CEEMDAN and modified flower pollination for wind speed forecasting. *Energy Convers Manag* 136(15):439–451
37. Rehman N, Mandic DP (2010) Empirical mode decomposition for trivariate signals. *IEEE Trans Signal Process* 58(3):1059–1068
38. Shen ZP, Zhang YM, Lu JW, Xu J, Xiao G (2020) A novel time series forecasting model with deep learning. *Neurocomputing* 396(5):302–313
39. Oord A, Dieleman S, Zen H, Simonyan K, Vinyals O, Graves A, Kalchbrenner N, Senior A, Kavukcuoglu K WaveNet: a generative model for raw audio. [arXiv:1609.03499](https://arxiv.org/abs/1609.03499)
40. Zhu SL, Sun JN, Liu YF et al (2020) The air quality index trend forecasting based on improved error correction model and data preprocessing for 17 port cities in China. *Chemosphere* 252:126474

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.