WEP Warwick Evans Publishing

International Journal of Computer Science and Information Technology

ISSN: 3005-9682 (Print), ISSN: 3005-7140 (Online) | Volume 3, Number 2, Year 2024
DOI: https://doi.org/10.62051/ijcsit.v3n2.26
Journal homepage: https://wepub.org/index.php/IJCSIT/index

Bi-directional LSTM-GRU Based Time Series Forecasting Approach

Bo He [a], Longbing Li [b, *], Yunya Bo [c], Jingxuan Zhou [d]

School of Computer Science and Engineering, Chongqing University of Technology, Chongqing,400054, China

[a] hebo@cqut.edu.cn, [b] llbgod@163.com, [c] 1459948258@qq.com, [d] zjx@stu.cqut.edu.cn

ABSTRACT

Time series prediction is a basic regression task in data mining, and the research of traditional methods, machine learning and deep learning has made great progress in this area. In this paper, starting from the concept of long time series, feature extraction and other related techniques and data series prediction methods, we introduce the current research status of deep learning networks in time series data and analyze the application of deep learning networks in time series prediction.

KEYWORDS

Long Time Series; Neural Networks; BILSTM; Deep Learning

1. INTRODUCTION

In recent years, with the popularization of the Internet and the development of science and technology, various types of time series data are being collected in large quantities in various industries, such as financial aspects, weather forecasting, and power distribution. Through time series forecasting, future trends can be predicted to provide reference and support for decision making [1]. Time series forecasting is the use of the characteristics of the time of an event in the past period to predict the characteristics of the event in the future period. This is a relatively complex class of predictive modeling problems, and regression analysis model prediction is different, the time series model is dependent on the sequence of events, the same size of the value to change the order of the input model produces different results [2-4]. First of all, it needs to be clear that time series can be categorized into smooth series (i.e., there is some kind of cycle, seasonality and trend of the variance and mean do not change over time), non-smooth series.

Therefore, how to construct accurate and interpretable time series prediction models based on deep learning methods has become a current research hotspot. However, since time series forecasting models are commonly used in decision making, if the model cannot explain the forecasting results, decision makers may have doubts about the forecasting results, which affects the decision making. Therefore, when constructing a time series forecasting model, the accuracy and interpretability of the model need to be considered. In deep learning methods, neural networks and their variants are commonly used models. Due to the complexity of neural networks, their expressive power is also outstanding, which can capture the complex nonlinear relationship between feature variables, and they are also widely used in the field of time series forecasting. Therefore, how to construct accurate and interpretable time series forecasting models based on deep learning methods has both theoretical research value and important practical significance.

2. TIME SERIES FEATURE EXTRACTION METHOD

In terms of time series feature extraction, many effective feature extraction methods have been proposed at home and abroad, which are statistic-based feature extraction method and transform-based feature extraction method.

Statistic-based feature extraction is a feature extraction method that is directly extracted or simply calculated according to the statistical features of the time series data. For statistical features of time series are often divided into two categories: time domain and frequency domain. Features in the time domain can be divided into quantitative and dimensionless features, quantitative features such as mean, variance, root mean square, peak, etc., and dimensionless features such as impulse factor, peak factor, waveform factor, etc.; while features in the frequency domain include the mean-square frequency, the root-mean-square frequency, the frequency variance, and the standard deviation of the frequency.

For example, Deng et al [5] proposed a tree integration method for time series classification called time series forest (TSF), which uses features such as mean, standard deviation, and slope to make the TSF algorithm computationally efficient, with algorithmic complexity linear in the length of the time series, and the algorithm outperforms competitors such as single nearest neighbor classifiers with dynamic time regularization. The method proposed by Nanopoulos et al [6] for extracting statistical features such as mean and deviation of the entire time series, and then using a multilayer perceptron neural network for classification. This method only captures the global attributes of the time series, which may leave the local attributes that help in classification ignored. Zheng Enming et al [7] proposed a time delay difference estimation method based on frequency variance weighting, which utilizes the frequency variance as a feature to estimate the time delay difference of the target radiated signals received by different receiver array elements in a hydroacoustic positioning system.

Statistical-based methods are commonly used in data analysis, this class of methods is simple and the acquired features are interpretable, but usually also have to be used in conjunction with other methods to achieve the best algorithmic performance, and statistical-based features are usually intuitive, making it difficult to acquire features with a high degree of abstraction.

Transformation-based methods for feature extraction are methods that map and transform time series data in different domains, making the features come to the fore in a certain dimension. Common domain transforms are transforms on the time and frequency domains, typically including Fourier transform [8] and wavelet transform [9]. Such transforms have been applied in different forms for different application scenarios, such as discrete Fourier transform, fast Fourier transform, short-time Fourier transform, and discrete wavelet transform. There are also some methods based on linear transforms, such as principal component analysis.

There are many methods that are based on the Fourier transform, for example, Samiee [10] et al. proposed a technique for epileptic EEG feature extraction using the discrete short-time Fourier transform (DSTFT). The method utilizes rational functions for adaptive, localized time-frequency representation of the EEG signal in order to separate seizure periods from seizure-free periods. Shang Rongyan [11] et al. proposed a feature extraction method based on Fast Fourier Transform, which is mainly aimed at the feature extraction of vibration signals of electromechanical equipment and is applied in the field of fault diagnosis of electromechanical equipment. The preprocessed vibration signal data is subjected to fast Fourier transform (FFT) to obtain the vibration signal spectrum information. Compared with the wavelet packet decomposition (WPD) and empirical modal decomposition (EMD) and other fault feature extraction methods, this method has simple principles and formulas, small computational volume, fast computing speed, high fault recognition rate, and is easy to be implemented in embedded systems and DSP programs.

Wavelet transform based feature extraction has attracted great interest due to its high accuracy and good interpretability.Seena et al [12] found that the wavelet transform based method performs the

best in measuring irregularities and is suitable for ECG data, which led to a comparison of different feature extraction and denoising techniques for wavelet transform. The paper compares different wavelet transform techniques for feature extraction and denoising of ECG signals, suitable for selecting the most applicable wavelet transform technique and proves that wavelet transform is one of the powerful tools for ECG signal analysis. Wavelet transform-based methods extract or learn shapes from training time series, and although they can achieve higher accuracy, they still face some challenges. Wavelet transform-based feature extraction methods are less accurate when the training dataset is small. In addition, the wavelet transform requires a priori knowledge to pre-set some parameters.

In addition to the transformation methods mentioned above, there are also methods based on principal component analysis (PCA), for example, Yang et al [13] proposed a similarity measure for multivariate time series datasets based on principal component analysis, where PCA is applied to a multivariate dataset represented by a matrix to generate the principal components and associated eigenvalues. These principal components and eigenvalues are then utilized to compare the similarity between multivariate matrices. Experiments are conducted on three datasets (2 real and 1 synthetic). The results show the superiority of the method in terms of precision and recall compared to traditional similarity measures such as Euclidean distance and dynamic time regularization.

Since time series document the historical process of variables over time, analyzing time series can reveal their development patterns and analyze and predict future states [14]. Generally speaking, the mathematical features of time series mainly include random terms, periodic terms, and trend terms, and the information of a time series is mainly contained in these features [15], therefore, in order to facilitate the analysis, suitable methods for feature extraction are needed. For example, LIU Yi et al. used ARMA model for feature extraction and damage warning of time series and obtained ideal results [16]; ZHANG Zhigang et al. used algorithms such as sliding peak state and other algorithms to effectively carry out feature extraction [17]; WANG Gang et al. applied genetic algorithms to analyze and achieved good results [18]. Therefore, through appropriate means and methods, feature extraction of time series and analysis and forecasting of monitoring targets can be better understood and grasped the current status of monitoring targets and future development trends.

3. TIME SERIES PREDICTION METHODS

In terms of time series forecasting methods, many effective feature extraction methods have been proposed at home and abroad, which are traditional time series forecasting methods, time series forecasting methods based on machine learning, and time series forecasting methods based on deep learning.

The traditional time series forecasting methods mainly focus on specific data, design mathematical (morphology function) models, and capture the temporal feature laws to complete the forecasting work [19]. Classical time series models include Moving Average (MA), Autoregressive (AR), Autoregressive Moving Average (ARMA), Integrated Moving Average Autoregressive (Autoregressive Integrated Moving Average model (ARIMA).Instead of using the past values of the predictor variables in a regression, the MA moving average model uses the past prediction errors in a regression-like model; the AR model is a statistical way of dealing with a time series, using the previous values of the same variable, e.g., X, for all previous time periods, i.e., X 1 values of the same variable, e.g., X 1 through X t-1, to predict the performance of X t, assuming that they are in a linear relationship. Because this is a development from linear regression in regression analysis, instead of using X to predict Y, X is used to predict X (itself); hence the name autoregressive; the ARMA model consists of a "fusion" of an autoregressive model (AR) with a moving average model (MA) as a base. It is modeled only for smooth data [19]. The data series formed by the predictors over time is regarded as a random sequence, and the dependence of this group of random variables reflects the continuity of the original data in time; ARIMA model for non-stationary time series, after eliminating its local

level or trend, it shows a certain degree of homogeneity, at this time, some parts of the series are very similar to the other parts of the series, and this kind of non-stationary time series can be converted into a stationary time series after difference processing. This kind of non-stationary time series can be converted to a smooth time series after difference processing, and this kind of time series is called chi-square non-stationary time series, in which the number of difference is the order of chi-square. The traditional time series prediction methods are highly targeted, robust and interpretable, but with low learning freedom and poor generalization.

In recent years, machine learning has been used in time series forecasting research. The machine learning method is mainly to construct sample datasets, using the "time feature" to "sample value" approach, through supervised learning, to learn the correlation relationship between features and labels, so as to realize time series prediction. LightGBM and XGBoost, as a representative, generally is to convert the time series problem into supervised learning, through feature engineering and machine learning methods to predict; this model supports complex data modeling, support for nonlinear problems; and these methods have a high degree of freedom in learning, and can effectively introduce covariate factors. This model with high accuracy and good robustness can solve most of the complex time-series prediction models, however, this method requires a more complex part of the artificial feature process, and feature engineering requires a certain degree of expertise or rich imagination. This is because the level of feature engineering capability often determines the upper limit of machine learning, and machine learning methods simply approximate this upper limit as closely as possible. After the features are established, the tree model algorithm LightGBM or XGBoost can be applied directly, these two models are very common fast modeling methods, in addition, they have fast computation speed, high model accuracy; missing values do not need to be processed, which is more convenient; support Category variables; support for feature crossover and so on, but the feature engineering is time-consuming, laborious, and relies on the experience of experts, plus the complexity is high. However, feature engineering is time-consuming and depends on experts' experience, plus high complexity.

With the rapid development of deep learning theory centered on neural networks and the breakthrough of many scientific research results, using deep learning to solve time series prediction problems has become a current trend. These methods are mainly LSTM [20-22], GRU, seq2seq, wavenet, 1D-CNN, Transformer. The LSTM/GRU model in deep learning is specifically designed to solve time series problems [23, 24], and although the CNN model was originally designed to solve image problems, it has evolved and developed to solve time series problems as well. The neural networks commonly used in time series prediction tasks are Recurrent Neural Networks (RNN), Long Short-Term Memory Networks (LSTM) and Gated Recurrent Unit Neural Networks (GRU);

LSTM is an improved RNN that solves the long-term dependency problem in RNNs by introducing three "gates", namely, the forgetting gate, the input gate, and the output gate, to control the transfer of information. The function of the forgetting gate is to decide what information should be discarded or retained, the input gate is used to update the cell state, and the output gate is used to determine the value of the next hidden state. state value, the hidden state contains the information from the previous inputs [25-27]; LSTM can choose to accumulate those information that is needed and forget those that are not needed, and thus is more suitable for dealing with time series prediction problems;

The biggest difference between LSTM and previous RNN is the addition of some structures called "gates", which are actually small structures that control the range of information through sigmoid functions, in RNN there is only one tanh function to process information, while LSTM has three gate structures responsible for forgetting respectively, input and output. The structure is shown in Fig. 1.

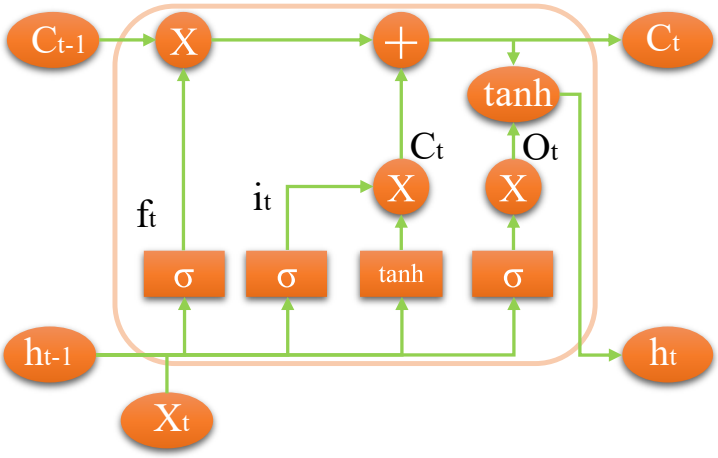

Figure 1. Structure of LSTM

The bi-directional LSTM layer contains two layers, forward and backward LSTM, each with the same structure. Each neuron contains four parts: an output gate O_t, an input gate i_t, a forgetting gate f_t, and a long and short memory state C_t. The input gate of a neuron at moment t contains three parameters: the current input X_t, the hidden state of the neuron at the previous moment h_{t-1}, and the state of the neuron at the previous moment C_{t-1}. The output gate of a neuron is two-dimensional and contains the weight h and the categorized category C. W is its corresponding word embedding matrix, b is the bias term, σ is the softmax function, and *tanh* is the activation function. The feature sample data are input into the bidirectional LSTM for training at one time to complete one forward and backward propagation and parameter update as follows.

1) Forgetting gate layer. The forgetting gate decides what information to discard from the current cell state, it controls the degree of forgetting of the previous neuron's input memory C_{t-1}, which is computed using Eq. (1), and outputs a number between 0 and 1 corresponding to the current cell state, where 0 means completely discarded, and 1 means completely retained.

$$f_t = \sigma(W_f \bullet [h_{t-1}, x_t] + b_f) \tag{1}$$

2) Input gate layer. The input gate decides what kind of new information will be stored in the cell state, i.e., it controls the updated value as shown in Eq. (2), and then after a tanh process, the update result is added to the new state to obtain the new candidate value as shown in Eq. (3):

$$i_t = \sigma(W_i \bullet [h_{t-1}, x_t] + b_i) \tag{2}$$

$$\tilde{C}_t = \tanh(W_c \bullet [h_{t-1}, x_t] + b_c) \tag{3}$$

3) Memory state layer. The memory state layer updates the state of the cell according to the results of the first two layers by multiplying the old state with the forgotten value, discarding the information that needs to be forgotten, and adding the new candidate value, so the memory state of the neuron at moment t is shown in Equation (4):

$$c_t = f_t * c_{t-1} + i_t * \tilde{C}_t \tag{4}$$

4) Output gate layer. The output gate determines the output value according to the cell state, and first determines which part of the cell state will be output according to the sigmoid, as shown in Eq. (5), and then goes through the tanh process to get the final output $\boldsymbol{h}_t$ as shown in Eq. (6):

$$o_t = \sigma(W_o \bullet [\mathrm{h}_{t-1}, x_t] + b_o) \tag{5}$$

$$h_t = o_t * \tanh(c_t) \tag{6}$$

Above is the computation process of forward LSTM layer. In the reverse LSTM layer to get $\overleftarrow{h}_t, \overrightarrow{\mathrm{h}}_t$, the final output of the implicit layer of the data vector weights from the combination of forward and reverse outputs, for updating to get, the update equation is equation (7):

$$h_t = \overleftarrow{h}_t \oplus \overrightarrow{\mathrm{h}}_t \tag{7}$$

The Bi-LSTM algorithm is based on the LSTM algorithm, which records the backward information at the same time when the LSTM algorithm only records the forward information, and combines the forward and backward information for feature extraction. For the output at time ***t***, the forward LSTM layer has information at time ***t*** and before time ***t*** in the input sequence, and the backward LSTM layer has information at time t and after in the input sequence. The output of forward LSTM layer at time ***t*** and the output of backward LSTM layer at time t are obtained by superposition, and the vector outputs of the two LSTM layers are processed by addition, averaging, or concatenation, so that the output information reflects the forward and backward connections of the time signals better.Although the LSTM is able to capture the dependency relationship of the longer distance and the connection between the features in time, it is difficult for it to The Bi-LSTM neural network structure model is divided into two independent LSTMs, the input sequence is fed into the two LSTM neural networks in forward and reverse order for feature extraction, and the feature vector formed by splicing the two output vectors (i.e., the extracted feature vectors) is used as the final feature expression. The concept of Bi-LSTM is to make the feature data obtained at time t have information between past and future at the same time, and the experiments prove that the efficiency and performance of this neural network structural model for feature extraction are better than that of a single LSTM structural model. It is worth mentioning that the two LSTM neural network parameters in Bi-LSTM are independent of each other, and they only share the input vector list.

While GRU, like LSTM, is proposed to solve the problems of long-term memory and gradient in backpropagation, GRU uses two gating units, reset gate and update gate, to control the transfer of information, the reset gate determines how to combine the new input information with the previous memory, and the update gate defines the amount of the previous memory saved up to the current timestep [28].

The design of GRU is the same as the purpose of LSTM, and it can also be said to be a variant of LSTM. GRU redesigns some gate structures of LSTM, which is summarized into two gate structures, one is reset gate and the other is update gate. The structure is shown in Figure 2.

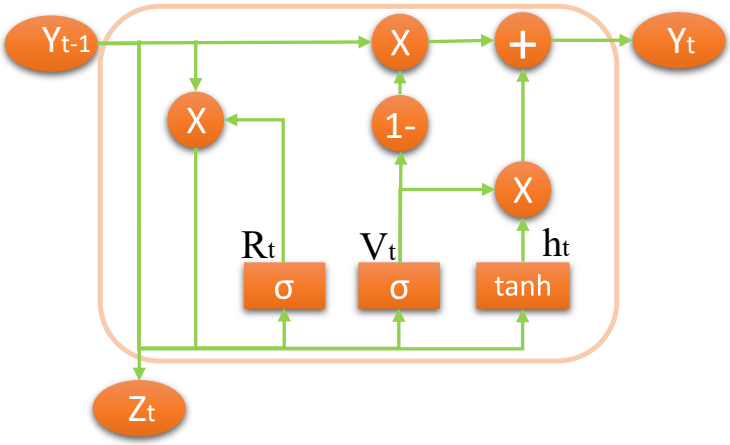

Figure 2. Structure of GRU

GRU has two gates, the first one is the update gate V_t, the update gate V_t determines how much historical information can continue to be passed on to the future, the update gate V t is calculated [1] as shown in equation (8).

$$V_t = \sigma(W_V \bullet [Y_{t-1}, Z_t] + b_V) \tag{8}$$

where: W_v is the weight matrix of the update gate, b_v is the deviation vector, and σ denotes the activation function sigmoid. the second gate is the reset gate R_t, and the main function of the reset gate R_t is to determine how much of the history information cannot be passed to the next state, the calculation method of the reset gate R_t [1] is shown in Equation (9).

$$R_t = \sigma(W_R \bullet [Y_{t-1}, Z_t] + b_R) \tag{9}$$

Where: W_r is the weight matrix of the reset gate and b_r is the deviation vector. After calculating the update gate V_t and the reset gate R_t, the GRU will calculate the candidate hidden state h_t, and the candidate hidden state h_t is calculated [1] as shown in Equation (4).

$$h_t = \tanh(W_h \bullet [R_t \bullet Y_{t-1}, Z_t] + b_h) \tag{10}$$

where W_h is the corresponding weight parameter, b_h is the corresponding deviation parameter, and *tanh* represents the hyperbolic tangent function. The output Z_t of the GRU at the final time t is calculated [1] as shown in Equation (5).

$$Y_t = (1 - V_t) \bullet Y_{t-1} + V_t \bullet h_t \tag{11}$$

Traditional GRU is unidirectional, data can only be processed in one direction, important information is easily lost, and the network can only combine the input of the current moment and the hidden state information of the previous moments to compute the new hidden layer state information h_t. Bi-LSTM was proposed by Schuster, Paliwal, et al. in 1997, Bi-LSTM means bidirectional, which means using both forward and reverse information of the time series data. which means using both forward and backward information of the temporal data.The advantage of Bi-GRU is that it can capture information over a longer period of time, and the use of bi-directional GRU has more expressive power compared to uni-directional GRU, which has fewer parameters compared to LSTM, and thus can converge faster.

For example, Huang, Li, Zhang and Ren used empirical mode decomposition (EMD) to decompose the PM2. 5 concentration sequence [29], after this decomposition, the obtained smooth eigenmode sequences and meteorological features were successively inputted into the constructed GRU neural network for training and prediction, and finally the prediction results of the individual sub-sequences were summed up to get the predicted value of PM2. 5 concentration. Wang and Tang used the complete envelopment EMD (CEEMD) method to decompose the original sequences [30] to obtain different subsequences at different frequencies, subsequently, they used the fuzzy entropy algorithm to reconstruct the continuity of these decompositions, and finally, they utilized the Long Short-Term Memory (LSTM) network and Extreme Learning Machines (ELMs), which were optimized by the Whale algorithm, in order to predict the different frequencies of the continuity. Specifically, LSTM is used to predict high-frequency sequences, while ELM is used to predict low-frequency and trending sequences. Similarly, Zhang, Zeng, and Yan employed VMD to decompose the original PM2.5 sequence into multiple sub-signal components based on the frequency domain [31], and they further utilized a bidirectional LSTM (BiLSTM) model to predict these sub-signal components.Ma et al. proposed a stacked BiLSTM based on transfer learning [32], which initially uses data from existing monitoring stations to data to pre-train the base BiLSTM model. Subsequently, the initial layer of the base model was frozen and the remaining layers were fine-tuned using data from new monitoring stations. Finally, the Transfer Learning Stacked BiLSTM (TLS-BLSTM) was trained and evaluated using data from the new monitoring stations.Tariq et al. proposed a transfer learning-based ResNet framework for sequential health risk prediction of PM2.5 levels in subway stations [33], which utilizes subway stations with abundant monitoring data to develop reliable and acceptable prediction models that are subsequently transferred to another subway environment. Specifically, the ResNet architecture is initially pretrained by direct training. Following this, the pre-trained model was fine-tuned using limited data from the new station.Ma Cheng, Lin Tan and Zhang introduced a methodological framework [34] that combines a BiLSTM network with a transfer learning strategy to improve the prediction accuracy of air pollutant concentrations at various temporal resolutions.Unlike the traditional transfer strategy, this study transfers the acquired knowledge from the finer temporal resolution to coarser temporal resolution. Experimental results show that transfer learning helps to improve the prediction accuracy of BiLSTM models at higher temporal resolutions.Fong Li, Fong Wong used an LSTM recurrent neural network (RNN) model to predict the air pollution levels in Macau, China [35].In addition, they utilized a transfer learning approach to transfer the knowledge from monitoring stations with richer observational data to other data-limited monitoring stations, aiming to improve the accuracy of prediction.

4. CONCLUSION

Although time series forecasting methods have experienced a long period of development, the rapid growth of data size has brought serious challenges to traditional forecasting methods and seriously affected the efficiency of forecasting methods. Traditional time series forecasting methods can play a greater advantage when dealing with univariate forecasting problems; however, if there are too many problems or variables, then the traditional time series model is out of reach, and the traditional time forecasting methods have low degrees of freedom and poor generalization. Time series prediction methods based on machine learning, this type of method has a higher degree of freedom and better generalization, which can make up for the shortcomings of traditional time series prediction, but the difficulty lies in feature engineering. The use of deep learning methods for time series data prediction has become a new trend, which also points out the direction for our further research. However, among the many deep learning-based time series prediction methods, the prediction accuracy of a single time series prediction model is limited and the degree of freedom is slightly lower than that of the fusion model, so we should comprehensively consider the degree of information extraction from the data and the time spent on running the model. We should consider both the information extraction degree of data and the time spent on model operation, and solve the problem

that time series prediction methods do not rely on the extraction of long time series data, resulting in low prediction accuracy through further research.

ACKNOWLEDGMENTS

This research is funded by the Chongqing University of Technology 2023 Graduate Education High Quality Development Project (No. gzlcx20233255)

REFERENCES

[1] Liu M, Wei L. EMD-LSTM algorithm and its prediction in PM2.5 [J]. Journal of Changchun University of Technology, 2020, 41(4):322-327.

[2] LI Jianping, WANG Xingwei, MA Lianbo, et al. Research on interval-based time series classification algorithm [J]. Cyberspace Security, 2019, 10(8):10.

[3] Zhang Ke, Cui Le. Research on multivariate time series classification algorithm based on PCA-LSTM model [J]. Statistics and Decision Making, 2020(15):6

[4] Wang Wei, Wang Wenfa, Zhang Zhe. PM2.5 concentration prediction based on gated recurrent unit neural network [J]. Wireless Interconnection Technology, 2019, 16(4): 29-32.

[5] Houtao Deng, George Runger, Eugene Tuv, Martyanov Vladimir, A time series forest for classification and feature extraction [J], Information Sciences,Volume 239, 2013, Pages 142-153,https://doi.org/10.1016/j.ins.2013.02.030.

[6] Nanopoulos A, Alcock R, Manolopoulos Y. Feature-based classification of time-series data [J]. International Journal of Computer Research, 2001, 10(3): 49-61.

[7] Zheng Enming, Chen Xinhua, Sun Changyu. Delay difference estimation method based on frequency variance weighting [J]. Systems Engineering and Electronics, 2014, 36(2): 224-229.

[8] Bracewell R N, Bracewell R N. The Fourier transform and its applications [M]. New York: McGraw-Hill, 1986.

[9] Zhang D. Wavelet transform[M]//Fundamentals of Image Data Mining. Springer, Cham, 2019: 35-44.

[10] Samiee K, Kovacs P, Gabbouj M. Epileptic seizure classification of EEG time-series using rational discrete short-time Fourier transform [J]. IEEE transactions on BIomedical Engineering, 2014, 62(2): 541-552.

[11] SHANG Rongyan, PENG Changqing, FANG Ruiming,et al. A feature extraction method for vibration signal of electromechanical equipment [P]. Fujian:CN112268615A, 2021-01-26.

[12] Seena V, Yomas J. A review on feature extraction and denoising of ECG signalusing wavelet transform[C]//2014 2nd international conference on devices, circuits and systems (ICDCS). IEEE, 2014: 1-6.

[13] Yang K, ShahaBI C. A PCA-based similarity measure for multivariate timeseries[C]//Proceedings of the 2nd ACM international workshop on Multimedia databases. 2004: 65-74.

[14] WANG Hong, SU Shanmai, LIU Dongqin. A preliminary study of time series analysis and its application in the field of surveying and mapping [J]. Surveying and Mapping Science, 2008, 33(1):155-158

[15] MA She-Xiang, LIU Guizhong, ZENG Zhaohua. Analysis and prediction of non-stationary time series based on wavelet analysis [J]. Journal of Systems Engineering, 2000(04):305-311.

[16] LIU YI, LI AI-QUN, FEI QING-GUO, et al. Featureextraction and damage alarming using time series analysis [J]. Journal of Southeast University (English Edition), 2007, 23(1):86-91.

[17] ZHANG Zhigang, SHI Xiaohui, CHEN Zheming, et al. Rolling bearing fault feature extraction based on improved EMD and sliding peak state algorithm [J]. Vibration and Shock, 2012, 31(22):80-83.

[18] WANG Gang, NI Shihong, SHA Mengchun. Non-synchronized time series feature extraction method based on genetic algorithm [J]. Computer Engineering, 2005, 31 (17):155-156.

[19] GUO S, QIAO W, CHEN B, et al. Prediction and abnormality analysis of climate change based on PCA-ARMA and PCC[C]//2020 IEEE International Conference on Networking, Sensing and Control (ICNSC). IEEE, 2020:1-6.

[20] HOCHREITER S, SCHMIDHUBER J. Long short-term memory [J]. Neural computation, 1997, 9(8):1735-1780.

[21] NI R, CAO H. Sentiment Analysis based on GloVe and LSTM-GRU[C]// 2020 39th Chinese Control Conference (CCC). Shenyang: IEEE, 2020:7492-7497.

[22] CHEN Y C, LI D C.Selection of key features for PM 2.5 prediction using a wavelet model and RBF-LSTM [J]. Applied Intelligence, 2021(51):2534-2555.

[23] SONG Feiyang, TIE Zhixin, HUANG Zehua, et al. PM2.5, Concentration Prediction Model Based on KNN-LSTM [J]. Computer Systems & Applications, 2020, 29(7):193-198.

[24] Liu M, Ren S, Ma S, et al. Gated Transformer Networks for Multivariate Time Series Classification [J]. arXiv preprint arXiv:2103.14438, 2021.

[25] Li Xiangwei, Liu Siyan, Gao Kunlun. Transient Stability Assessment of Power System Based on Bidirectional Long and Short Memory Network and Convolutional Neural Network [J]. Science technology and engineering, 2020, 20(7): 2733-2739.

[26] Bai Shengnan, Shen Xiaoliu. PM2.5 prediction based on LSTM recurrent neural network [J]. Computer Application and Software, 2019, 36(1):67-70, 104.

[27] RANI N, DAS P, BHARDWAJ A K. A hybrid deeplearning model based on CNN-Bi-LSTM for rumor detection [C]. Proceedings of the 6th International Conference onCommunication and Electronics Systems. Piscataway: IEEE Press, 2021:1423-1427.

[28] Munir HS, Ren SAO, Mustafa M, et al. Attention based GRU-LSTM for software defect prediction [J]. PLoS One, 2021, 16(3):e0247444.

[29] Huang, G., Li, X., Zhang, B., & Ren, J. (2021). PM2. 5 concentration forecasting at surface monitoring sites using GRU neural network based on empirical mode decomposition. Science of the Total Environment, 768, Article 144516.

[30] Wang, W., & Tang, Q. (2023). Combined model of air quality index forecasting based on the combination of complementary empirical mode decomposition and sequence reconstruction. Environmental Pollution, 316, Article 120628.

[31] Zhang, Z., Zeng, Y., & Yan, K. (2021). A hybrid deep learning technology for PM 2.5 air quality forecasting. Environmental Science and Pollution Research, 28, 39409–39422.

[32] Ma, J., Li, Z., Cheng, J. C., Ding, Y., Lin, C., & Xu, Z. (2020). Air quality prediction at new stations using patially transferred bi-directional long short-term memory network. Science of the Total Environment, 705, Article 135771.

[33] Tariq, S., Loy-Benitez, J., Nam, K., Lee, G., Kim, M., Park, D., et al. (2021). Transfer learning driven equential forecasting and ventilation control of PM2. 5 associated health risk levels in underground public facilities. Journal of Hazardous Materials, 406, Article 124753.

[34] Ma, J., Cheng, J. C., Lin, C., Tan, Y., & Zhang, J. (2019). Improving air quality prediction accuracy at larger temporal resolutions using deep learning and transfer learning techniques. Atmospheric Enviroment, 214, Article 116885.

[35] Fong, I. H., Li, T., Fong, S., Wong, R. K., & Tallon-Ballesteros, A. J. (2020). Predicting concentration levels of air pollutants by transfer learning and recurrent neural network. Knowledge-Based Systems, 192, Article 105622.