Power Load Forecasting Based on the Combined Model of LSTM and XGBoost

Chen Li[1,2], Zhenyu Chen[2,3], Jinbo Liu[4], Dapeng Li[2], Xingyu Gao[5], Fangchun Di[2], Lixin Li[2], Xiaohui Ji[1*]

[1]School of Information Engineering, China University of Geosciences, Beijing 100083, China

[2]Beijing Key Laboratory of Research and System Evaluation of Power Dispatching Automation Technology (China Electric Power Research Institute), Beijing 100192, China

[3]Big Data Center, State Grid Corporation of China, Beijing 100031, China

[4]Power Dispatching and Control Center, State Grid Corporation of China, Beijing 100031, China

[5]Institute of Microelectronics, Chinese Academy of Sciences, Beijing 100029, China

[*]Corresponding author, E-mail: xhji@cugb.edu.cn

ABSTRACT
Accurate power load forecasting can provide effective and reliable guidance for power construction and grid operation, and plays a very important role in the power grid system. In order to improve the accuracy of power load forecasting, this paper proposes a combined forecast model based on LSTM and XGBoost. The LSTM forecast model and the XGBoost forecast model are firstly established and the power load is predicted by using the two models respectively. Then the combined model predicts the power load by using the error reciprocal method to combine the results from the two single models. Through the experimental verification of the power load data of The Electrician Mathematical Contest in Modeling, the forecast error of the combined model we got is 0.57%, which is significantly lower than the single forecast model.

CCS Concepts
• **Computing methodologies→Artificial intelligence**

Keywords
Power load forecasting; Combined Model; LSTM; XGBoost

1. INTRODUCTION
Power load forecasting is to forecast the power consumption of an area based on historical power load data [1]. With the development of smart grids, the requirements for power load forecast accuracy have gradually increased [2-5]. According to the relevant research results, for every 1% increase in the error of the power load forecast, the annual operating cost and maintenance cost of the power system will increase by 10 million pounds [6]. In order to ensure the normal operation of the power system, it is necessary to continuously improve the accuracy of power load forecasting.

Common power prediction methods include time series analysis, traditional machine learning methods, and deep learning methods. Time series analysis includes Autoregressive (AR) methods, Exponential smoothing (ES) methods, and Autoregressive Integrated Moving Average model (ARIMA) methods [7-9]. The disadvantage of the time series analysis is that when there is a special time period in the time series and there is no adaptive law, you can't use it. Traditional machine learning methods include support vector machine (SVM), random forest, BP neural network, generalized regression neural network and boosting [10-14]. The XGBoost algorithm is one of the boosting algorithms and has achieved good results in many forecast fields [15-18]. Deep learning methods include Recurrent Neural Network (RNN) and Convolutional Neural Networks (CNN) [19-20]. The deep learning method has achieved good results in power load forecasting with a powerful network model structure. The Long- and Short-Term Memory Network (LSTM) is an improved RNN model that has solved the gradient disappearance and explosion problems of RNN to make the network to efficiently process long-term time series data [21]. LSTM has been applied to power load forecasting, and its prediction error is significantly lower than other methods [22-23].

The combined forecast model combines one or more of the above methods to perform power load prediction, which can improve the prediction accuracy [24-26]. Previous studies have combined particle swarm optimization BP neural networks with generalized regression neural networks [27]. There are also studies that combine regression analysis models, elastic coefficient method, grey prediction models, and time series models [28]. The accuracy of their combined models is higher than the single forecast model.

Because of the good performance of LSTM and XGBoost in power load forecasting, this paper combines LSTM and XGBoost models to improve the forecast accuracy of the power load. The experimental results show that our combined model has higher accuracy than the single model.

2. BACKGROUND
This section describes the LSTM and XGBoost methods used in this article.

PRAI '19, August 26–28, 2019, Wenzhou, China

ACM ISBN 978-1-4503-7231-2/19/08…$15.00
https://doi.org/10.1145/3357777.3357792

mechanism overcomes the shortcomings of RNN's easy gradient disappearance or explosion, making network convergence faster and better. The unit structure of the LSTM is shown in Figure 1.

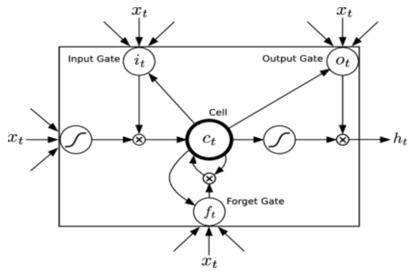

Figure 1. LSTM unit

Forget gate: $f_t = \sigma\left(W_f[h_{t-1}, x_t] + b_f\right)$ (1)

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

Cell: $\tilde{c}_t = tanh(W_c[h_{t-1}, x_t] + b_c)$ $c_t = f_t \circ c_{t-1} + i_t \circ \tilde{c}_t$ (3)

Output gate: $o_t = \sigma(W_o[h_{t-1}, x_t] + b_o)$ (4)

Final output: $h_t = o_t \circ tanh(c_t)$ (5)

In formula (1), W_f is the weight matrix of the forget gate, $[h_{t-1}, x_t]$ is the connection of two vectors into a longer vector, b_f is the bias term of the forget gate, and σ is the sigmoid function. In the formula (2), W_i is the weight matrix of the input gate, and b_i is the bias term of the input gate. Formula (3) calculates the unit state c_t at the current time. It is multiplied by the element of the last unit state c_{t-1} by the forget gatef_t, and then multiplied by the element of the currently input unit state $\tilde{c}_t$ by the input gatei_t, and then the sum of the two products is generated. The LSTM is combined with the current memory $\tilde{c}_t$ and the long-term memory c_{t-1} to form a new unit statec_t. Due to the control of the forget gate, it can save information long before a long time. Due to the control of the input gate, it can prevent the current insignificant content from entering the memory. The output gate output is calculated by formula (4), and the LSTM output is finally obtained by formula (5).

2.2 XGBoost

XGBoost is an improved algorithm for the boosting algorithm that integrates many weak classifiers to form a strong classifier [31]. XGBoost continuously iterates during the learning process. A new tree is generated during iteration to fit the residuals of the previous tree. As the iteration progresses, the accuracy increases. The tree model it uses is the CARTs regression tree model [32]. The XGBoost model can be expressed as formula (6).

$$\hat{y}_i = \sum_{t=1}^{n} f_t(x_i) \quad f_t \in F \tag{6}$$

Where n is the number of trees; f_t is a function in function space F; $\hat{y}_i$ is the predicted value; x_i is the ith sample of the input; F is the set of all possible CARTs.

Iteration retains the original model and adds a new function to the model. A function corresponds to a tree, and the newly generated tree fits the residual of the last prediction. The iterative process is shown in formula (7).

$$\hat{y}_i^{(0)} = 0 \tag{7}$$

$$\hat{y}_i^{(1)} = f_1(x_i) = \hat{y}_i^{(0)} + f_1(x_i)$$

$$\hat{y}_i^{(t)} = \hat{y}_i^{(t-1)} + f_t(x_i)$$

The objective function of XGBoost is shown in formula (8) and (9).

$$\mathrm{Obj} = \sum_{i=1}^{n} l(y, \hat{y}) + \sum_{k=1}^{K} \Omega(f_k) \tag{8}$$

$$\Omega(f_k) = \gamma T + \lambda \frac{1}{2} \sum_{j=1}^{T} \omega_j^2 \tag{9}$$

In formula (8), $\sum_{i=1}^{n} l(y, \hat{y})$ is used to measure the difference between the predicted score and the true score, and $\sum_{k=1}^{K} \Omega(f_k)$ is the regularization term. In formula (9), T represents the number of leaf nodes, ω represents the fraction of leaf nodes, γ is used to control the number of leaf nodes, and γ is used to control the score of leaf nodes not to be too large. The goal of regularization is to choose a simple prediction function to prevent the model from over fitting. When the regularization parameter is zero, XGBoost degenerates into a traditional boosting model. The iteration of the model uses additive training to further minimize the objective function [33]. The objective function is updated to formula (10) on iteration.

$$\ell^{(t)} = \sum_{i=1}^{n} l\left(y_i, \hat{y}_i^{(t-1)} + f_t(X_i)\right) + \Omega(f_t) \tag{10}$$

In order to find f_t that can minimize the objective function, XGBoost approximates it by Taylor's second-order expansion at f_t =0, and generalizes the Taylor series of the loss function to the second order. Therefore, the objective function is approximated by formula (11).

$$\ell^{(t)} \simeq \sum_{i=1}^{n} \left[l\left(y_i, \hat{y}_i^{(t-1)} + f_t(X_i)\right) + \frac{1}{2} h_i f_t^2(X_i) \right] + \Omega(f_t) \tag{11}$$

Formula (11) adds up the loss function values for each sample as shown in formula (12).

$$Obj \simeq \sum_{i=1}^{n} \left[g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i) \right] + \Omega(f_t) \tag{12}$$

$$= \sum_{i=1}^{n} [g_i w_q(x_i) + \frac{1}{2} h_i w_q^2(x_i)] + \Omega(f_t) + \gamma T + \lambda \frac{1}{2} \sum_{j=1}^{T} \omega_j^2$$

$$= \sum_{j=1}^{T} \left[\left(\sum_{i \in I_j} g_i \right) w_j + \frac{1}{2} \left(\sum_{i \in I_j} h_i + \lambda \right) w_j^2 \right] + \gamma T$$

In the formula (12), $g_i = \partial\, \hat{y}^{t-1}\, l(y_i, \hat{y}^{t-1})$, $h_i = \partial^2\, \hat{y}^{t-1}\, l(y_i, \hat{y}^{t-1})$.

Formula (12) rewrites the objective function into a unitary quadratic function with respect to the leaf node scoreω, and the optimal ω and objective function values obtained by the solution are shown in formula (13) and (14), respectively.

$$\omega_j^* = -\frac{G_j}{H_j + \lambda} \tag{13}$$

$$Obj = -\frac{1}{2} \sum_{j=1}^{T} \frac{G_j}{H_j + \lambda} + \lambda T \tag{14}$$

In the formula (13) and (14),$G_j = \sum_{i \in I_j} g_i$, $H_j = \sum_{i \in I_j} h_i$.

3. POWER LOAD FORECASTING BASED ON THE COMBINED MODEL OF LSTM AND XGBOOST

This paper proposes a method of combining LSTM and XGBoost models for power load forecasting. The method firstly preprocesses the power load data, then establishes the forecast model of LSTM and XGBoost, and finally uses the combined method to calculate the final forecast result.

3.1 Data Processing

Some errors may occur during the data collection process, resulting in some missing data. This paper assumes that the data is similar in a short period of time. Then the missing data is got by calculating the mean value at the same time of the first two days and the last two days.

In order to get the input sequence X and the output y for forecast, a time-series data set of the power load is constructed as shown in Figure 2. For the current forecast time t, the power load data at the time t-n to t-1 is taken as the input sequence X of the model, and the power load data at the current time is taken as the output y.

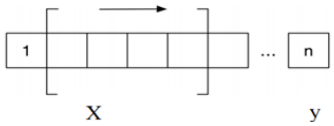

Figure 2. Input and output data

3.2 LSTM Forecast Model

This paper first uses LSTM for power load forecast. The constructed LSTM is shown in Figure 3. The input layer performs a preliminary processing of the original load time series to meet the network input requirements. The input data is normalized by the standardization of the dispersion as shown in formula (15), and the data is linearly mapped to the range of 0-1.

$$x_i^* = \frac{x_i - x_{min}}{x_{max} - x_{\min}} \tag{15}$$

In formula (15), x_{min} is the minimum value in the input data, x_{max} is the maximum value in the input data, and x_i is the data to be normalized.

The hidden layer uses LSTM cells and Dropout to build a double-layered LSTM. The more layers of the LSTM neural network module, the stronger the learning ability. But the excessive number of layers will make the network training difficult to converge. So the number of layers in the training process generally does not exceed 3 [34]. This article uses 2-layer LSTM. A Dropout layer is added to each hidden layer in this paper. In the forward propagation, the activation value of the neuron stops working with a specified probability, which enhances the generalization of the model and prevents over-fitting.

The output layer uses the fully connected layer to reduce the dimensionality of the results, and then deformalizes the predicted data to obtain the predicted result.

The network training uses the Adam optimization algorithm [35], and the prediction is performed point by point using an iterative method.

3.3 XGBoost Forecast Model

The XGBoost model needs to determine three parameters when it is predicted: General Parameters, Booster Parameters, and Learning Task Parameters. The general parameters determine the type of ascending model during the ascent and often use a tree or a linear model; the booster parameters depend on the selected ascending model; the learning task parameters define the learning tasks and the corresponding learning objectives.

The booster parameters have a great influence on the performance of the algorithm. The booster parameters including the learning rate, the maximum height of the tree and the random sampling ratio[36]. Because the maximum height of the tree has a large impact on the final result, the parameter is first tuned. The tuning method is to first give other parameters initial values, and the important parameters are set to common typical values. The n_estimators is set as 1000; subsample is set as 0.8; colsample bytree is set as 0.8 and the learning rate is set as 0.1. Other parameters are set to default values. Change the maximum height of the tree to compare the loss of test data. When the maximum height of the available tree is 5, the error of the test data is the smallest.

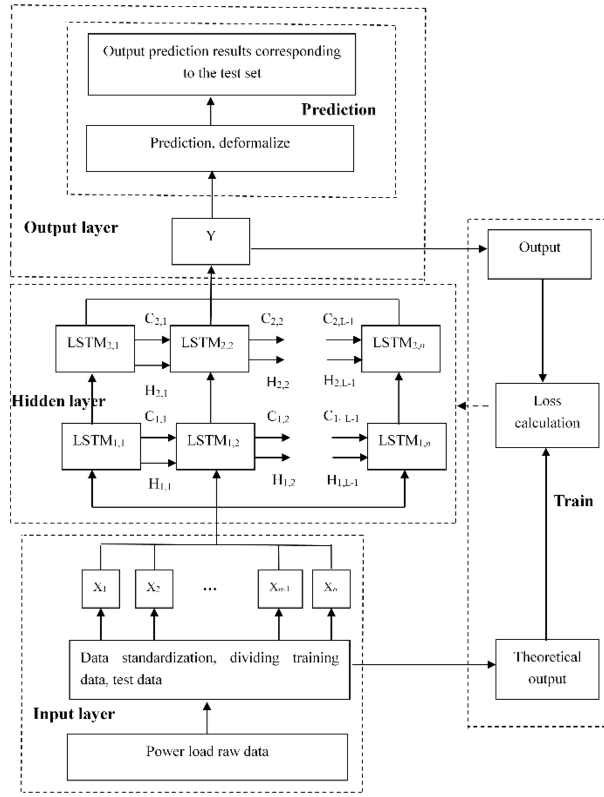

Figure 3. LSTM model for power load forecasting

Table 1. List of best parameters of XGBoost

Class	Parameter	Value
General Parameters	Booster	Gbtree
Booster Parameters	learning rate	0.05
Booster Parameters	max_depth	5
Booster Parameters	min_child_weight	1
Booster Parameters	Subsample	0.2
Booster Parameters	n_estimators	1000
Learning Task Parameters	eval-metric	MAPE
Learning Task Parameters	Objective	Gamma

After determining the maximum height of the tree, other parameters are tuned. First, the ranges of other parameters are given, and the traversal method is used to get the best combination [37]. The learning rate range is set to 0.01-0.9; the iteration number range is set to 100-10 000; the random sampling ratio range is set to 0-0.9. Through the search traversal, the optimal parameters of the XGBoost model tree used in this paper are finally determined as shown in Table 1.

3.4 Combined Forecast Model

After obtaining the forecast results of LSTM and XGBoost as described in 3.2 and 3.3, the weighted combination of the two results is performed by the error reciprocal method as shown in formula (16). ω_i is the weight coefficient, and f_{it} is the predicted value obtained by LSTM or XGBoost. The calculation of the weight coefficient ω_i is as shown in formulas (17) and (18), where ε_1 and ε_2 are the MAPE (Mean Absolute Percentage Error) errors of LSTM and XGBoost, respectively, and the MAPE error is calculated as shown in formula (19).

$$f_t = \omega_1 f_{1t} + \omega_2 f_{2t},\ t = 1, 2 \ldots n \tag{16}$$

$$\omega_1 = \frac{\varepsilon_2}{\varepsilon_1 + \varepsilon_2} \tag{17}$$

$$\omega_2 = \frac{\varepsilon_1}{\varepsilon_1 + \varepsilon_2} \tag{18}$$

$$MAPE = \frac{1}{n}\sum_{t=1}^{n}\frac{|y_t - \hat{y}_t|}{y_t} * 100\% \tag{19}$$

It can be seen from formula (16), (17) and (18) that the method assigns a larger weight coefficient to the model with the smaller error, so the error of the combined model is decreased and the overall forecast accuracy is improved.

4. EXPERIMENTAL EVALUATIONS

The method is implemented in Python. Keras is used to implement LSTM, and py-xgboost is used to implement XGBoost. NVIDIA Tesla P100 GPU is used to accelerate the training process. The proposed combined model was tested using the power load data of the Electrician Mathematical Contest in Modeling. The sampling period of the electrical load data was set to 15 minutes, and 96 sets of data were generated each day. The training data of the LSTM model and the XGBoost model is from January 1, 2009 to January 7, 2015. The test data is from January 8, 2015 to January 10, 2015, and the power load from January 9, 2015 to January 10, 2015 is forecasted. The comparisons between the forecasted values and the actual values of each model are shown in Figure 4 to Figure 6. The forecast error is shown in Figure 7. The MAPE values of the model are shown in Table 2.

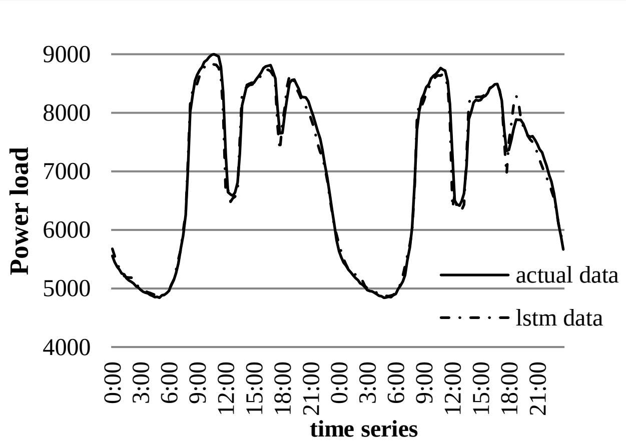

Figure 4. Comparison of the actual data and the forecasted data LSTM

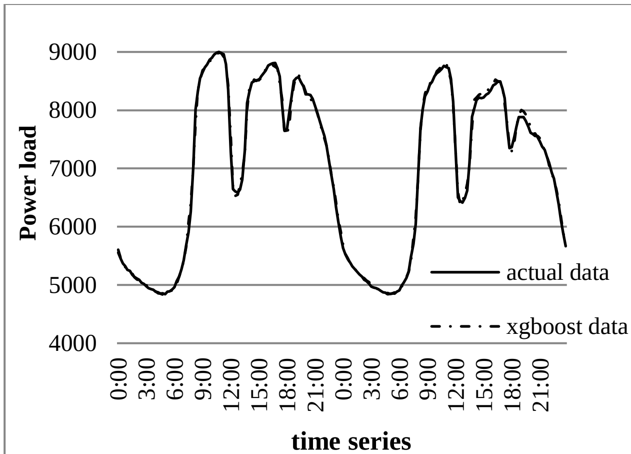

Figure 5. Comparison of the actual data and the forecasted data of XGBoost

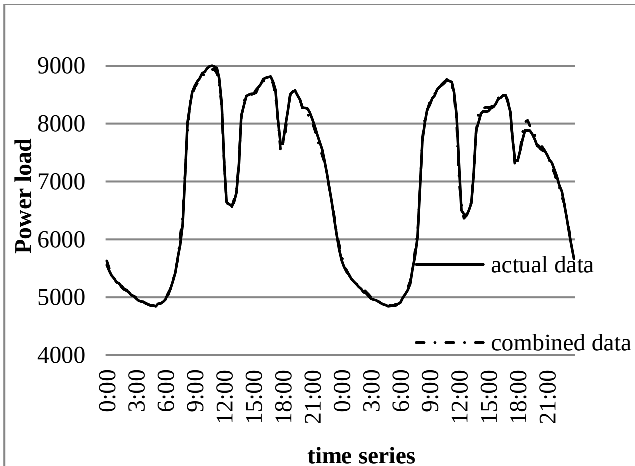

Figure 6. Comparison of the actual data and the forecasted data of the combined model

Table 2. MAPE

Model	LSTM	XGBoost	Combined model
MAPE	0.0137	0.0060	0.0057

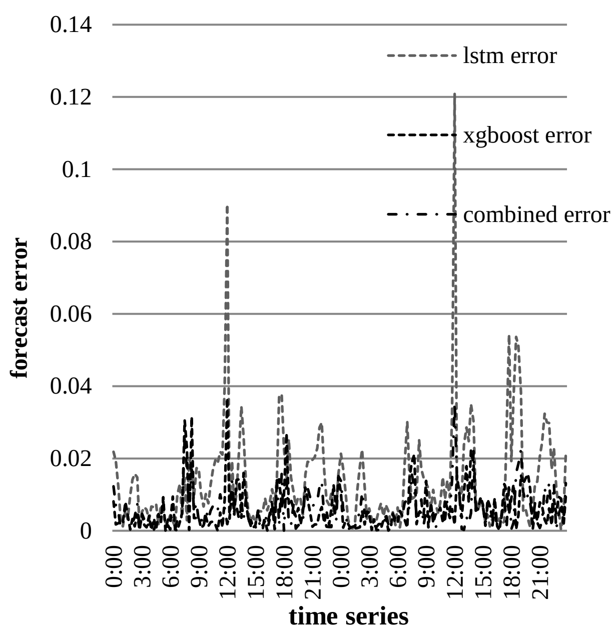

Figure 7. Comparison of the errors

It can be seen from Figure 4 to 6 that the fitting precision of the LSTM model is lower than that of XGBoost model. The fitting accuracy of the combined model is higher than that of the LSTM model and the XGBoost model. The forecast curve of the combined model is basically consistent with the actual curve. From Figure 7 we can see that the combined model reduces the overall error and has the highest accuracy. From Table 2 we can see that the forecast accuracy of the LSTM, XGBoost and the combined model is 1.37%, 0.6%, and 0.57%, respectively. The forecast accuracy of the three models is higher than that in [27], which is 4.57%.

5. CONCLUSION

In this paper, a combined model based on LSTM and XGBoost is presented. The combined model is got by using the error reciprocal method after the forecast results are achieved from the LSTM and XGBoost models respectively. The combined model assigns a larger weight coefficient to the model with the smaller error, so the error of the combined model is decreased and the overall forecast accuracy is improved.

6. ACKNOWLEDGMENTS

The research of the paper is supported by the Open Fund of Beijing Key Laboratory of Research and System Evaluation of Power Dispatching Automation Technology (China Electric Power Research Institute) (No.DZB51201801541), Science and Technology Program of State Grid Corporation of China (52110418002A), Science and Technology Innovation Program of China Electric Power Research Institute (5242001700DZ), and Sichuan Science and Technology Program (2019YFSY0016).

7. REFERENCES

[1] Zhao Y. Research on short term power load forecasting method based on data mining technology in Nanjing area [D]. North China Electric Power University, 2016.

[2] Yang S C , Yao J G, Wang K , et al. Study on the Supporting System for Smart Grid Control Center[J]. Advanced Materials Research, 2011, 354-355:7.

[3] D S.H. Application of outlier mining in power load forecasting[C]// International Conference on Computer Application & System Modeling. IEEE, 2010.

[4] Li H. Short Term Load Forecasting by Adaptive Neural Network [J]. IOP Conference Series Materials Science and Engineering, 2018, 449(1):012028.

[5] Wang N.L, Fu P, Chen D.G, et al. Application of Big Data Method in Optimal Load Dispatching of Power Plant[J]. Proceedings of the CSEE, 2015(1).

[6] Zhang Y.H, Qiu C.M, He X, et al. A Short-Term Load Forecasting Based on LSTM Neural Network[J]. Electric Power Information and Communication Technology, 2017.

[7] Baharudin Z, Kamel N. Autoregressive method in short term load forecast[C]// IEEE International Power & Energy Conference. 2009.

[8] Ji P, Xiong D, Wang P, et al. A Study on Exponential Smoothing Model for Load Forecasting[C]// Asia-pacific Power & Energy Engineering Conference. 2012.

[9] Zhu X , Shen M . Based on the ARIMA model with grey theory for short term load forecasting model[C]// International Conference on Systems & Informatics. IEEE, 2012.

[10] Ma W. Power System Short-Term Load Forecasting Based on Improved Support Vector Machines[C]// International Symposium on Knowledge Acquisition & Modeling. IEEE Computer Society, 2008.

[11] Huo J, Shi T, Jing C. Comparison of Random Forest and SVM for electrical short-term load forecast with different data sources[C]// IEEE International Conference on Software Engineering & Service Science. 2017.

[12] Lahouar A, Ben H S J. Day-ahead load forecast using random forest and expert input selection[J]. Energy Conversion and Management, 2015, 103:1040-1051.

[13] Papadopoulos S, Karakatsanis I. Short-term electricity load forecasting using time series and ensemble learning methods[C]// Power & Energy Conference at Illinois. 2015.

[14] Qiu X, Suganthan P N, Amaratunga G A J. Ensemble Incremental Learning Random Vector Functional Link Network for Short-term Electric Load Forecasting[J]. Knowledge-Based Systems, 2018:S0950705118300236.

[15] Luckner M, Topolski B, Mazurek M. Application of XGBoost Algorithm in Fingerprinting Localization Task[J]. 2017.

[16] Li L, Situ R, Gao J, et al. A Hybrid Model Combining Convolutional Neural Network with XGBoost for Predicting Social Media Popularity[C]// Acm on Multimedia Conference. 2017.

[17] Pan B. Application of XGBoost algorithm in hourly PM2.5 concentration prediction[J]. IOP Conference Series: Earth and Environmental Science, 2018, 113:012127.

[18] Wang J, Lou C, Yu R, et al. Research on Hot Micro-blog Forecast Based on XGBOOST and Random Forest[M]// Knowledge Science, Engineering and Management. 2018.

[19] Hayashi Y, Iwamoto S. Long-Term Load Forecasting Using Improved Recurrent Neural Network[J]. Electrical Engineering in Japan, 2010, 114(8):41-54.

[20] Li L, Ota K, Dong M. Everything is Image: CNN-based Short-Term Electrical Load Forecasting for Smart Grid[C]// International Conference on International Symposium on Pervasive Systems. 2017.

[21] Graves A. Long Short-Term Memory[M]// Supervised Sequence Labelling with Recurrent Neural Networks. 2012.

[22] Li T, Wang B , Zhou M , et al. Short-term load forecasting using optimized LSTM networks based on EMD[J]. 2018.

[23] Kong W, Dong Z.Y, Jia Y, et al. Short-Term Residential Load Forecasting based on LSTM Recurrent Neural Network[J]. IEEE Transactions on Smart Grid, 2017:1-1.

[24] Mao L.F, Yao J.G, Jin Y.S, et al. Theoretical study of combination model for medium and long term load forecasting[J]. Zhongguo Dianji Gongcheng Xuebao/Proceedings of the Chinese Society of Electrical Engineering, 2010, 30(16):53-59.

[25] Liu L. Short-Term Power Load Forecasting Based on SARIMA and SVR [D]. Nanchang: East China University of Technology, 2018.

[26] Duan J. Research on Short-term Load Forecasting Method Based on Neural Network [D]. Lanzhou: Lanzhou University, 2017.

[27] Chen H, Li X.R, Leng H, et al. Bidirectional Combined Short-term Load Forecasting by Using PSO and GRNN[J]. Proceedings of the CSU-EPSA, 2018.

[28] Cheng J, Li Y, Xia X.Y, et al. Economic-electricity conduction prediction model based on dual combination prediction[J]. Journal of Electric Power Science and Technology, 2018.

[29] Graves A. Supervised Sequence Labelling with Recurrent Neural Networks[M]. 2012.

[30]https://www.researchgate.net/figure/Long-Short-term-Memory-Cell-Reprinted-from-4_fig2_308896333

[31] Chen T, Guestrin C. XGBoost: A Scalable Tree Boosting System[J]. 2016.

[32] Trendowicz A, Jeffery R. Classification and Regression Trees[J]. International Journal of Public Health, 2014, 57(1):243-246.

[33] Gómez-Ríos A, Luengo J, Herrera F. A Study on the Noise Label Influence in Boosting Algorithms: AdaBoost, GBM and XGBoost[J]. 2017.

[34] Zhou Y.S. PM2.5 Prediction Based on Lstm Neural Network[D]. Changsha: Hunan University 2018.

[35] Wang S, Sun J, Xu Z. HyperAdam: A Learnable Task-Adaptive Adam for Network Training[J]. 2018.

[36] Luckner M, Topolski B, Mazurek M. Application of XGBoost Algorithm in Fingerprinting Localisation Task[J]. 2017.

[37] Complete Guide to Parameter Tuning in XGBoost https://www.analyticsvidhya.com/blog/2016/03/complete-guide-parameter-tuning-xgboost-with-codes-python/

[38] Wang X. Study on power system load prediction method for medium and long term based on comprehensive model [J]. Huadian Technology, 2013, 35(6):40-41.