



Transformer-Based Multi-industry Electricity Demand Forecasting

Zhenli Deng¹, Fangzhao Deng¹, Meng Yang¹, Xingwu Guo¹, Jianan Si¹, Boning Yu¹,
and Boxiang Feng²(✉)

¹ State Grid Henan Electric Power Company Economic and Technological Research Institute,
Zheng Zhou, China

² School of Informatics, Xiamen University, Xiamen, China
boxiangfeng@stu.xmu.edu.cn

Abstract. The accuracy of electricity demand forecasting is closely related to the correctness of decision-making in the power system, ensuring stable energy supply. Stable energy supply is a necessary guarantee for socioeconomic development and normal human life. Accurate electricity demand forecasting can provide reliable guidance for electricity production and supply dispatch, improve the power system's supply quality, and ultimately enhance the security and cost-effectiveness of power grid operation, which is crucial for boosting economic and social benefits. Currently, research on electricity demand forecasting mainly focuses on the single-factor relationship between power consumption and economic growth, industrial development, etc., while neglecting the study of multiple influencing factors and considering different time dependencies.

To address this challenge, we propose a transformer-based forecasting model that utilizes transformer networks and fully connected neural networks (FC) for electricity demand forecasting in different industries within a city. The model employs the encoder part of the transformer to capture the dependencies between different influencing factors and uses FC to capture time dependencies. We evaluate our approach on electricity demand forecasting datasets from multiple cities and industries using various metrics. The experimental results demonstrate that our proposed method outperforms state-of-the-art methods in terms of accuracy and robustness. Overall, we provide a valuable framework in the field of electricity demand forecasting, which holds practical significance for stable power system operations.

Keywords: Electricity Demand Forecasting · Time Series Prediction · Transformer

1 Introduction

Electricity demand forecasting is crucial for optimizing power supply-demand structures [1]. With the evolving power industry, renewable energy growth, and unpredictable weather events, accurately predicting demand across regions and industries is essential. Recent research, shifting from traditional statistical methods to machine and deep

learning models, has improved accuracy and service provision [2]. Common methods include grey system analysis [3] and regression [1], while innovative deep learning models like LSTM [4] and GRU [5] show promising results. However, these models lack interpretability. Decision trees and gradient boosting algorithms enhance accuracy by learning complex patterns within time series [6]. Feature selection and processing, including dimensionality reduction, are crucial during forecasting. This paper builds on advanced time series prediction models, capturing dependencies between influencing factors and improving prediction accuracy by analyzing historical data dependencies. This paper contributes to electricity demand forecasting in the following aspects:

1. Firstly, the transformer model can capture the dependency relationship between different positions in a sequence, achieving context awareness. We leverage the advantages of the transformer model to analyze the dependencies between different feature factors, effectively capturing the complex relationships between multiple input variables and the target variable. This is crucial for improving the performance of the model.
2. Secondly, we capture the time dependencies between different historical time series through the decoder layer composed of fully connected networks. This can potentially improve the accuracy of the prediction results.
3. Finally, we have validated our proposed method on electricity demand datasets from different cities and industries in the real world to demonstrate its effectiveness in predicting city electricity consumption.

2 Related Work

2.1 Classical Statistical Methods

In the past century, classical statistical methods dominated time series prediction, relying on experts' experience and simple relationships, resulting in lower accuracy. Methods included time series analysis, regression, exponential smoothing, and grey forecasting. Time series analysis uses historical data to model power load changes, divided into autoregressive, moving average, and integrated processes [7]. It has fast convergence but overlooks internal factors. Regression predicts future electricity levels based on historical data, offering simplicity and generalization but limited adaptability [8]. Exponential smoothing averages past sequences to predict future trends but struggles with unstable sequences and complex factors [9]. Grey forecasting suits uncertainty, with ordinary models for exponential growth and optimized models for fluctuating sequences. Advantages include simplicity, fewer parameters, and strong mathematical foundations, but they struggle with longer forecasts [10]. Classical methods require small datasets and lack adaptability to complex relationships, making them suitable for monthly predictions but challenging for practical applications involving temporal and spatial aspects [11].

2.2 Machine Learning Methods

With the development of machine learning, a series of classical algorithms have emerged. Compared to traditional statistical methods, machine learning-based time series forecasting has the advantage of powerful nonlinear fitting capabilities, resulting in higher prediction accuracy. One of the most popular time series techniques for electricity demand

forecasting is Long Short-Term Memory (LSTM). Recurrent Neural Network (RNN) is a typical type of recurrent neural network that incorporates internal feedback connections and feedforward connections between processing units in different layers, enabling it to associate past information with present tasks. However, as the length of the time series increases, RNN struggles to learn long-term dependencies across distant time steps. LSTM, a special type of RNN, overcomes this limitation by incorporating three gates within the units to control internal states, thus addressing the vanishing gradient problem. As a result, it not only possesses the short-term dependency learning capability of RNN but also learns long-term dependencies. In literature [12], an algorithm for load forecasting is proposed based on the integration of LGBM and LSTM. In literature [13], the gate structure of LSTM is adjusted to reduce model parameters and improve computing speed. In literature [14], the strengths of both RNN and LSTM are combined for prediction, and an Attention mechanism is used to aggregate the prediction results of the two models, applied to small-scale monthly electricity sales datasets.

Transformer is a neural network model based on attention mechanisms, originally proposed by Google for natural language processing tasks such as machine translation, text summarization, and speech recognition. Compared to recurrent neural network models such as LSTM and GRU, which are representative of RNN and its variants, the Transformer model exhibits better parallelization and shorter training time. It performs well not only in processing long sequences but also in capturing contextual dependencies within sequences and internal dependencies between different sequences. As a result, it has found wide applications in various fields. The DehazeFormer approach proposed by Song et al. [15] modifies the Transformer model for image dehazing tasks. VideoBERT is a joint representation model based on Transformer for extracting representations from both image and language data, achieving excellent results in video content recognition datasets and serving as a fundamental architecture for multimodal fusion tasks [16]. Radford et al. proposed CLIP, a zero-shot learning method based on the ViT network, which combines language and image data and achieved promising results in various tasks [17]. Roy et al. introduced a multimodal fusion attention mechanism for extracting class labels from multimodal data using Transformer with cross-attention weights on input labels, and verified its performance on multimodal remote sensing classification tasks [18]. The relatively simple structure and outstanding performance of Transformer greatly enhance its application potential in the field of machine learning.

3 Proposed Method

This section provides a detailed description of the method proposed in this paper for predicting the electricity demand of different industries in cities. Firstly, we propose a general model framework for performing this task. Then, we analyze the different variables that influence the prediction of city electricity demand based on time series theory and select relevant covariance features. Finally, we provide a detailed description of the transformer-based model prediction framework and validate it on real-world datasets. Through this research, we aim to provide an accurate method for forecasting city electricity demand to support relevant decision-making and planning.

3.1 Overall Framework

In this article, we employ the combination of transformers and fully connected neural networks, making full use of the contextual learning ability of transformers, and considering the dependency among multiple time series. Our algorithm framework, as shown in Fig. 1, consists of two main parts. The encoder layer of the transformer is mainly responsible for capturing the spatial dependencies between different features, while the fully connected neural network primarily captures the temporal dependencies among different time series.

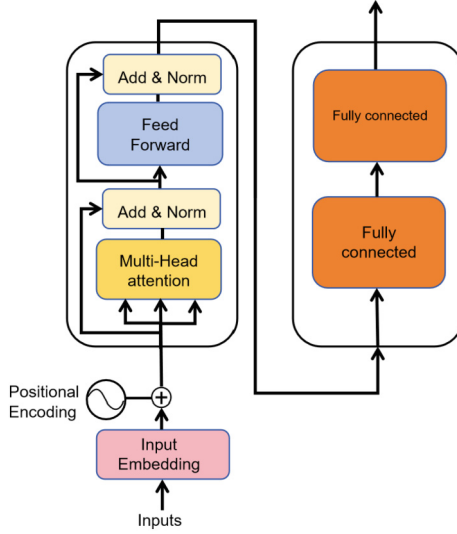


Fig. 1. Prediction model framework

Our proposed algorithm framework for electricity demand forecasting can be viewed as two stages: feature aggregation using the encoder layer of the transformer to learn the relationship between features and past electricity demand values, and prediction using the fully connected neural network layer. The encoder layer of the transformer is used to aggregate various features from the input time series data and learn their nonlinear relationship with electricity demand. The fully connected neural network layer then uses the aggregated features to predict future electricity demand values.

A high-dimensional time series regarding electricity consumption across various dimensions is as follows:

$$\{Y_t = (y_{t1}, y_{t2}, \dots, y_{tp}), t = 1, 2, \dots, N\} \tag{1}$$

as well as time series data for related covariates:

$$\{X_t = (x_{t1}, x_{t2}, \dots, x_{tp}), t = 1, 2, \dots, N\} \tag{2}$$

In this context, ‘t’ represents the time step, and ‘p’ represents different dimensions of time series.

3.2 Feature Extraction Module

In this section, we analyze and address the factors affecting electricity demand from meteorological and social perspectives and construct relevant covariates.

From a social perspective, residential electricity consumption levels are generally lower during working days compared to holidays. The age distribution of the population in a given region also affects electricity demand. For instance, regions with a higher number of students during summer vacation experience a significant increase in electricity demand. Therefore, it is necessary to quantify holidays and summer vacation. Holidays are non-numeric data and need to be encoded to transform them into numerical values. For the “day of the week” data, we use one-hot encoding. For the data on “whether it is a holiday” and “festival type,” we use 0–1 encoding, where 1 represents a holiday, 0 represents a working day, and 1 represents a specific festival, while the rest are represented as 0.

From a meteorological perspective, weather is the most important factor influencing electricity demand. Therefore, studying and analyzing meteorological conditions is an important step in improving the accuracy of the forecasting model.

From the mechanism of variation, it is known that temperature has the most significant impact among all meteorological factors, especially in some extreme natural environments. During cold winters and hot summers, electricity generation is significantly higher than in other seasons. Therefore, we construct three types of covariates to characterize temperature changes in the region: average temperature, maximum temperature, and minimum temperature. We also incorporate humidity information to build feature sequences that capture the meteorological impacts on electricity demand.

3.3 Time Series Prediction Module

We first introduce how to learn the dependencies of complex features.

Transformer, as the most advanced model in natural language processing, has been widely used due to its efficiency and strong contextual awareness.

In Transformer, the input to the Encoder is a sequence of text, and the output is a feature vector that represents the semantic information of the input text. The input to the Decoder is a specific token, based on which it generates a new sequence of text, and the output is a sequence of text. The Encoder is typically used for text encoding and representation learning. Therefore, we can use the encoder layers of Transformer for feature encoding and representation learning. The encoder layers primarily include four components: Positional Encoding, Multi-Head Attention, Add and Norm, and Feedforward and Add and Norm.

1. In the Positional Encoding positional encoding is performed using sine and cosine functions, as shown in the following formula:

$$PE(pos, 2i) = \sin(pos/10000^{2i/d_{model}}) \quad (3)$$

$$PE(pos, 2i + 1) = \cos(pos/10000^{2i/d_{model}}) \quad (4)$$

Here, pos represents the position of the feature in the entire sequence, and ‘ i ’ refers to the dimension of the feature vector. After positional encoding, we obtain an

encoding array X_{pos} that is completely consistent with the input dimension. When this encoding array is added to the original feature embeddings, we obtain new feature embeddings:

$$X_{embedding} = X_{embedding} + X_{pos} \quad (5)$$

2. The multi-head self-attention mechanism calculates the similarity between each input vector and all other input vectors, and then weights and sums them to obtain a new representation for each input vector. The mathematical expression for multi-head self-attention is as follows:

$$\text{Multihead}(Q, K, V) = \text{Concat}(\text{head}_1, \text{head}_2, \dots, \text{head}_h)W^o \quad (6)$$

Among them,

$$\text{head}_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V) \quad (7)$$

In Eq. (6), Q(Query), K(Key), V(Value) represents three vectors obtained from the input sequence through three linear mapping layers, with dimensions d_q , d_k , and d_v respectively. ‘Concat’ represents the concatenation function, which combines all the output results of head_i .

In Eq. (7), $W_i^Q \in R^{s \times d_k}$, $W_i^K \in R^{s \times d_k}$, $W_i^V \in R^{s \times d_k}$, $W_i^O \in R^{hd_v \times s}$, they respectively represent the weight matrices for the Q, K, and V vectors of the i -th ‘head’, and the weight matrix for the final output after dimension reduction. Here, it is mentioned that $d_k = d_v = s/h$. The computation of the attention mechanism is as follows:

$$\text{Attention}(Q, K, V) = \text{SoftMax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (8)$$

In Eq. (8), $d_h = s/h$, ‘SoftMax’ represents an activation function, while $\sqrt{d_k}$ is used to transform the attention matrix into a standard normal distribution.

3. In the ‘‘Add and Norm’’ section, the input ‘x’ from the previous layer is connected with the output from the previous layer through residual connections.
4. In the ‘‘Feedforward and Add and Norm’’ section, the feature representation is obtained by passing the input through a feedforward network, which includes linear mappings and activation functions:

$$X_{hidden} = \text{Relu}(X_{hidden} * W_1 * W_2) \quad (9)$$

In the Eq. (9), W_1 and W_2 are the weights of the two linear layers, and ‘Relu’ represents the activation function.

Next, we will discuss how to learn the temporal dependencies of different historical sequences.

A fully connected neural network is a multi-layer perceptron structure.

We use a 2-layer fully connected network to learn the nonlinear temporal dependencies of each time segment. In the current connection layer ‘ l ’, we have:

$$X^l = f(W^l X^{l-1} + b^l) \quad (10)$$

In the equation, X^l represents the output of the current connection layer ' l ', W^l represents the weights of the current layer, b^l represents the bias of the hidden layer, and ' $f()$ ' represents the nonlinear activation function. In this paper, we choose SoftMax as the activation function.

4 Proposed Method

4.1 Experimental Dataset

The dataset used in this paper includes the electricity demand and related covariate information for 13 industries from January 1, 2020, to January 31, 2023. The dataset is divided into a training set and a testing set in a 4:1 ratio. The training set consists of electricity demand data from January 1, 2020, to June 18, 2022, which is used for model training. The testing set includes electricity demand data from June 19, 2022, to January 31, 2023. Additionally, the dataset also includes the meteorological feature data and holiday feature data constructed in the previous section. The data has been preprocessed to eliminate outliers and missing values. We use Prophet, GBDT, and CNN-LSTM as experimental baseline methods.

4.2 Data Pre-processing

In this article, we utilize the min-max normalization method, which linearly transforms data to a specified range to eliminate dimensional impact. The commonly used ranges are $[0, 1]$ or $[-1, 1]$:

$$X^* = \frac{x - x_{min}}{x_{max} - x_{min}} \quad (11)$$

In the Eq. (11), x represents the electricity demand data, while x_{max} and x_{min} represent the maximum and minimum values of the data, respectively.

Furthermore, in terms of the loss function, we use the Mean Squared Error (MSE) function to measure the average difference between the actual observed values and the predicted values. As shown in Eq. (12), Y_i represents the predicted electricity demand at the current time step, \hat{Y}_i represents the true electricity demand at the current time step, and n represents the number of training samples. Additionally, we utilize the Adam optimizer [19] to optimize the model gradients.

$$MSE_{loss} = \frac{1}{n} \sum_{i=1}^n (Y_i - \hat{Y}_i)^2 \quad (12)$$

4.3 Experimental Results and Analysis

The formulas for the daily average error indicator and the monthly average error indicator are as follows:

$$\text{month_degree}_{error} = \frac{\sum_{i=1}^n y_{pred}(i) - \sum_{i=1}^n y_{true}(i)}{\sum_{i=1}^n y_{true}(i)} \quad (13)$$

$$\text{day_degree}_{error} = \frac{1}{n} \sum_{i=1}^n \left| \frac{y_{pred}(i) - y_{true}(i)}{y_{true}(i)} \right| \quad (14)$$

In the formulas, $y_{pred}(i)$ represents the predicted electricity demand for the i -th day, and $y_{true}(i)$ represents the true electricity demand for the i -th day.

In the experiment, we conducted electricity demand prediction tasks for different industries. Here is the specific industry breakdown, consisting of 13 industries. For the sake of readability, we use abbreviations to represent each industry. 1) Urban and rural residents' electricity demand (Ure); 2) Agriculture, forestry, animal husbandry, and fishery (Afahf); 3) Accommodation and catering industry (Aci); 4) Construction business (Cb); 5) Real estate industry (Ri); 6) Industrial sector (Is); 7) Information transmission, software, and information technology services industry (Isit); 8) Total electricity demand in society (Tes); 9) Financial business (Fb); 10) Wholesale and retail industry (Wi); 11) Leasing and business services industry (Rbi); 12) Public services and management services (Pm); 13) Transportation, warehousing, postal industry (Twp).

We selected the prediction results for each industry in August 2022 for comparison, with the monthly average error abbreviated as M-E and the daily average error as D-E. To ensure data confidentiality, we refer to the predicted city as X and the proposed prediction model based on Transformer and fully connected networks in this paper as Transformer-F:

Table 1. Error Results of Electricity Demand Forecast for 13 Industries of City X in August.

Industry	Model Error							
	Transformer-F		Prophet		GBDT		CNN-LSTM	
	M-E	D-E	M-E	D-E	M-E	D-E	M-E	D-E
Ure	-0.18	0.16	3.50	3.50	-0.38	0.38	-0.27	0.17
Afahf	-0.019	0.02	1.30	1.30	-0.20	0.30	-0.01	0.09
Aci	-0.01	0.01	0.18	0.19	-0.14	0.20	-0.02	0.01
Cb	-0.01	0.01	-0.05	0.07	-0.15	0.17	0.04	0.06
Ri	-0.02	0.01	0.13	0.14	-0.18	0.20	-0.02	0.01
Is	0.02	0.02	-0.04	0.04	0.12	0.11	0.33	0.33
Isit	-0.02	0.02	0.04	0.04	-0.06	0.08	0.02	0.03
Tes	0.03	0.01	0.04	0.65	-0.12	0.16	-0.04	0.01
Fb	-0.02	0.01	0.04	0.07	0.10	0.20	0.01	0.02
Wi	-0.01	0.11	0.17	0.21	-0.13	0.16	0.02	0.04
Rbi	-0.06	0.01	0.37	0.94	-0.20	0.22	-0.01	0.02
Pm	0.01	0.01	0.10	0.11	-0.09	0.17	-0.07	-0.1
Twp	0.02	0.01	-0.19	0.18	0.01	0.06	0.02	0.02

From Table 1, it can be seen that, compared to the comparative methods, the proposed Transformer-F prediction model in this paper has the lowest monthly average error and daily average error for 9 industries. This fully validates the effectiveness of the Transformer-F model. In contrast, the Prophet model performs the worst, indicating that the Prophet model may have limitations in predicting long-term trends. Additionally, the Prophet model typically requires the original data to have certain seasonal variations. If the training set lacks noticeable seasonal patterns, the Prophet model may struggle to effectively model the data.

Furthermore, the GBDT model has higher error results compared to the CNN-LSTM model and the proposed model in this paper. This is because tree-based models are generally not suitable for high-dimensional sparse data and are sensitive to parameter values, requiring careful tuning.

As shown in Fig. 2, we also presented the fitting performance of various models for the overall societal electricity demand in August. The overall societal electricity demand is defined as the sum of daily electricity demands across 13 industries. The curves of different colors in the graph represent the predicted values of different models.

From Fig. 2, it can be observed that our proposed model shows a good fit to the real curve, and the performance of the CNN-LSTM model is also considerable. However, the prediction results of the GBDT model are slightly worse compared to our proposed method and CNN-LSTM.

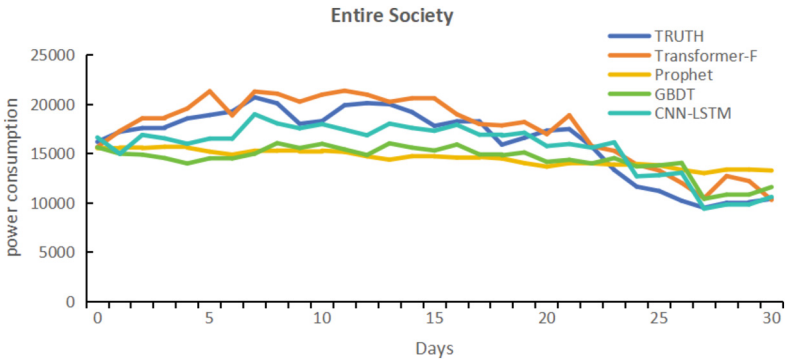


Fig. 2. Comparative results of different models in predicting the electricity demand of entire society in August.

In addition, our proposed method also outperforms the comparative methods in the segmented 13 industries.

As shown in Fig. 3 below, the curve fitting of the model for the real estate industry in August 2022 is very close to the actual situation on the ground. The GBDT method also performs well in some industries, but its performance is not as good as CNN-LSTM. Compared to the other three methods, the Prophet method performs relatively poorly across all industries:

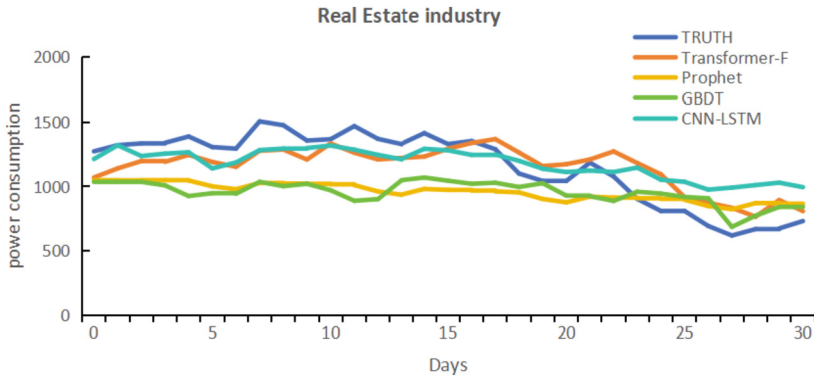


Fig. 3. Comparative results of different models in predicting the electricity demand of real estate industry in August.

Overall, the prediction results, as evidenced by comparing different numerical indicators and examining the fitting of different models to the real curves, demonstrate the effectiveness of our proposed model in electricity demand forecasting.

5 Conclusion

Translation: In this paper, we utilized time series statistical analysis methods to analyze historical electricity demand data. We established a multi-category electricity prediction model. We validated the effectiveness of our proposed method using a real-city electricity demand dataset. By comparing the prediction errors of different models across various industries, we demonstrated that our proposed method outperforms others in terms of accuracy.

Acknowledgement. This work is supported by the Research Funds from State Grid Henan (SGHAYJ00NNJS2310068);

References

1. Anjady, N.: Short-term hourly load forecasting using time-series modeling with peak load estimation capability. *IEEE Trans. Power Syst.* **16**(3), 498–505 (2001)
2. Li, J., Jiao, R., Wang, S., et al.: An ensemble load forecasting model based on online error updating. *Proc. CSEE* **43**(4), 1402–1412 (2023)
3. Vähäkylä, P., Hakonen, E., Léman, P.: Short-term forecasting of grid load using box-jenkins techniques. *Int. J. Electr. Power Energy Syst.* **2**(1), 29–34 (1980)
4. Jing, O., Lü, Y., Kang, Y., et al.: Short-term load forecasting method for integrated energy system based on ALIF-LSTM and multi-task learning. *Acta Energiæ Solaris Sinica* **43**(9), 499–507 (2022)
5. Deng, D., Li, J., Zhang, Z., et al.: Short-term electric load forecasting based on EEMD-GRU-MLR. *Power Syst. Technol.* **44**(2), 593–602 (2020)

6. Ahmad, A.S., et al.: A review on applications of ANN and SVM for building electrical energy consumption forecasting. *Renew. Sustain. Energy Rev.* **33**, 102–109 (2014)
7. Moghram, I., Rahman, S.: Analysis and evaluation of five short term load forecasting techniques. *IEEE Trans. Power Syst.* **4**(4), 1487–1491 (1989)
8. Amral, N., Ozveren, C.S., King, D.: Short term load forecasting using Multiple Linear Regression. In: *International Universities Power Engineering Conference*. IEEE (2007)
9. Pang, Y.: Research on market share prediction of highway passenger transport based on exponential smoothing method. *Am. J. Traffic Transp. Eng.* **8**(2) (2023)
10. Kumar, T.S., Rao, K.V., Balaji, M., et al.: Online monitoring of crack depth in fiber reinforced composite beams using optimization Grey model GM (1, N). *Eng. Fract. Mech.* **271** (2022)
11. Shumway, R.H., Stoffer, D.S., Shumway, R.H., Stoffer, D.S.: *Arima models. Time Series Analysis and Its Applications: With R Examples*, pp. 75–163 (2017)
12. Torres, J.F., Martínez-Álvarez, F., Troncoso, A.: A deep LSTM network for the Spanish electricity consumption forecasting. *Neural Comput. Appl.* **34**(13), 10533–10545 (2022)
13. Greff, K., Srivastava, R.K., Koutnik, J., et al.: LSTM: a search space odyssey. *IEEE Trans. Neural Netw. Learn. Syst.* **28**(10), 2222–2232 (2017)
14. Cui, Q., Sun, M., Na, M., et al.: Regional electricity sales forecasting research based on big data application service platform. In: *2020 IEEE 3rd International Conference on Electronics and Communication Engineering (ICECE)*, pp. 229–233. IEEE, Xi'an (2020)
15. Song, Y., He, Z., Qian, H., Du, X.: Vision Transformers for Single Image Dehazing. *arXiv 2022*, [arXiv:2204.03883](https://arxiv.org/abs/2204.03883)
16. Sun, C., Myers, A., Vondrick, C., Murphy, K., Schmid, C.: VideoBERT: a joint model for video and language representation learning, *arXiv preprint arXiv:1904.01766* (2019)
17. Radford, A., et al.: Learning transferable visual models from natural language supervision, *arXiv preprint arXiv:2103.00020* (2021)
18. Roy, S.K., Deria, A., Hong, D., Rasti, B., Plaza, A., Chanussot, J.: Multimodal fusion transformer for remote sensing image classification, *arXiv preprint arXiv:2203.16952* (2022)
19. Kingma, D.P., Ba, J.L.: Adam: a method for stochastic optimization. In: *International Conference on Learning Representations*, San Diego, CA, USA, pp. 1–15, May 2015