Short-term Load Forecasting with Transformer Based on Fusion CNN-BiGRU

Tao Xie, Zhongchuang Xiao, Xiaojun He, Yinchun Fan

***Abstract*—To improve the accuracy of short-term power load forecasting, a short-term power load forecasting method based on Transformer with fused CNN-BiGRU is proposed. First, the best input data sequence is selected using the Partial Autocorrelation Function (PACF). Next, the importance scores and rankings of the feature data are obtained through the gradient boosting tree algorithm of CatBoost, and the optimal input features are selected. Then, the feature data and load data are combined. Finally, the combined data is used as input for the Transformer fused with CNN-BiGRU. In model training and forecasting, a hybrid forecasting strategy is employed, incorporating elements of multi-step forecasting into single-step forecasting. For the data of each moment, personalized and independent model training is performed, along with forecasting that include hybrid elements. The model replaces the original word embedding and position encoding components of Transformer. It uses CNN-BiGRU to extract high-dimensional feature representations of latent feature information and relative positional information from the input data. The proposed model demonstrates higher forecasting accuracy through validation on two different datasets and comparison with other forecasting models. Additionally, two ablation experiments are conducted. Through systematic ablation experiments, we demonstrate that modifications to the Transformer input layer significantly improve model performance in time series tasks. These results validate the rationality and effectiveness of the proposed approach. The ablation experiments on the method of PACF selecting the optimal input data sequence and CatBoost filtering the optimal input feature data, as well as the hybrid forecasting strategy, further verify the effectiveness and rationality of the data selection methods and forecasting strategies used in this study for short-term power load forecasting. Moreover, to eliminate the zigzagging jitter phenomenon in the forecast results, Gaussian smoothing is applied to process the forecasting results. The results show that smoothing the forecast results can improve forecast accuracy.**

***Index Terms*—Transformer fuses CNN-BiGRU, PACF, CatBoost, hybrid forecasting strategy**

Manuscript received September 7, 2024; revised March 5, 2025.

Tao Xie is a senior engineer of Chongqing Key Laboratory of Complex Systems and Bionic Control, Chongqing University of Posts and Telecommunications, Chongqing 400065, China (e-mail: xietao@cqupt.edu.cn).

Zhongchuang Xiao is a postgraduate student of Chongqing Key Laboratory of Complex Systems and Bionic Control, Chongqing University of Posts and Telecommunications, Chongqing 400065, China (email: 17772430868@163.com).

Xiaojun He is an undergraduate student majoring in electrical engineering and automation at Chongqing Jiaotong University's School of Electromechanical and Vehicle Engineering, Chongqing 400065, China (email: 13657633901@163.com).

Yinchun Fan is the charging and switching operations manager of SINOPEC Sales Co Ltd's Chongqing EJI branch, Chongqing 400065, China (email: 18223950425@139.com).

I. Introduction

ACCURATE short-term electricity load forecasting is essential for accurate load forecasting for upcoming periods. Precise forecasts enable effective grid scheduling and power generation planning, thereby improving generation efficiency and reducing energy waste. Additionally, they provide critical trading guidance for market entities, including electricity sales companies and users participating in market transactions, helping to improve competitiveness in the electricity market and minimize unnecessary expenses.

With the continuous advancement of artificial intelligence research, short-term power load forecasting algorithms based on artificial intelligence have gradually replaced traditional forecasting methods, and it has become a trend to use artificial intelligence algorithms to conduct further research [1], and representative forecasting methods include Support Vector Regression (SVR) [2-4], fuzzy logic [5-6], Extreme Gradient Boosting (XGBoost) [7-8], wavelet analysis [9-10], deep neural network [11-12], artificial neural network (ANN) [13-15] and other methods. In [16], a fully differentiable cost-oriented loss function, combined with MLR and ANN models, is proposed to minimize the real economic cost caused by forecasting errors. In [17], combined quantile regression with parallel CNNs and BiGRUs for forecasting by dividing the load series into long-term and short-term data. CNNs are used to process the long-term data to identify electricity consumption patterns, while BiGRUs capture short-term consumption behavior. The outputs of CNNs and BiGRUs are then combined and forecasted through a fully connected layer, and the experiments show that the forecasts are reliable. In [18], used ResNet Plus as the overall network framework, applies LSTM layers to residual blocks in ResNet Plus, each residual block contains two LSTM layers and adds DRN-specific shortcuts between the LSTM layers, and the model demonstrates superior forecasting performance compared to both standalone LSTM and ResNet Plus models. Forecasting methods based on artificial neural networks (ANNs) and recurrent neural networks (RNNs) have achieved remarkable success in power load forecasting. However, the Transformer exhibits stronger capabilities in addressing complex temporal features, long-term dependencies, and large-scale data processing.

The Transformer is a network architecture proposed by the Google team that is built entirely on the self-attention mechanism. Unlike traditional recurrent neural network models, it foregoes recurrence and relies entirely on the attention mechanism to extract correlations between sequences. Transformer enables parallel data input and

computation, with its multi-head self-attention mechanism theoretically reducing the propagation path of correlation signals to a minimum of $O(1)$ [19]. When applied to time series forecasting tasks, the Transformer exhibits superior long-term dependency modeling capabilities and faster processing of large-scale time series data compared to RNNs. In [20], combined Transformer and RNN for better learning of temporal and global information in sequences for comprehensive feature extraction. The encoding phase introduces a feature-temporal multi-head attention mechanism that simultaneously considers feature and temporal dimensions, improving the capture of intra-sequence correlations and dependencies. In [21], a hybrid model (CEEMDAN-SE-TR) containing fully integrated empirical modal decomposition (CEEMDAN) with adaptive noise, sample entropy (SE), and Transformer is proposed. Experimental results on New York City load data demonstrate that CEEMDAN-SE-TR achieves the highest forecasting accuracy compared to multiple machine learning and Transformer-based models.

The traditional Transformer model is designed for natural language processing tasks. It effectively captures the long-term dependence of sequences through the self-attention mechanism. However, when applied to time-series data, its input layer structure limits its ability to capture temporal characteristics. In contrast, the recurrent neural network adopts a sequential input structure, which can be a better way to capture the temporal characteristics of the input data [22]. Therefore, this study proposes a short-term power load forecasting model based on the fusion of CNN-BiGRU and Transformer. By replacing the word embedding and position encoding layers of the Transformer input with CNN-BiGRU, the model extracts latent features from the data and implements relative position encoding, thereby maintaining the consistency of content information. In the data preparation phase, the best input data sequence is selected using the Partial Autocorrelation Function (PACF), and the feature data in the dataset is assigned importance scores through the CatBoost gradient boosting tree, thereby selecting the optimal input features. Furthermore, a hybrid forecasting strategy is employed by adding partial historical forecasting results to the input data, introducing multi-step forecasting features into single-step forecasting. This approach overcomes the isolation problem of moment-based forecasting methods, which are unable to capture historical load variations. The main contributions of this paper are summarised below:

(1) In this paper, a model input screening method combining PACF and CatBoost are employed. By using PACF to measure the correlation between current load data and historical load data, the input data sequence can be determined in a more reasonable and efficient manner, thereby avoiding the limitations of relying on empirical methods for data selection. In addition, by applying CatBoost to evaluate the importance of different feature data, the noise perturbing the model in the feature data of low importance can be reduced and the forecasting accuracy of the model can be improved.

(2) This paper proposes a moment-based hybrid forecasting strategy where personalised and independent forecasting model training is performed on the data at each moment. The individual models are then used to forecast the power loads at each moment, providing corresponding load forecasting for each moment.

(3) During the forecasting process, a single-step forecasting method is used, where partial historical forecast results are added to the input data sequence, incorporating features of multi-step forecasting. Unlike traditional single-step or multi-step forecasting methods, this strategy integrates elements of both single-step and multi-step forecasting. It combines the advantages of both approaches while mitigating the forecasting errors caused by their respective limitations.

(4) This paper proposes an version of the traditional Transformer, making it more suitable for time series power load forecasting. By replacing the word embedding and position encoding layers of the Transformer input with CNN-BiGRU, the model effectively extracts latent features from the time series and performs relative position encoding without losing content information. This approach also enables the extraction of contextual information from both the past and future within the data sequence, thereby enhancing the forecasting effectiveness and accuracy of the Transformer model.

The remainder of this paper is organized as follows: Chapter 2 introduces the Partial Autocorrelation Function (PACF), the CatBoost algorithm, and the hybrid forecasting strategy; Chapter 3 presents the proposed forecasting model based on a Transformer network fused with CNN-BiGRU; Chapter 4 evaluates the feasibility and effectiveness of the proposed method through two numerical examples. Finally, Chapter 5 presents the conclusions and discusses directions for future research.

II. Data Selection and Hybrid Forecasting Strategy

This section describes the input sequence selection method based on PACF and the optimal input feature data selection method based on CatBoost used in this study, and explains the forecasting strategy used for load forecasting.

A. Partial Autocorrelation Function (PACF)

The Yule-Walker equations describe the relationship between the parameters and the autocorrelation function in an autoregressive (AR) model. The specific steps for calculating the PACF for each lag order by solving a series of Yule-Walker equations are as follows:

First, the autocorrelation function (ACF) of the time series is calculated. For an ACF with lag k, the formula is as follows:

$$\rho(k) = \frac{\mathrm{cov}(X_k, X_{k-t})}{\sigma_X^2} \tag{1}$$

where, X_t is the value of the time series at the moment t.

Secondly, a system of Yule-Walker equations is constructed, which for an $AR(p)$ model can be expressed as:

$$\rho(k) = \phi_1\rho(k-1) + \phi_2\rho(k-2) + \cdots + \phi_p\rho(k-p) \tag{2}$$

The Yule-Walker system of equations is formulated as

follows:

$$\begin{bmatrix} \rho(0) & \rho(1) & \cdots \rho(p-1) \\ \rho(1) & \rho(0) & \cdots \rho(p-2) \\ \vdots & \vdots & \ddots \quad \vdots \\ \rho(p-1) & \rho(p-2) & \cdots \quad \rho(0) \end{bmatrix} \begin{bmatrix} \phi_1 \\ \phi_2 \\ \vdots \\ \phi_p \end{bmatrix} = \begin{bmatrix} \rho(1) \\ \rho(2) \\ \vdots \\ \rho(p) \end{bmatrix} \tag{3}$$

where, ϕ_i is the parameter of the AR model and $\rho(k)$ is the autocorrelation coefficient at lag order k.

Then, the parameters ϕ_i of the AR model are obtained by solving the system of equations:

$$\Phi = \mathrm{P}^{-1} p \tag{4}$$

where, P is the autocorrelation matrix, p is the autocorrelation vector, and Φ is the AR parameter vector.

Finally, the PACF can be calculated by the following recursive relation:

For $k=1$:

$$\alpha_1 = \phi_1 \tag{5}$$

For $k>1$:

$$\alpha_k = \phi_k - \sum_{j=1}^{k-1} \alpha_j \phi_{k-j} \tag{6}$$

After PACF exceeds a certain lag order, it abruptly drops close to 0, and the value of the coefficients is much smaller than the 95% confidence level [23]. Based on this property of PACF, the optimal sequence of input load data for the forecasting model can be determined, i.e., the maximum lag order of the PACF value not smaller than the 95% confidence level is selected as the number of input load data [24].

B. CatBoost

CatBoost is a decision tree-based gradient boosting algorithm with a built-in capability to determine feature importance during the fitting of a supervised machine learning model. During the training process, CatBoost splits each feature multiple times to identify optimal cut-points that partition the dataset into subsets, thereby minimizing the load forecasting error within each subset. CatBoost records how often each feature is used for splitting and measures the impact of each split on the model's performance. It calculates a feature importance score by evaluating the features' ability to reduce impurities in the data, such as Gini impurity, or other loss function-based metrics during tree construction [25]. Features with higher scores have a more significant impact on load forecasting and can be considered more critical.

In [26], it is stated that for a given feature importance of the feature set $F=\{f_1, f_2, \ldots, f_N\}$, $f_i (i=1,2,\ldots,N)$ is calculated by the following equation:

$$feature_{f_i} = \sum_{trees, leaf S_{f_i}} (v_1 - avr)^2 \cdot c_1 + (v_2 - avr)^2 \cdot c_2 \tag{7}$$

and

$$avr = \frac{v_1 \cdot c_1 + v_2 \cdot c_2}{c_1 + c_2} \tag{8}$$

where, S denotes the different paths to the leaf nodes in the decision tree, c_1 and c_2 denote the total weight coefficients in the left and right leaves, respectively, and c_1 and c_2 denote the formula values in the left and right leaves, respectively.

C. Hybrid Forecasting Strategy

Electricity loads are characterized by stability and strong cyclical regularity, meaning that the load tends to fluctuate minimally and exhibits a consistent pattern at the same moments across different consecutive days. Therefore, the forecasting strategy adopted in this study is as follows: a specialized moment-based forecasting method is used, where a personalized and independent neural network forecasting model is trained for the data at each time point. Each model is then used to forecast the future load values for its corresponding time point, resulting in the forecasted power load for each moment. After the forecasting is complete, all the forecast results are arranged in chronological order to obtain the final load forecasting outcome.

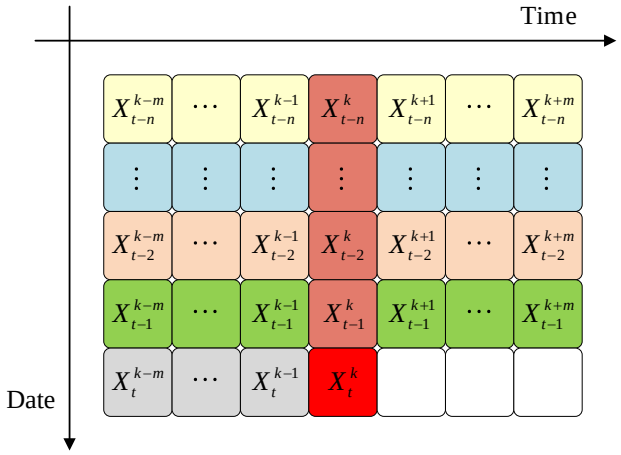
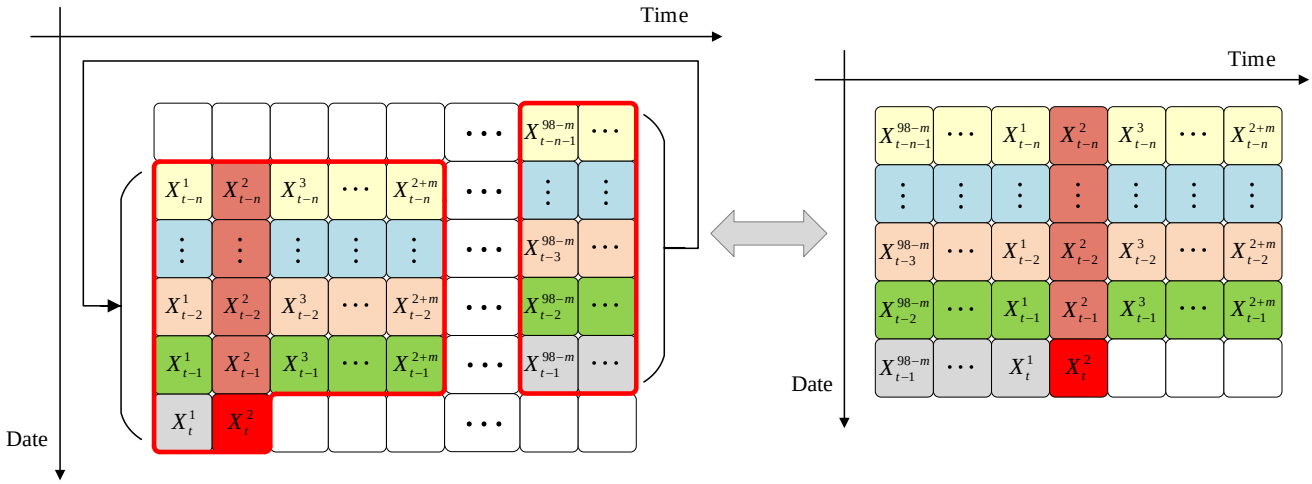
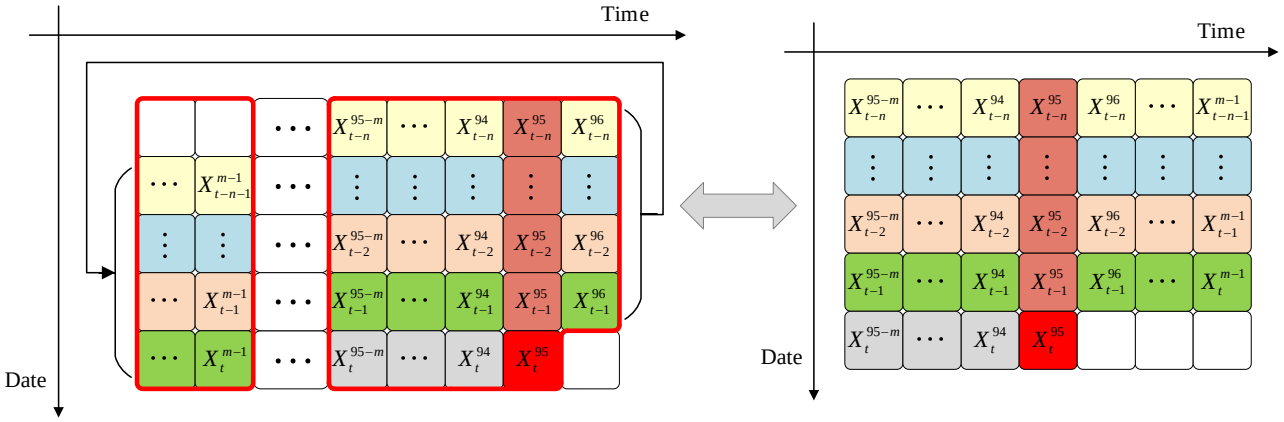


Fig. 1. Selected input data sequences.

Considering the inertia of electricity loads, where loads typically do not change drastically over successive periods and that changes in loads at historical moments can have an impact on loads at subsequent moments. Therefore, this effect must be accounted for during model training and testing. However, due to the adoption of the moment-based forecasting method, when selecting the input load data, it is necessary to include not only the historical daily load data for the current time point but also some historical load data from previous time points on historical days. To capture the load change features from the historical moments that most influence the current time point's load, the load data from some historical moments on the target forecasting date are also included as input load data. If there are no historical moments or an insufficient number of historical moments for the target forecast date, the load data from the last few moments of the previous day is used as a supplement. In this way, the impact of historical load changes on the current time point's load is incorporated into the forecasting model. Historical loads directly affect the current load, and the future load data can also reflect the current load conditions. Therefore, to provide the forecasting model with more comprehensive information on current load variations, some future time-point load data from the historical day are included as input load data. If there are no future moments or not enough future moments for the target forecast date, the load data from the first few moments of the next day is used as a supplement. In this way, the impact of load changes between future moments is introduced into the forecasting model.



(a)Input load data sequence selection method when moments are forward and the number of historical moments used is determined by PACF to be greater than the number of historical moments on the date of the current moment in time



(b)Input load data sequence selection method when moments are backward and the number of future moments adopted as determined by PACF is greater than the number of future moments at the date of the current moment

Fig. 2. Data sequence selection methods when the number of historical or future moments at the current moment is insufficient.

The selection of the input load data sequence is shown in Fig. 1 and Fig. 2, where Fig. 2 shows the load data for one day with a granularity of 96 intervals. In Fig. 1, t is the date at which the load to be forecasted is located; k is the moment at which the load to be forecasted is located; n and m are the optimal number of input data analyzed using the PACF for the current moment and each day, respectively; X is the input data, including the electric loads and various characteristic data, and if X is the load, then X_t^k is the value of the load to be forecasted, and X_t^{k-m} to X_t^{k-1} are the load values for the $k-m$ to $k-1$ moments of the day dated t, and X_{t-n}^k to X_{t-1}^k are the load values for the k moments on day $t-m$ to day $t-1$. Data at other positions follow the same pattern. Fig. 2 (a) illustrates the method for selecting the input load data sequence when the historical moments chosen, as determined by PACF, are greater than the number of historical moments of the forecast target in the target date. Fig. 2 (b) illustrates the method for selecting the input load data sequence when the number of future moments, determined by PACF, exceeds the number of future moments of the forecasting target at the target date.

In forecasting load data for a future day using a moment-based approach, each model is responsible for forecasting only one load value at its respective moment, which is a single-step forecasting strategy. However, forecasts are also made with multiple historical moments as part of the input load data, which incorporating elements of recursive forecasting typical of multi-step forecasting. Therefore, the forecasting strategy proposed in this study combines both single-step and multi-step forecasting elements, making it a hybrid forecasting strategy.

The basic steps in using the hybrid forecasting strategy are as follows:

Step 1: The Partial Autocorrelation Function (PACF) was computed separately for the load data at each time point and on a daily basis. The number of consecutive dates n used for training and forecasting at each moment, as well as the amount of load data used daily, are determined by confidence intervals. The average number of load data utilized daily for training and forecasting is designated as the true quantity of load data selected for each day, denoted as m. This value, m, is subsequently employed to determine the number of historical and future load instances to be selected for training and forecasting on a daily basis.

Step 2: Based on the results of Step 1, the load data for the same moment on the n consecutive days, excluding the target date for which the load needs to be forecast, is selected. At the same time, for each of these n load data points, the load data for m consecutive historical and future moments is selected as the input load data sequence for model training and forecasting, as shown in Fig. 1. If the number of historical moments m' for the current moment is less than

m, the load data from the last $m-m'$ moments of the previous day is selected as a supplement, as shown in Fig. 2 (a). Similarly, if the number of future moments m'' for the current moment is less than m, the load data from the first $m-m''$ moments of the previous day is used as a supplement, as shown in Fig. 2 (b).

Step 3: Based on the load data sequence selected in Step 2, the corresponding feature data for the same date and moment is selected, and combine them such that each row contains one load data point with the corresponding feature data in other columns.

Step 4: A similar moment-based forecasting method is adopted, and data is input into the model using a sliding window approach similar to that shown in Fig. 1 and Fig. 2. For each time point's data, personalized and independent forecasting model training is conducted, and the respective model for each time point is used to forecast the load at that time point. The forecasts for each moment are then arranged in chronological order to obtain the final forecast.

III. Method

This section describes the basic structure of the Transformer of the fused CNN-BiGRU as employed in this study.

A. Convolutional Neural Network (CNN)

The structure of the classical CNN network consists of 2 convolutional layers and 1 pooling operation, and each convolutional layer contains 1 convolutional operation and 1 pooling operation, as shown in Fig. 3. The input data is first processed by the convolutional filter to obtain the feature map, which is then downsampled by the pooling layer to reduce its dimensionality. Finally, the simplified feature map is passed to the fully connected layer for output [27].

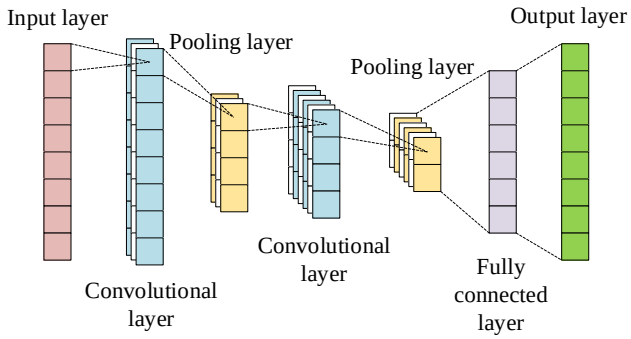


Fig. 3. CNN structure diagram.

The convolution process of CNN is described as follows:

$$y^{l(i,j)} = k^{l(i)} \cdot x^{l(j)} = \sum_{h=0}^{h'} \sum_{w=0}^{w'} k_{(h,w)}^{l(i)} x_{(h,w)}^{l(j)} + b^{l(j)} \tag{9}$$

where, $y^{l(i,j)}$ is the output of the i-th convolution kernel on the j-th feature map in the l-th layer, $k^{l(i)}$ is the i-th convolution kernel in the l-th layer, $x^{l(j)}$ is the j-th feature map in the l-th layer, h' and w' are the height and width of the convolution kernel, respectively, $k_{(h,w)}^{l(i)}$ is the weight of the i-th convolution kernel at position (h,w), $x_{(h,w)}^{l(j)}$ is the value of the position (h,w) of $x^{l(j)}$, and $b^{l(i)}$ is the corresponding deviation of $x^{l(j)}$.

The pooling layer is mainly used to reduce the parameters through downsampling operations and the pooling process is described as follows:

$$A_k^l(i,j) = [\sum_{x=1}^{f} \sum_{y=1}^{f} A(s_0 i + x, s_0 j + y)^p]^{\frac{1}{p}} \tag{10}$$

where, s_0 denotes the step size, f denotes the convolution kernel size, and p denotes the number of filled layers.

B. Bidirectional GRU neural network (BiGRU)

The internal structure of GRU consists of only reset and update gates. GRU utilizes update and reset gates to reduce gradient dispersion and to achieve the ability of long-term memory of sequences and less computational complexity [28]. The structure of the GRU is shown in Fig. 4. The computational procedure of each GRU unit is as follows:

$$r_t = \sigma(W_r x_t + U_r h_{t-1}) \tag{11}$$

$$z_t = \sigma(W_z x_t + U_z h_{h-1}) \tag{12}$$

$$\tilde{h}_t = \tanh(r_t \cdot U h_{t-1} + W x_t) \tag{13}$$

$$h_t = (1 - z_t) \cdot \tilde{h}_t + z_t \cdot h_{t-1} \tag{14}$$

where, r_t is the reset gate, z_t is the update gate, $\tilde{h}_t$ is the candidate hidden layer state reflecting the input information at the moment t and the selective retention of the output h_{t-1} at the moment $t-1$. h_t is the output of the hidden layer at the moment t. σ is the *Sigmoid* function; tanh is the activation function; and W_r, U_r, W_z, U_z, W, and U are all the matrix of training parameters in the network.

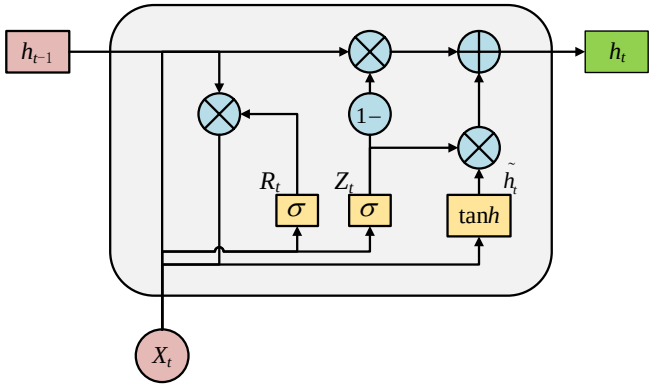


Fig. 4. GRU structure diagram.

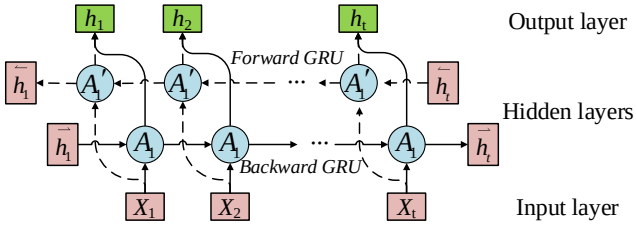


Fig. 5. BiGRU structure diagram.

BiGRU is a bidirectional recurrent network that combines two GRUs with opposite propagation directions, enabling feature extraction from both directions [29]. It can consider the feedback from a given moment to future values in the sequence. Its output layer provides complete historical and future information for each time point in the input data sequence, as shown in Fig. 5. The specific calculation process is as follows:

$$\begin{cases} \vec{h}_t = GRU(X_t, \vec{h}_{t-1}) \\ \overleftarrow{h}_t = GRU(X_t, \overleftarrow{h}_{t-1}) \\ h_t = w_t \vec{h} + v_t \overleftarrow{h}_t + b_t \end{cases} \tag{15}$$

where, $\vec{h}_t$ is the output of the forward hidden layer at the moment t; $\overleftarrow{h}_t$ is the output of the reverse hidden layer at each time at moment t; h_t is the output of the hidden layer at the moment t. w_t and v_t denote the weights corresponding to the forward hidden layer state $\vec{h}_t$ and the reverse hidden state $\overleftarrow{h}_t$ corresponding to the BiGRU at the moment t, respectively; and b_t denotes the bias corresponding to the hidden layer state at the moment t.

C. Transformer

Transformer relies entirely on the attention mechanism to model the global dependencies of inputs and outputs and can avoid the structure of the model with meaningless loops. The overall structure of the Transformer is shown in Fig. 6. From an organizational perspective, the Transformer can be divided into three main parts: the Embedding, the Encoder-Decoder, and logistic regression.

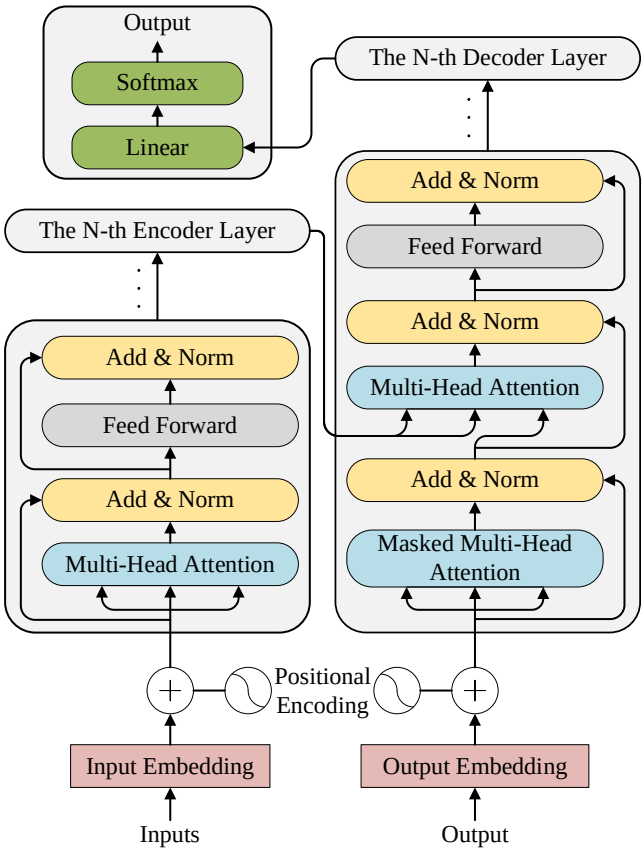


Fig. 6. BiGRU structure diagram.

a. Embedding

The input sequence undergoes a WordEmbedding (WE) operation, mapping each value to a 512-dimensional feature vector. It is then processed with Positional Encoding (PE), where a sine-cosine function encodes the input sequence and generates a fixed representation of the absolute position, which is then pairwise added to the sequence with the completed word embedding [30]. The positional encoding formula is shown below:

$$\begin{aligned} PE(pos, 2i) &= \sin(pos / 10000^{2i/d_{\text{mod}}}) \\ PE(pos, 2i+1) &= \cos(pos / 10000^{2i/d_{\text{mod}}}) \end{aligned} \tag{16}$$

where, pos is the index of the location of the data in the input sequence; d_{mod} is the dimension of the word embedding of the input sequence; and i is the dimension of the vector, which is coded using sinusoidal coding for even positions and cosine coding for odd positions.

b. Encoder-Decoder

The Transformer's encoder consists of a stack of multiple independent encoding layers, each containing a multi-head attention layer and a fully connected layer. The decoder is similarly composed of multiple independent decoding layers, with each decoding layer featuring two multi-attention layers, in contrast to the encoding layer.

The input data, after passing through the embedding layer, generates a data representation matrix. The matrix is passed to the encoder after the attention mechanism processed data is passed to the feed-forward neural network and the result obtained by parallel computation is input to the next encoder [31]. After N encoding operations the encoded information matrix is obtained and passed to the decoder. The decoder then forecasts the next data point y_{i+1} based on the previously forecasted data y_i

The core of the encoder-decoder architecture is the self-attention mechanism. In this mechanism, the inputs—query, key, and value—are all outputs from the previous layer. The received data is multiplied by different weights to produce three matrices: Q, K, and V. The similarity between the data is then calculated using the following formula.

$$Attention(Q, K, V) = soft\max(\frac{QK^T}{\sqrt{d_k}})V \tag{17}$$

where, Q is the query matrix; K is the keyword matrix; V is the value matrix; and T denotes the transpose.

c. logistic regression

The logistic regression part, shown in green in the upper left corner of Fig. 6, consists of a linear transformation with a *Softmax* mapping, which serves to map the output of the decoder to the forecast probability of the electricity load at the next moment.

D. Transformer with CNN-BiGRU fusion

The traditional Transformer does not use a temporal input structure, and the input layer consists of word embedding and positional coding. The sine-cosine absolute positional encoding approach mainly considers local relative features and lacks consideration of global information. The same position encoding is used for the same position in one previous cycle and the same position in the current cycle, making it challenging for the attention layer to capture potential sequence variations. In the load forecasting task, there is an obvious periodic change in the input data sequence, i.e., there is a similar order and regularity of the loads in all the input data sequences. If word embedding and positional coding structures are used, the input layer may lose orientation information after linear transformation and dot product operations [32]. Additionally, the relative positional information between the load and feature data may also be lost. These losses can ultimately affect the accuracy of forecasting.

The GRU inherently possesses a cyclic structure and processes sequential inputs. Its hidden layer retains the sequence's relative positional information, which serves as a trainable relative position encoding. This encoding can reflect global features [33]. The update gate controls the transfer of previous hidden states to the current hidden state. The reset gate determines the number of previous hidden states used to compute output candidates. The BiGRU comprises two GRUs that propagate in opposite directions. In addition to the standard GRU, which processes data in the forward direction, the BiGRU also processes data in the reverse direction, providing the output layer with complete historical and future information for each time point in the sequence. This dual-direction approach offers richer feature representations, effectively addressing the issue of temporal location information loss in the Transformer and providing additional features that enhance forecasting accuracy. In addition, historical load, weather, user behavior, and other influencing factors contain rich information that determines the next moment of power load. This critical information is embedded in the complex correlations of multi-dimensional data. CNN can extract key information hidden in the data through convolutional processing of the input, providing rich input features for the Transformer and enhancing the model's efficiency and forecasting performance.

Therefore, in this study, CNN-BiGRU is used to replace the word embedding and location coding in the input layer of the traditional Transformer. CNN-BiGRU combines the strengths of CNN and BiGRU to process the input time series data to extract high-dimensional features associated with load and other variables. These high-dimensional features, which contain more detailed positional information, are then used as input to the encoding layer. This approach addresses the limitations of the traditional Transformer in handling time series data and enhances the model's overall forecasting performance. The calculation formula is shown below:

$$X = W^i F_{CNN-BiGRU}(x) + b^i \tag{18}$$

where

$$F_{CNN-BiGRU}(x) = FC(BiGRU(CNN(x))) \tag{19}$$

where, x is the input vector consisting of load and feature data; $F_{CNN-BiGRU}$ denotes the extraction of local features by CNN, followed by processing of temporal features in historical and future moment data by BiGRU, and finally feature combination by the fully connected layer; W^i and b^i are the parameter matrix and bias vector of the fully connected layer, respectively; and X is the vector of high-dimensional load and feature data output from the input layer, which is used as the input to the encoder.

The structure of the Transformer load forecasting model fused with CNN-BiGRU proposed in this study is shown in Fig. 7. The model comprises three main components: the input layer, a 6-layer encoder, and the output layer. The load forecasting results are derived through a three-step process of input, encoding, and output.

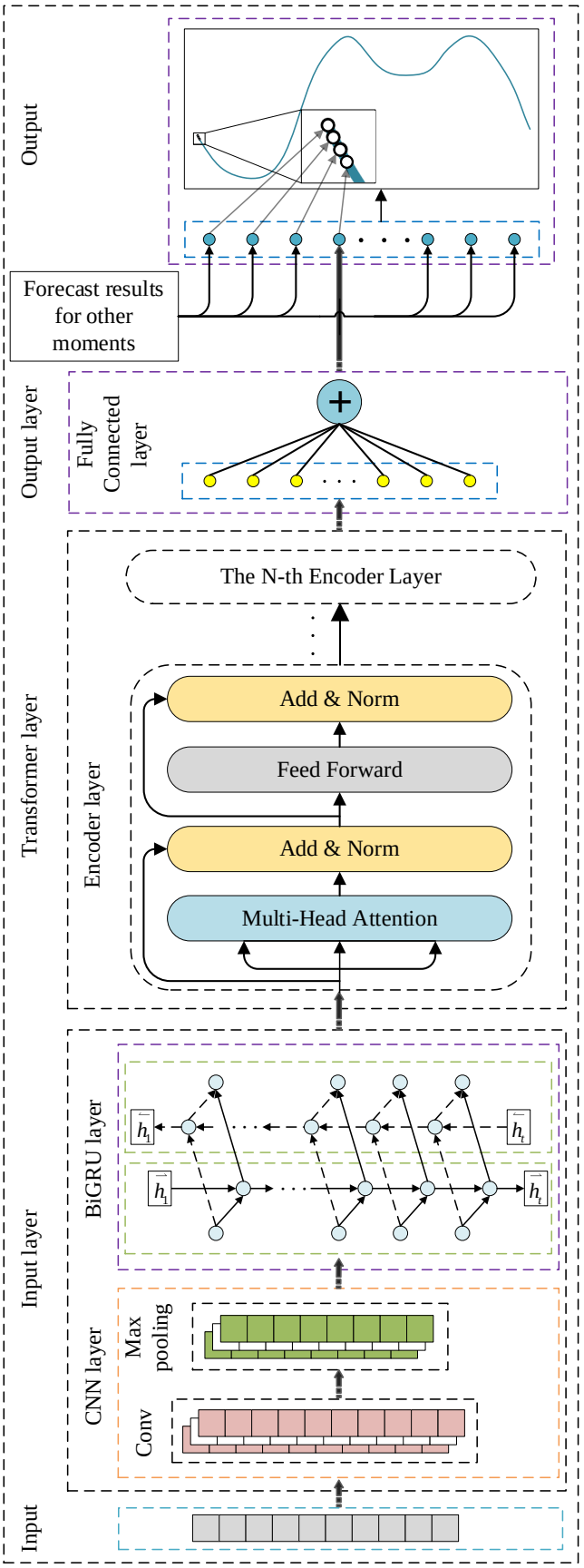


Fig. 7. Transformer structure diagram of fusion CNN-BiGRU.

IV. Case Study

The proposed forecasting method is validated using two datasets: a public dataset and a regional electricity consumption dataset. The public dataset includes historical load data from Panama's national grid spanning February

2015 to May 2020, along with daily temperature, relative humidity, rainfall, and holiday information for Panama City, Santiago, and David City, all recorded at an hourly granularity. The regional dataset consists of real-time electricity load data from a region in southwestern China, accompanied by corresponding temperature, wind speed, humidity, and rainfall data. For this study, daily data recorded at a 15-minute granularity from October 2021 to December 2022 was selected.

A. PACF-based data selection for forecasting sequences

To ensure accurate PACF calculations, the first 80% of the dataset was selected for analysis, and the PACF was computed separately for each momentary load data and daily load data. PACF values outside the 95% confidence interval were considered significant, and the consecutive maximum lag order n in the significant PACF was selected as the number of input load data, which can use $x_{t-1}, x_{t-2}, \ldots, x_{t-n}$ as input load data. The maximum lag order of the significant PACF for daily load data in the dataset varies across days. To ensure consistency in the input data for the neural network, the average of the maximum lag orders for all days is computed and rounded up to determine the number of input load data points per day. For each moment, the respective calculated significant PACF maximum lag order was used as the number of input load data.

B. Features selection based on CatBoost

The regional electricity dataset extracts nine types of time information from the date data: year, month, day of the month, hour, minute, day of the week, season, day of the year, and whether it is a weekday or a day off. These are combined with four types of meteorological data: temperature, rainfall, humidity, and wind speed, resulting in a total of 13 features. Among these, day of the week, season, and whether it is a weekday or a day off are categorical features. Since the input data sequences for the forecasting model consist of numerical data, it is essential to ensure consistency between the feature importance scores obtained from CatBoost and the input data of the forecasting model. Therefore, all three types of categorical features were encoded or transformed before applying CatBoost for feature importance scoring. To avoid feature sparsity after encoding or transforming these categorical features, one-hot encoding is not used in this study. The day of the week and season features, which are periodic and sequential by nature, are processed using label encoding. The 'whether it is a weekday or a day off' feature, which has more categories, is mapped to low-dimensional real vector representations using an Embedding Layer. All features in Panama's public dataset have already been converted to numerical form, so no additional processing is required.

The CatBoost gradient boosting tree model was employed, utilizing the load values as the regression target, to compute the importance scores of all feature data at each time step. Subsequently, the ranking results of feature importance were derived based on these scores.

C. Experimental results and comparative analysis

In this study, the proposed Transformer for fusion CNN-BiGRU is implemented in the PyTorch framework for model construction and training using Python language. The experimental hardware configuration is the 13th Gen Intel(R) Core(TM) i5-13400F CPU with NVIDIA GeForce RTX 3080 Ti graphics card.

Case 1: The public dataset contains loads from the Panamanian national grid, with data collected every hour. To validate the performance of the model used in this study, the dataset is divided into three different training and test set combinations according to the divisions in literature [34] to the literature [36], and the model is trained and forecasted using these three combinations. The three training and test set combinations are divided as shown in Table I.

TABLE I
THREE COMBINATIONS OF TRAINING AND TEST SETS ARE DIVISIONS

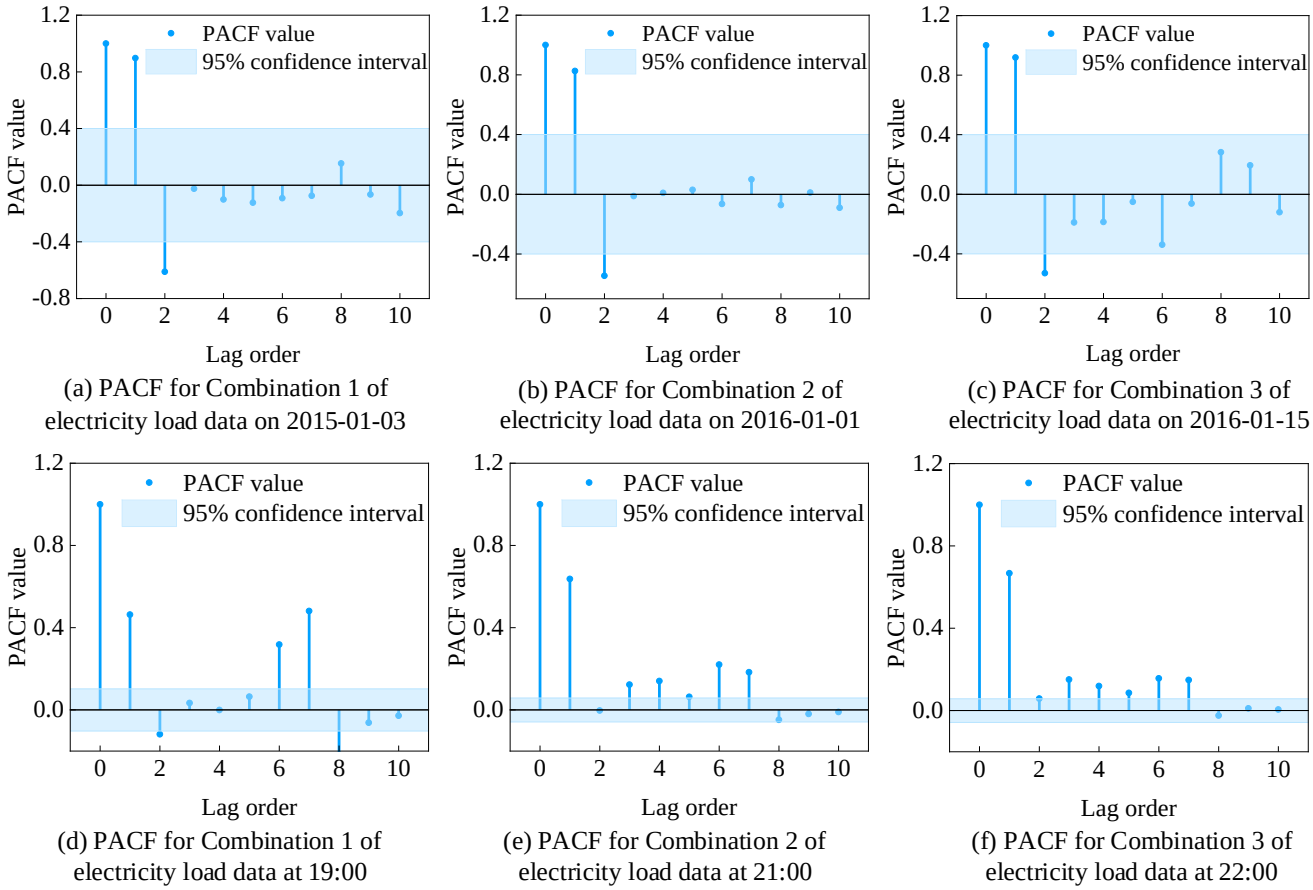
Combinations	Training set	Test set
1	3 January 2015 - 2 January 2016	3 January 2016 - 3 January 2017
2	1 January 2016 - 25 January 2019	26 January 2019 - 31 October 2019
3	1 January 2016 - 29 February 2020	1 March 2020-26 June 2020

TABLE II
THREE COMBINATIONS OF INPUT LOAD DATA SELECTION RESULTS

Combinations	Input Load Data	
	Corresponding Time	Every Day
1	x_{t-1}, x_{t-2}	x_{t-1}, x_{t-2}
2	x_{t-1}	x_{t-1}, x_{t-2}
3	$x_{t-1}, x_{t-2}, \ldots, x_{t-7}$	x_{t-1}, x_{t-2}

The PACFs are calculated for all moments and each day of the training set load data in the 3 combinations, and the optimal input load data sequence for the forecasting model is determined by 95% confidence intervals. Fig. 8 (a), Fig. 8 (b), and Fig. 8 (c) show the relationship between the first 10 lag order PACFs and 95% confidence intervals for some moments in the three combinations. Similarly, Fig. 8 (d), Fig. 8 (e), Fig. 8 (f) show the relationship between the first 10 lag order PACFs and 95% confidence intervals for some dates of the three combinations. Table II shows the results of the input load data selection for the three combinations at the time and for all days corresponding to Fig. 8.

The feature data used by the three combinations are not identical. For each combination, the feature importance of the training set is scored separately using CatBoost, and the features are ranked according to these scores. The feature importance scores and rankings for each combination at the corresponding moments shown in Fig. 8 are presented in Table III.



(a) PACF for Combination 1 of electricity load data on 2015-01-03

(b) PACF for Combination 2 of electricity load data on 2016-01-01

(c) PACF for Combination 3 of electricity load data on 2016-01-15

(d) PACF for Combination 1 of electricity load data at 19:00

(e) PACF for Combination 2 of electricity load data at 21:00

(f) PACF for Combination 3 of electricity load data at 22:00

Fig. 8. Plot of PACF versus 95% confidence intervals for three combinations of training set loading data.

Table III presents the feature names, importance scores, and rankings of the initial features for the three combinations at the corresponding moments. The features are input in descending order of their importance ranking. And, Fig. 9 illustrates the change in the corresponding forecast indicator (MAPE) for each combination as the number of input features varies. As can be seen from Fig. 9, in the beginning stage, the number of input features is small, the features provide limited useful information to the forecasting model, and the forecasting accuracy is low, however, as the number of input features increases, the error decreases. When the number of input features reaches a certain point, the number of input features is large, and the irrelevant noise affects the forecasting accuracy and leads to an increase in the forecasting error. By combining the information from Table III and Fig. 9, the final retained features for each combination are summarized in Table IV. These features are serve as the new feature set for each combination at the corresponding moment. The 24 moments for each combination are then scored for feature importance, and the corresponding best input features are selected.

Based on the above processing, a hybrid forecasting strategy is applied in the training of the forecasting model for each combination, generating forecasting sequences at different moments of time. A Transformer with fused CNN-BiGRU neural network forecasting model is separately constructed for the forecasting sequence of each moment to obtain the forecast values at each moment. These forecasted values are inverse normalized and sorted according to the order of the moments to produce the final load forecasting results for the three combinations.

TABLE III
THE FEATURE DATA NAMES AND CORRESPONDING IMPORTANCE SCORES FOR THE THREE COMBINATIONS

Combinations	Feature Name	Importance Score	Importance Ranking
1	temperature at 2 meters in Tocumen	42.541990	1
	holiday binary indicator	33.555044	2
	liquid precipitation in Tocumen	9.434405	3
	wind speed at 2 meters in Tocumen	7.252478	4
	relative humidity at 2 meters in Tocumen	7.216083	5
2	previous 24 h average demand	61.092865	1
	day of the week	10.191780	2
	previous day same hour demand	8.820358	3
	relative humidity at 2 meters in Tocumen	6.229707	4
	temperature at 2 meters in Tocumen	6.108733	5
	month	4.142111	6
	weekend binary indicator	3.414445	7
	hour of the day	0.000000	8
3	year	23.183588	1
	maximum temperature	15.572068	2
	day of the week	15.017160	3
	average temperature	12.191750	4
	holiday id	9.155069	5
	day of the year	9.063681	6
	minimum temperature	5.603165	7
	day of the month	5.539078	8
	month	4.674441	9

TABLE IV
OPTIMAL INPUT FEATURES OF THE THREE COMBINATIONS AT THE CORRESPONDING MOMENTS

Combinations	Feature Name
1	temperature at 2 meters in Tocumen
2	previous 24 h average demand
3	year minimum temperature day of the year maximum temperature average temperature day of the week month

In order to verify the scientific validity and effectiveness of the Transformer fusing CNN-BiGRU proposed in this study for load forecasting tasks, the mean absolute percentage error (MAPE) and the root mean square error (RMSE) are selected as performance metrics. The MAPE and RMSE are defined as follows:

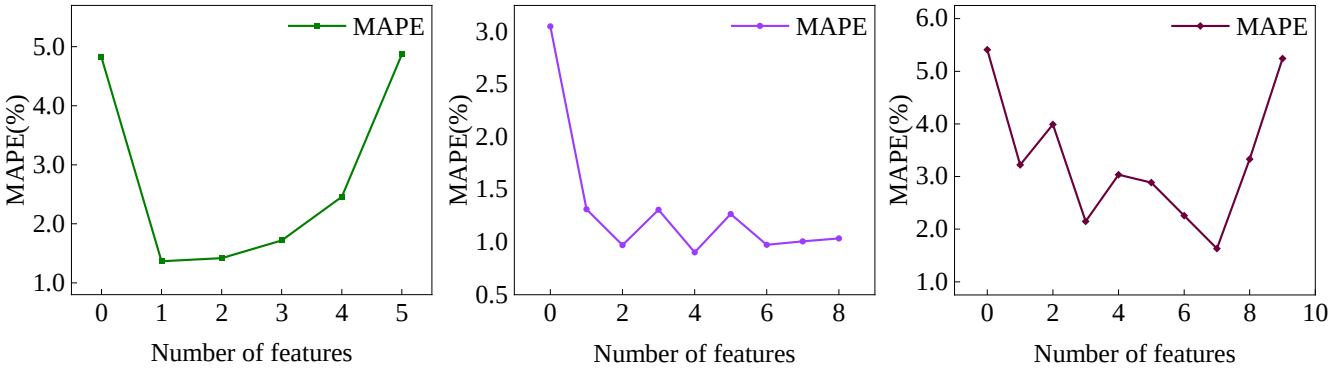
$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\frac{\left|y_i - y_i'\right|}{y_i} \tag{20}$$

$$RMSE = \sqrt{\left[\frac{1}{n}\sum_{i=1}^{n}\left(\left|y_i - y_i'\right|\right)^2\right]} \tag{21}$$

where, y_i is the actual load value and y_i' is the forecasted load value.

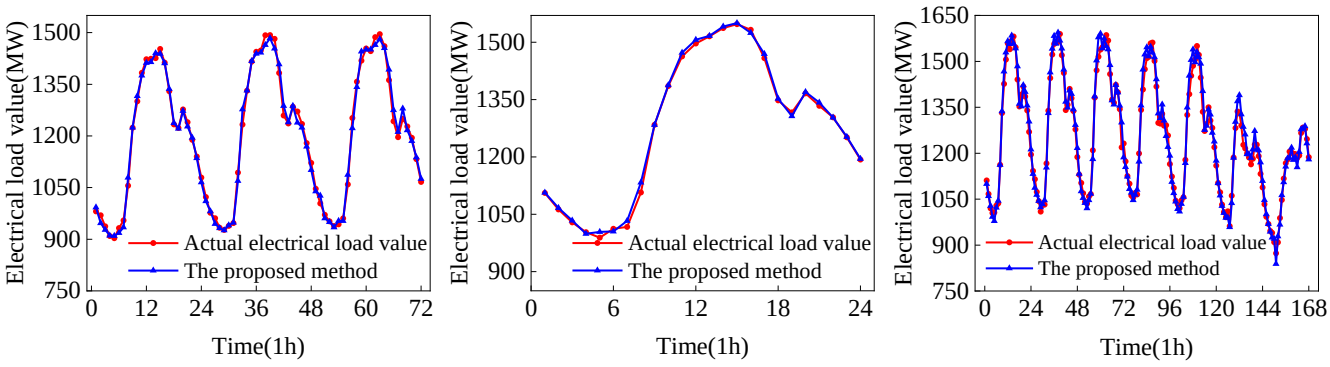
To verify the superiority of the proposed method, the forecasting approach in this study is compared with the methods described in literature [34] to [36], using the Panamanian national electric load dataset. Each method is tested under its respective optimal input data series and optimal input features, and the results from multiple experiments are recorded. The average values of these results are used as the final forecasting outcomes, with the corresponding forecast metrics presented in Table V. Furthermore, Fig. 10 provides a comparison between some of the forecasted load curves generated by the model in this study and the actual load curves under the three data combinations.

As shown in Fig. 10, when using three different data combinations, the forecast curves of the proposed method are all relatively close to the actual load curve. Table V demonstrates that this method outperforms the methods from the literature in terms of forecasting accuracy, particularly with Combination 2 and Combination 3, with MAPE reductions of 1.62% (1.28%, 2.90%) and 0.98% (1.52%, 2.50 %), and RMSE reductions of 27.62 MW (22.72 MW, 50.34 MW) and 10.07 MW (26.44 MW, 36.51 MW), respectively. Compared to methods from the literature using Combination 1, the proposed method achieved a 0.02% (1.38%, 1.40%) lower MAPE and a similar RMSE values (21.1 MW, 20.89 MW). The research results indicate that, based on using PACF to determine the optimal input data sequence and CatBoost to identify the best input features, the approach of optimizing the Transformer with fused CNN-BiGRU and combining it with a hybrid forecasting strategy is effective and feasible for short-term power load forecasting tasks.



(a) MAPE for Combination 1 of electricity load forecasting at 19:00 (b) MAPE for Combination 2 of electricity load forecasting at 21:00 (c) MAPE for Combination 4 of electricity load forecasting at 22:00

Fig. 9. Variation of forecast error with the number of input features at different moments for the three combinations.



(a) Results for Combination 1 of electricity load forecasting (b) Results for Combination 2 of electricity load forecasting (c) Results for Combination 3 of electricity load forecasting

Fig. 10. Forecast curves of this study's method in three combinations of datasets.

TABLE V
COMPARISON OF THE FORECAST INDICATORS OF THIS STUDY'S METHOD WITH THOSE OF THE METHODS IN THE THREE LITERATURES

Combinations	Method	MAPE/%	RMSE/MW
1	CLDNM[34]	1.40	20.89
	The proposed method	1.38	21.1
2	Deep learning[35]	2.90	50.34
	The proposed method	1.28	22.72
3	VBLA[36]	2.50	36.51
	The proposed method	1.48	24.53

By observing the final forecasting results, it can be seen that there is a jittery phenomenon in the forecasting results at local moments. Taking Combination 3 as an example, the local zoomed-in diagram is shown in Fig. 11. In the actual electricity load curve, the load values at several consecutive preceding and succeeding moments transition relatively smoothly, and this jittering phenomenon, characterized by jagged shapes, is generally less frequent in the actual load data. In this regard, in this study, after the model forecasting is completed, the forecasting results are Gaussian smoothed to improve this jitter phenomenon.

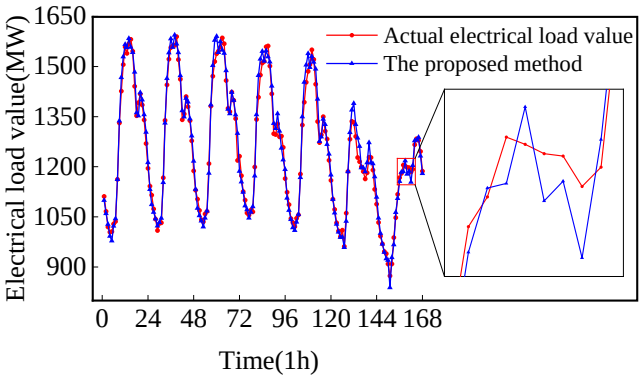


Fig. 11. The jitter phenomenon in the local moments forecasting results of Combination 3.

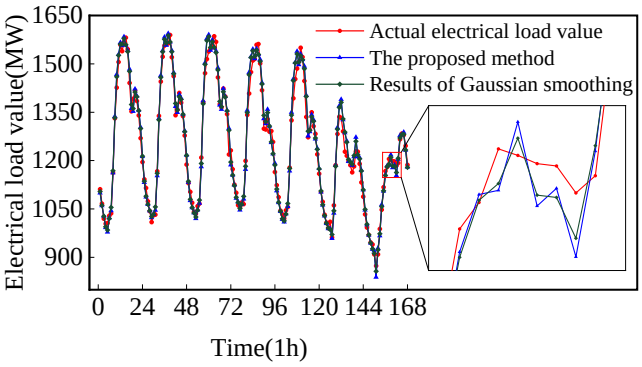
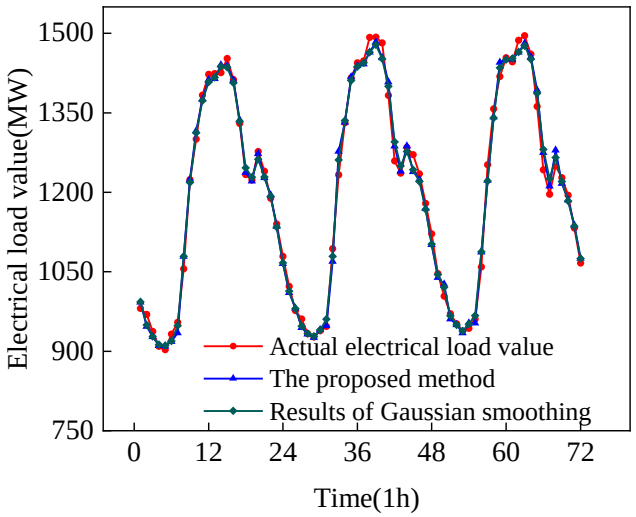


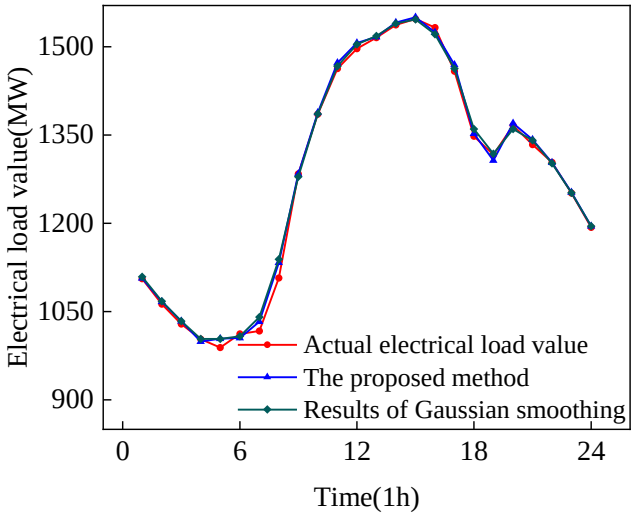
Fig. 12. The results after Gaussian smoothing of the forecasting results in Combination 3.

TABLE VI
COMPARISON OF THE FORECASTING RESULTS OF THE METHOD IN THIS STUDY AFTER GAUSSIAN SMOOTHING WITH THE FORECASTING INDEXES OF THE METHODS IN THREE LITERATURES

Combinations	Method	MAPE/%	RMSE/MW
1	CLDNM[34]	1.40	20.89
	The proposed method	1.31	20.10
2	Deep learning[35]	2.90	50.34
	The proposed method	1.22	21.20
3	VBLA[36]	2.50	36.51
	The proposed method	1.38	23.85



(a) Electricity load forecasts for combination 1 after Gaussian smoothing

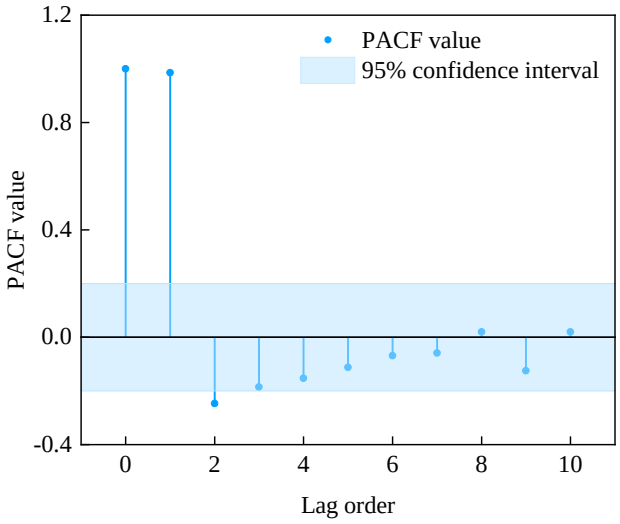


(b) Electricity load forecasts for combination 2 after Gaussian smoothing

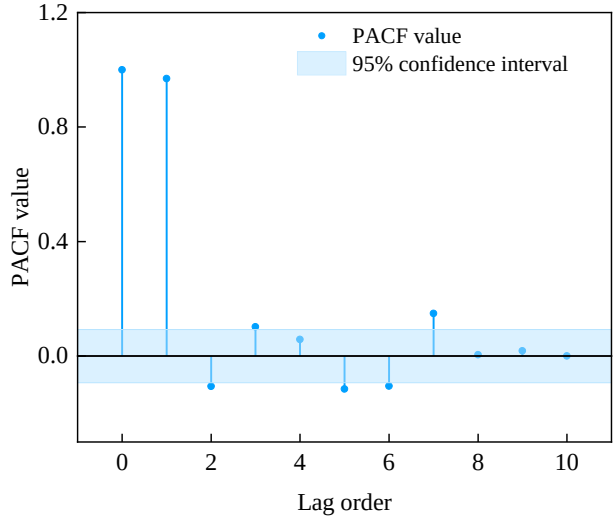
Fig. 13. Gaussian smoothing of the forecasts for combinations 2 and 3.

Gaussian smoothing is performed on the forecasting results of this study's method on combination 3, as shown in Fig. 12. By zooming in on the details, it can be observed that the jagged jitter present in the initial forecasting result is well suppressed. The forecasting curve becomes smoother and aligns more closely with the trend of the actual load curve. As shown in the forecasting indexes of Combination 3 in Table VI, the accuracy of the forecasting data improves after smoothing. Specifically, the MAPE decreased by 0.1% (1.38%, 1.48%) compared to the pre-smoothing result, and by 1.12% (1.38%, 2.50%) compared to the literature's method in Combination 3. Additionally, the RMSE decreased by 0.68 MW (23.85 MW, 24.53 MW) compared to the pre-smoothing result, and by 12.66 MW (23.85 MW, 36.51 MW) compared to the literature's method in Combination 3. This demonstrates that it is effective and reasonable to improve the sawtooth jitter phenomenon by Gaussian smoothing the forecasting results after the forecasting is completed. The forecasting results of this study's method on combination 1 and combination 2 after Gaussian smoothing are shown in Fig. 13, and the forecasting metrics are shown in Table VI. By comparing

Table V and Table VI, it can be observed that the smoothed metrics are all decreased to a certain extent compared to the pre-smoothing results. In Combination 1, the MAPE decreased by 0.07% (1.31%, 1.38%), and the RMSE decreased by 1 MW (20.10 MW, 21.10 MW) compared to the pre-smoothing period. Similarly, in Combination 2, the MAPE decreased by 0.06% (1.28%, 1.22%), and the RMSE decreased by 1.52 MW (21.20 MW, 22.72 MW) compared to the pre-smoothing results. These results indicate that the forecasting accuracy improved after smoothing. Compared to the method in the literature of Combination 1, before applying Gaussian smoothing to the forecasting results, the MAPE of this method showed a slight decrease, but the RMSE increased by 0.21 MW. However, after smoothing, both MAPE and RMSE are lower than those of the method in the literature for Combination 1. The MAPE decreased by 0.09% (1.40%, 1.31%), and RMSE decreased by 0.79 MW (20.89 MW, 20.10 MW).



(a) PACF for electricity loads data at 2021-10-14



(b) PACF for electricity loads data at 1:30

Fig. 14. The PACF of regional electricity load data and its relationship with the 95% confidence interval.

Case 2: In the regional electricity consumption dataset, the active load data of the area are collected every 15 minutes as the forecasting target. The experiment aims to forecast the electrical load at 96 moments on December 31, 2022, using the load data from October 14, 2021, to December 15, 2022, as the training set.

Fig. 14 (a) illustrates the relationship between the PACF lag orders and the 95% confidence intervals for the regional electricity consumption dataset at the 1:30 moment. Fig. 14 (b) shows the same relationship for the data from October 14, 2021. Table VII presents the results of selecting the input load data at the 1:30 moment and for each day.

TABLE VII
THE RESULT OF SELECTING THE INPUT LOAD DATA OF THE REGIONAL ELECTRICITY CONSUMPTION DATASET

Time	Input Load Data
1: 30	$x_{t-1}, x_{t-2}, x_{t-3}, x_{t-4}$
Every Day	$x_{t-1}, x_{t-2}, x_{t-3}$

TABLE VIII
THE FEATURE NAMES AND IMPORTANCE SCORES OF THE REGIONAL ELECTRICITY LOAD DATASET

Feature Name	Importance Score	Importance Ranking
day of the year	47.950278	1
month	37.114068	2
day of the month	7.432280	3
season	4.487404	4
day of the week	1.012315	5
year	1.009334	6
whether it is a weekday or a day off	0.994320	7
wind speed	0.000000	8
humidity	0.000000	9
rainfall	0.000000	10
temperature	0.000000	11
minute	0.000000	12
hour	0.000000	13

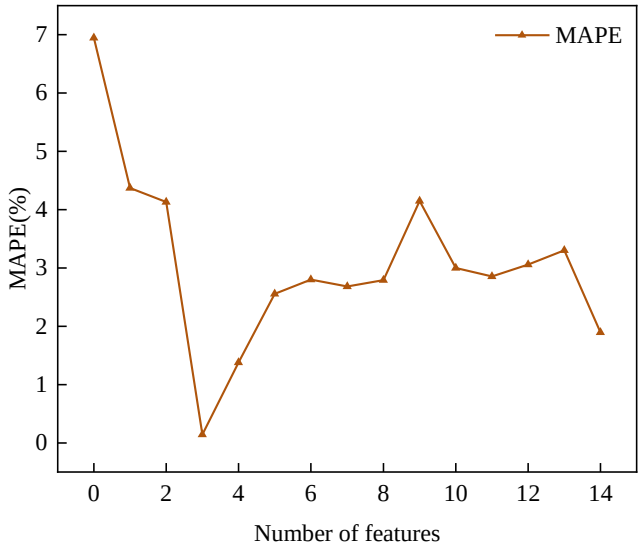


Fig. 15. Variation of forecast error with the number of input features at moment 1:30.

The regional electricity dataset includes 13 initial features, with the feature importance scores and rankings at 1:30 shown in Table VIII. As shown in Table VIII, the six features — wind speed, humidity, rainfall, temperature, minute, and hour—have a score of 0, which has an almost negligible impact on load forecasting.

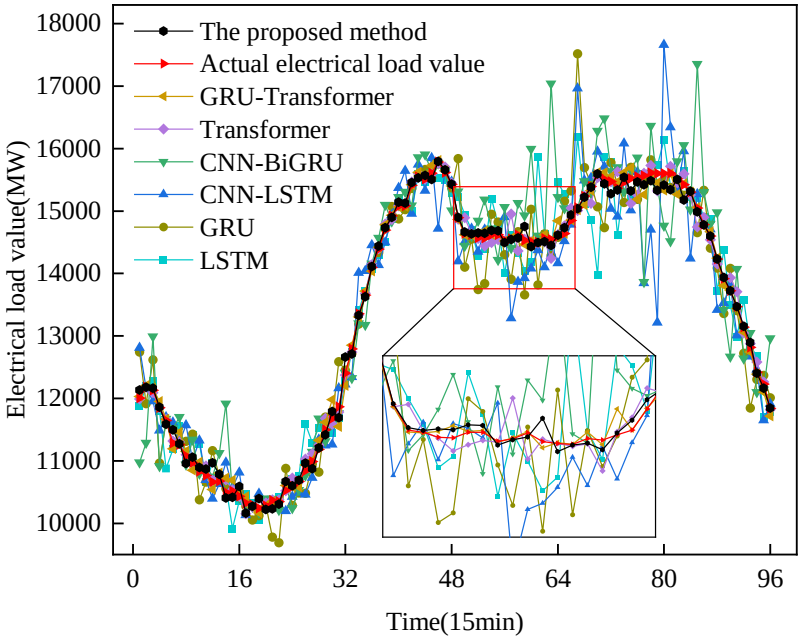
The features at the 1:30 moment are input into the model in order of importance ranking from high to low, and the variation of MAPE with the number of input features is shown in Fig. 15. From Fig. 15, it can be observed that when

the number of input features is three, the MAPE value is the smallest, indicating the highest accuracy. Combining Table VIII and Fig. 15, the final features retained at the moment of 1:30 are day of year, month, and day of month, totaling three features as the new feature set.

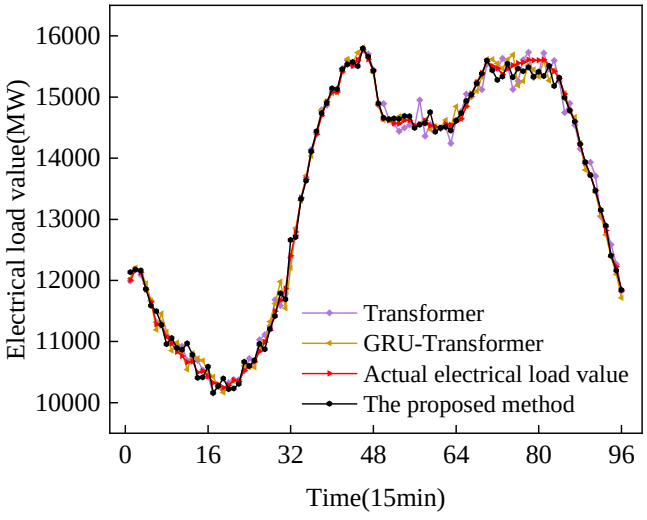
LSTM, GRU, CNN-LSTM, CNN-BiGRU, Transformer, and GRU-Transformer are selected as the comparison models. These models are trained and tested using a hybrid forecasting strategy under their respective optimal input data sequences and optimal input features. The forecasting metrics of each model, as well as those after Gaussian smoothing, are presented in Table IX. Fig. 16 presents a comparison of the forecast load curves from six contrast models and the model proposed in this study with the actual load curve. To mitigate the jaggedness in the forecast curves, Gaussian smoothing was applied to the forecast results. The smoothed curves are illustrated in Fig. 17. Fig. 16 and Fig. 17 are presented in the form of a 'general graph' and 'subgraphs', where the two 'subgraphs' represent the forecasting results of the recurrent neural network category models and the Transformer category models, respectively.

TABLE IX
COMPARISON OF THE FORECASTING METRICS OF EACH MODEL AND THE FORECASTING METRICS AFTER GAUSSIAN SMOOTHING

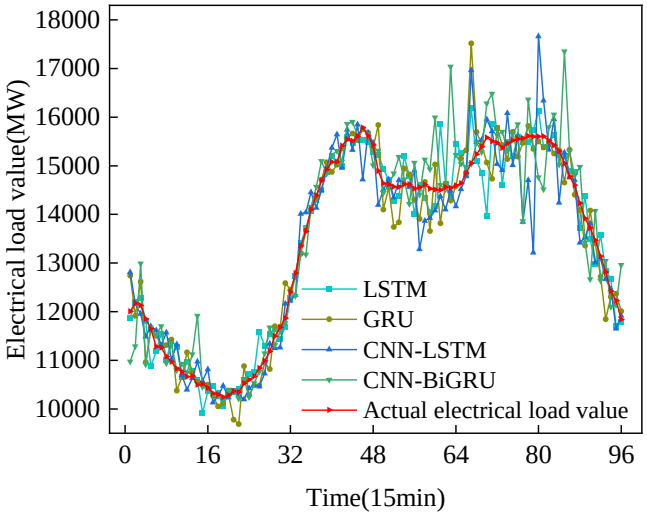
Method	Whether Gaussian smoothing	MAPE/%	RMSE/MW
LSTM	No	1.948	385.124
	Yes	1.038	189.329
GRU	No	2.251	450.665
	Yes	1.232	222.701
CNN-LSTM	No	2.717	573.928
	Yes	1.412	264.100
CNN-BiGRU	No	2.981	611.201
	Yes	1.857	333.356
Transformer	No	0.647	128.647
	Yes	0.613	101.807
GRU-Transformer	No	0.621	113.506
	Yes	0.609	98.656
The proposed method	No	0.584	103.714
	Yes	0.549	96.431



(a) Forecasting results of all models on regional electricity loads

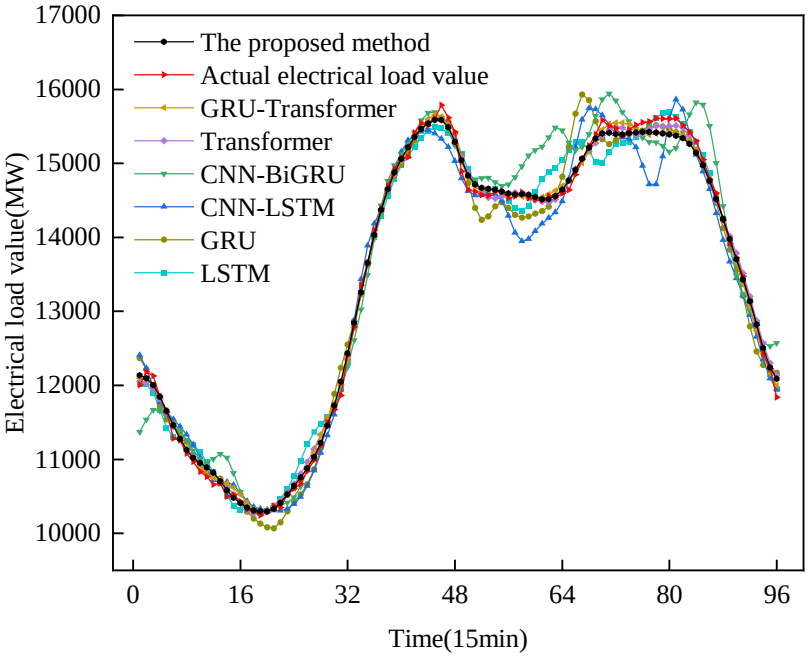


(b) Forecasting results of Transformer category models in regional electricity loads

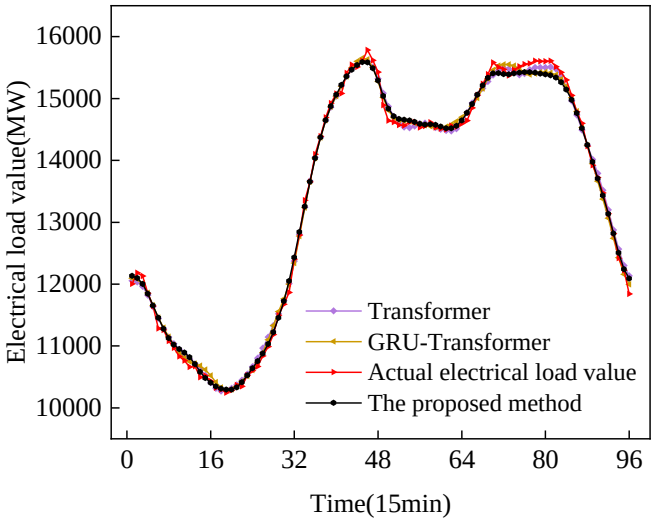


(c) Forecasting results of recurrent neural network-like models in regional electricity loads

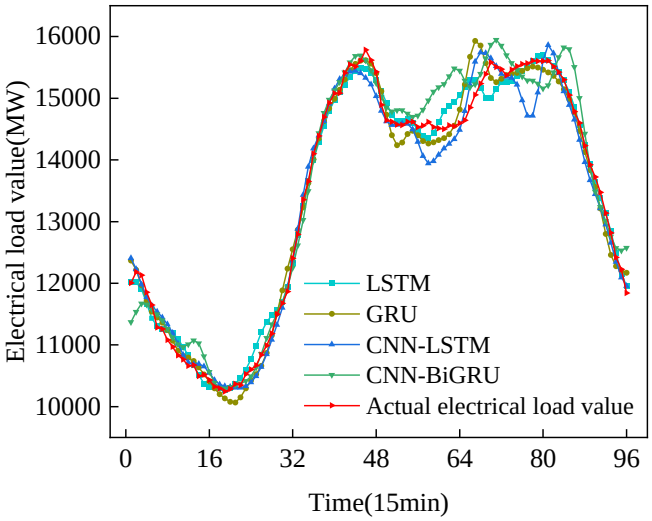
Fig. 16. Forecasting curves for each model.



(a) Forecasting results of Gaussian smoothing for all models in tegional electricity



(b) Forecasting results of Gaussian smoothing for Transformer category models in regional electricity



(c) Forecasting results of Gaussian smoothing for recurrent neural network category models in regional electricity

Fig. 17. Gaussian smoothing results of the forecasting curves for each model.

As shown in Fig. 16 (a), the forecasting curve of the Transformer, with the input layer replaced by the GRU, is closer to the actual load curve than the traditional Transformer. This is particularly evident in the middle segment, indicating that replacing the input layer of the Transformer with a model capable of encoding relative position information can effectively avoid the loss of such information. This adaptation better suits the time series task and enhances the forecasting accuracy of the Transformer. From Fig. 16 (b), it can be observed that the forecasting curve of the Transformer, which uses CNN-BiGRU to replace the input layer, is the closest to the actual load curve. This indicates that the use of CNN with multi-feature inputs provides the Transformer with more potential features, which are beneficial for model learning. Additionally, by utilizing the past and future information provided by BiGRU, the Transformer can learn richer feature representations, thereby further improving the forecasting accuracy of the model. From Fig. 16 (c), it can be seen that the recurrent neural network model exhibits more jagged jitter in the middle and later parts of the curve. Although the general trend follows the actual load curve, the forecasting accuracy is not sufficiently high, and the forecasting metrics in Table IX also show that the model of recurrent neural network class does not perform well. In contrast, the fusion of CNN-BiGRU of the Transformer proposed in this study, demonstrates better forecasting accuracy than the other six methods. Compared to the other six methods, the MAPE decreased by 1.364% (0.584%, 1.948%), 1.667% (0.584%, 2.251%), 2.133% (0.584%, 2.717%), 2.397% (0.584%, 2.981%), 0.063% (0.584%, 0.647%), and 0.037% (0.584%, 0.621%), while the RMSE decreased by 281.41 MW (103.714 MW, 385.124 MW), 346.951 MW (103.714 MW, 450.665 MW), 470.214 MW (103.714 MW, 573.928 MW),

507.487 MW (103.714 MW, 611.201 MW), 24.933 MW (103.714 MW, 128.647 MW), and 9.792 MW (103.714 MW, 113.506 MW).

Comparing the corresponding forecasting curves in Fig. 16 and Fig. 17, it can be seen that the amplitude and extent of the jagged portions of the forecasting curves are significantly suppressed after smoothing. Additionally, the smoothed forecasting curves achieve a smooth transition in regions where the jitter amplitude was previously too large. As shown by the forecasting indices in Table IX, the smoothing process positively impacts the improvement of forecasting accuracy. This effect is particularly notable in the recurrent neural network class of the model. Especially in the forecasting results of the recurrent neural network category models, the jagged jitter accounts for a higher proportion and exhibits larger amplitudes, making the improvement in forecasting accuracy more pronounced. As shown in Fig. 17 (b), the jagged jitter phenomenon in the forecasting of the Transformer category's model is relatively less frequent, and the jitter amplitude is small. However, Gaussian smoothing will make the load curve smoother, which can also play a role in improving the forecasting accuracy of the Transformer category's model. From the forecasting metrics in Table IX, it is known that the MAPE of LSTM, GRU, CNN-LSTM, and CNN-BiGRU decreased by 0.91% (1.038%, 1.948%), 1.019% (2.251%, 1.232%), 1.305% (2.717%, 1.412%), 1.124% (2.981%, 1.857%), respectively. Meanwhile, the RMSE decreased by 195.795 MW (385.124 MW, 189.329 MW), 227.964 MW (450.665 MW, 222.701 MW), 309.828 MW (573.928 MW, 264.100 MW), 277.845 MW (611.201 MW, 333.356 MW); the MAPE of Transformer, GRU-Transformer, and the methods used in this study decreased by 0.034% (0.647%, 0.613%), 0.012% (0.621%, 0.609%), and 0.035% (0.584%, 0.549%), respectively, and RMSE was decreased by 26.84 MW (128.647 MW, 101.807 MW), 14.85 MW (113.506 MW, 98.656 MW), 7.283 MW (103.714 MW, 96.431 MW).

An ablation study was conducted to illustrate how each modification to the Transformer input layer improves forecasting accuracy in time series tasks. By progressively modifying the Transformer input layer, the layers were sequentially changed to CNN, BiGRU, and CNN-BiGRU, and the forecasting results were compared with those of the original, unmodified Transformer. Fig. 18 and Fig. 19 illustrate the forecasting results of the four models on the regional electricity consumption dataset, along with their results after Gaussian smoothing. Table X compares the performance metrics of the four models. Meanwhile, to verify the practical significance of the data selection methods—PACF for selecting the best input data sequences and the CatBoost model for selecting the best input features—as well as the hybrid forecasting strategy used in this study, we divided the methods into two parts. The first part includes the PACF and CatBoost model for data selection, while the second part is the hybrid forecasting strategy. Based on whether these two parts were used in forecasting, we divided the experiments into four groups for ablation studies. The forecasting model used in these experiments was the Transformer fused with CNN-BiGRU, as proposed in this study. The forecasting results are shown in Fig. 21 and Table XI.

From the forecasting curves in Fig. 18 and Fig. 19, it can be observed that the overall fit between the model's forecasting curve and the actual power load curve improves as the structure of the Transformer input layer is progressively modified. When the input structure proposed in this study is adopted, the overall fit reaches its highest level. By zooming in on certain parts of the curves in Fig. 18 and Fig. 19, it is evident that the model with the CNN-BiGRU replaced as the input layer outperforms the other three models in forecasting the power load during two instances of rapid load variation. The forecasting metrics in Table X also show that the modifications to the Transformer input layer effectively improve its accuracy in power load forecasting. With the continuous modifications to the Transformer input layer, the model's metrics slightly decrease. The original Transformer with its input structure has a MAPE of 0.647% and an RMSE of 128.647 MW. Compared to this, the MAPE of the Transformer with the BiGRU input layer was reduced by 0.011% (0.636%, 0.647%), the MAPE of the Transformer with the CNN input layer was reduced by 0.032% (0.615%, 0.647%), and the RMSE of the CNN model decreased by 11.339 MW (117.308 MW, 128.647 MW). This demonstrates that modifying the Transformer input layer structure can indeed enhance forecasting accuracy for power load forecasting. The Transformer with the CNN-BiGRU input layer achieved the lowest MAPE and RMSE, with MAPE reduced by 0.063% (0.584%, 0.647%) and RMSE reduced by 24.933 MW (103.714 MW, 128.647 MW). After Gaussian smoothing of the forecasting results, the metrics for all four models showed a decline. The MAPE and RMSE for the original Transformer input structure decreased by 0.034% and 26.84 MW, respectively. For the Transformer with the BiGRU input layer, the MAPE and RMSE decreased by 0.073% and 25.68 MW. For the Transformer with the CNN input layer, the MAPE and RMSE decreased by 0.035% and 20.357 MW, respectively. And the Transformer with the CNN-BiGRU input layer maintained the lowest metrics among the four models, with MAPE and RMSE decreasing by 0.06% and 7.283 MW. Furthermore, by comparing the model metrics after Gaussian smoothing, it can be observed that there remains a trend of declining performance metrics as the input layer of the Transformer undergoes continuous modifications. The original Transformer's MAPE and RMSE are 0.613% and 101.807 MW. Compared to the original structure, the Transformer with the BiGRU input layer saw a MAPE decrease of 0.05% (0.563%, 0.613%), the Transformer with the CNN input layer saw a MAPE decrease of 0.033% (0.580%, 0.613%), and the RMSE for the CNN model decreased by 4.856 MW (96.951 MW, 101.807 MW). The Transformer with the CNN-BiGRU input layer achieved the lowest MAPE and RMSE, with MAPE reduced by 0.064% (0.549%, 0.613%) and RMSE reduced by 5.376 MW (96.431 MW, 101.807 MW). In conclusion, whether examining the forecasting results before or after Gaussian smoothing, it is evident that there is a consistent trend of declining model performance metrics as the input layer of the Transformer undergoes successive modifications. The modification method employed in this study exhibits the lowest performance metrics, indicating that the proposed alterations to the Transformer's input layer

are effective. Replacing the Transformer's input layer with a CNN-BiGRU architecture enhances the forecasting accuracy in power load forecasting tasks.

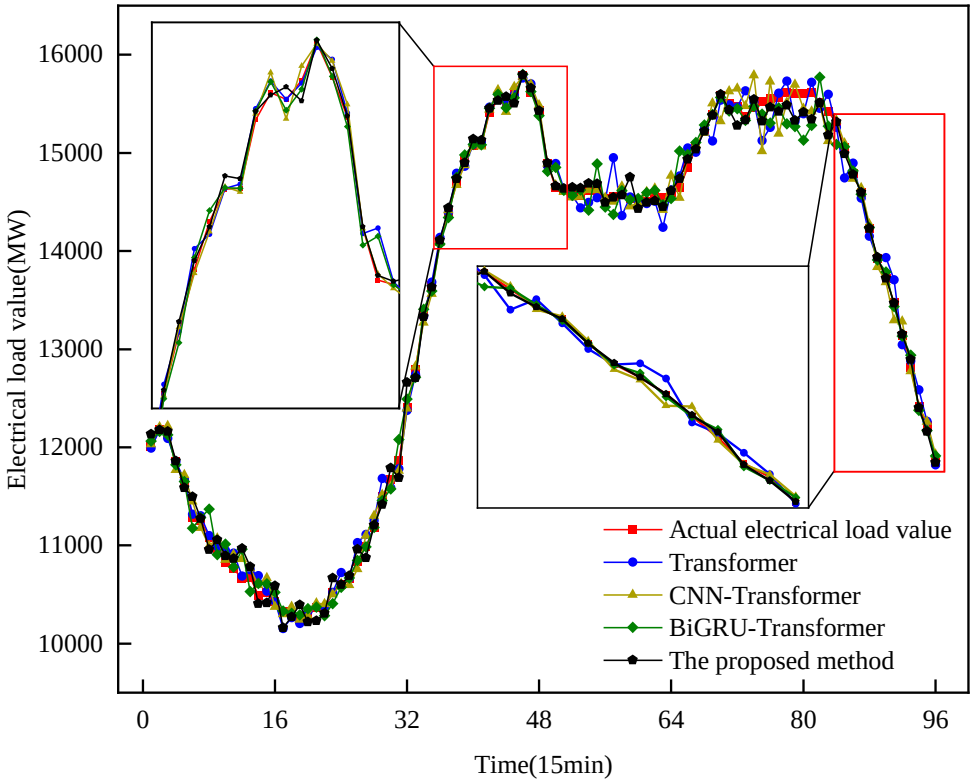


Fig. 18. Transformer forecast curves after gradual modification of the input layer.

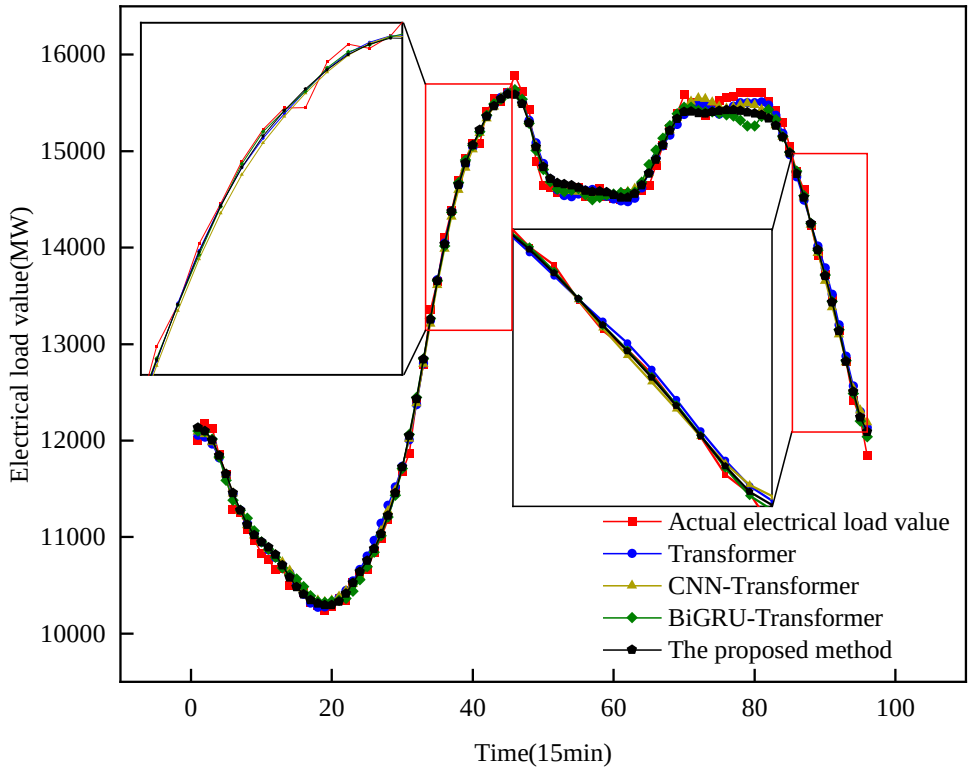


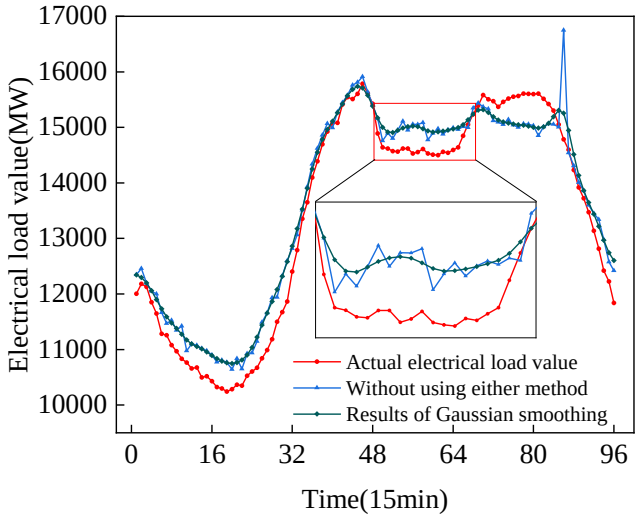
Fig. 19. Gaussian smoothing of Transformer's forecasts after gradual modification of the input layer.

TABLE X
COMPARISON OF THE FORECASTING METRICS OF THE TRANSFORMER WITH PROGRESSIVELY MODIFIED INPUT LAYERS AND THE FORECASTING METRICS AFTER GAUSSIAN SMOOTHING

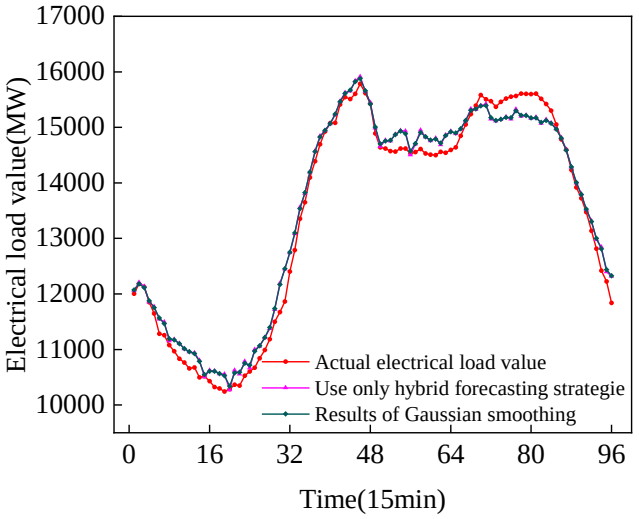
Whether to add CNN	Whether to add BiGRU	Whether Gaussian smoothing	MAPE/%	RMSE/MW
Yes	Yes	No	0.584	103.714
		Yes	0.549	96.431
Yes	No	No	0.615	117.308
		Yes	0.580	96.951
No	Yes	No	0.636	129.521
		Yes	0.563	103.841
No	No	No	0.647	128.647
		Yes	0.613	101.807

TABLE XI
COMPARISON OF THE FORECASTING METRICS FOR EACH MODEL OF THE ABLATION EXPERIMENT AND THE FORECASTING METRICS AFTER GAUSSIAN SMOOTHING

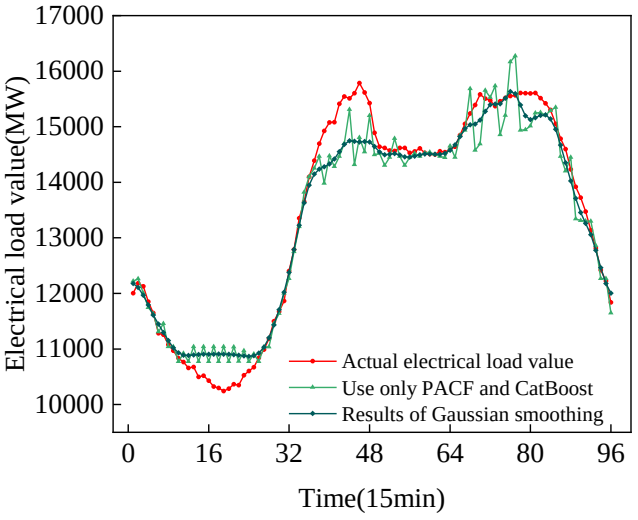
Groups	Whether to use PACF and CatBoost	Whether to use hybrid forecasting strategy	Whether Gaussian smoothing	MAPE/%	RMSE/MW
A	Yes	Yes	No	0.584	103.714
			Yes	0.549	96.431
B	Yes	No	No	2.270	421.567
			Yes	1.842	351.018
C	No	Yes	No	1.557	242.906
			Yes	1.557	239.806
D	No	No	No	2.769	432.282
			Yes	2.751	393.060



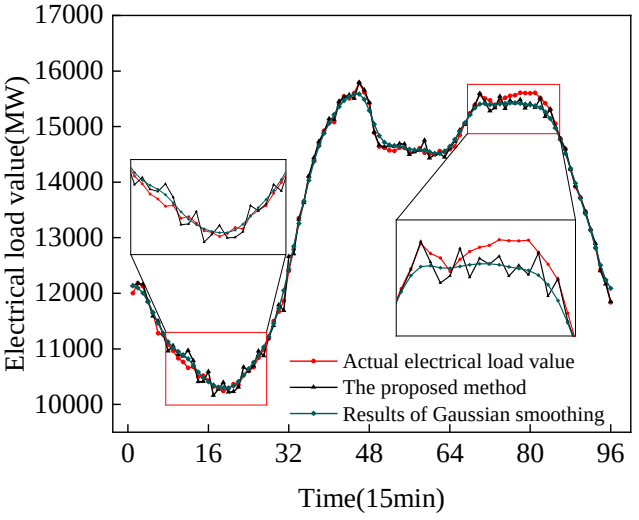
(a) Forecasting results for Group A using neither PACF nor CatBoost nor hybrid forecasting strategy and its Gaussian smoothing results



(b) Forecasting results for Group B using only the hybrid forecasting strategy and its Gaussian smoothing results



(c) Forecasting results for Group C using only PACF and CatBoost and its Gaussian smoothing results



(d) Forecasting results for Group D using both PACF and CatBoost as well as a hybrid forecasting strategy and its Gaussian smoothing results

Fig. 20. Forecasting curves for each model of the ablation experiment and their Gaussian smoothed curves.

As can be seen from the comparison of the forecasting metrics of group A with group C, and group B with group D in Table XI, it can be observed that the model forecasting errors are lower when the input data sequence is selected using PACF and the best input features are selected using CatBoost, compared to the models that do not use PACF and CatBoost for data selection. The MAPE decreased by 0.973% (0.584%, 1.557%) and the RMSE decreased by 139.192 MW (103.714 MW, 242.906 MW) for group A compared to group C, and the MAPE decreased by 0.499% (2.270%, 2.769%) for group B compared to group D, and RMSE decreased by 10.715 MW (421.567 MW, 432.282 MW). Fig. 20 (a)-Fig. 20 (d) shows that the forecasting curves of group A are closer to the actual load curves than those of group C, and the forecasting curves of group B are closer to the actual load curves than those of group D. This suggests that selecting the input load sequence and input features, while eliminating redundant and irrelevant features from the dataset, can effectively improve the accuracy of load forecasting. Additionally, it reduces the impact of irrelevant noise on forecasting accuracy. Furthermore, from the comparison of forecasting metrics between group A and group B, as well as group C and group D in Table XI, it is evident that the model forecasting errors are lower when the hybrid forecasting strategy is used. This is in contrast to the models that do not employ the hybrid forecasting strategy. Specifically, for group A compared to group B, the MAPE decreased by 1.686% (0.584%, 2.270%) and the RMSE decreased by 317.853 MW (103.714 MW, 421.567 MW) for group C compared to group D. The MAPE decreased by 1.212% (1.557%, 2.769%) and the RMSE decreased by 189.376 MW (242.906 MW, 432.282 MW). From the forecasting results in Fig. 20 (a)-Fig. 20 (d), it can be seen that the forecasting curves are all closer to the actual load curves after using the hybrid forecasting strategy than when it is not used, and there are fewer jagged jitters and smaller jitter amplitudes. By examining the enlarged portion of Fig. 20 (d), it can be seen that when the forecasting of historical moments is not satisfactory, there are some moments where the forecasting results can be closer to the actual load. It is because the hybrid forecasting strategy is that personalized and independent model training is performed for each moment of data. During forecasting, only a part of the forecasting data of the historical moments is used as the input data sequence to forecast the current moment's electricity load, and the forecasting data of the historical moments is not a large proportion of the input data sequence. This not only reduces the interference of distant historical load data characteristics on the current moment forecasting, but also avoids the isolation of the results of the moment-based single-step forecasting method at the time of forecasting. Additionally, since the models used at different time points are distinct, errors can be corrected during the forecasting process, thereby reducing error propagation and enhancing forecasting accuracy.

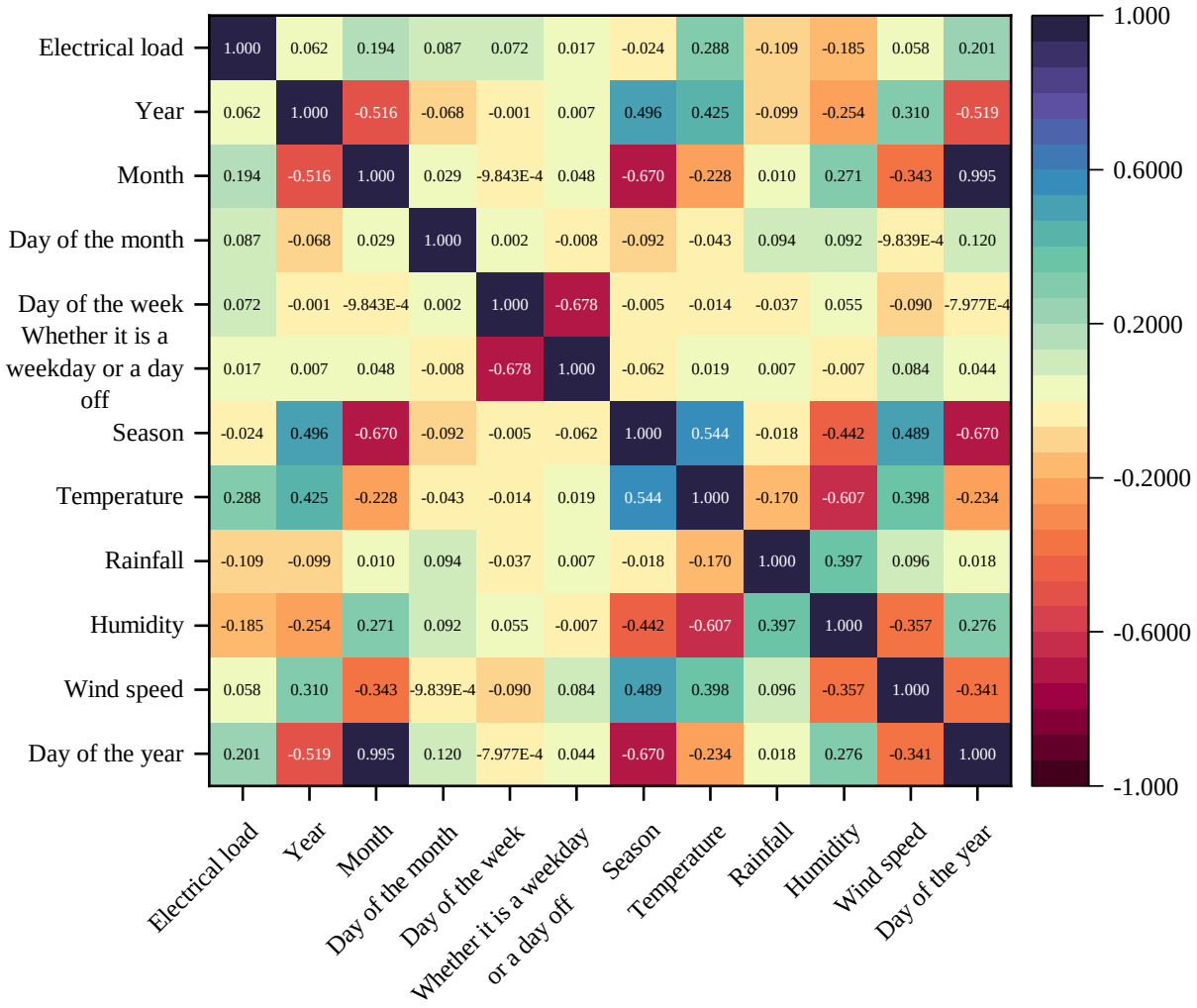


Fig. 21. Heatmap of the spearman correlation coefficient at the 1:30 moment.

Overall, the MAPE and RMSE of the model progressively decrease as the two data selection methods—PACF for selecting the optimal input data sequences and the CatBoost model for selecting the best input features—along with the hybrid forecasting strategy are gradually incorporated into the forecasting process. Specifically, the MAPE values change as follows: 2.769%, 1.557%, 2.270%, and 0.584%. Similarly, the RMSE values change as follows: 432.282 MW, 242.906 MW, 421.567 MW, and 103.714 MW. These results demonstrate that both components positively contribute to improving the model's forecasting accuracy.

In addition, Group A can be viewed as adding the use of hybrid forecasting strategy on the basis of Group B. It can also be viewed as using PACF for input data sequence selection and using CatBoost for feature selection on the basis of Group C. The forecasting accuracy of Group A is higher than that of both Group B and Group C. This suggests that in addition to improving the forecasting model, finding a suitable feature data screening method and a suitable input data sequence selection method, as well as finding a suitable forecasting strategy, can effectively improve the accuracy of load forecasting.

In addition, to illustrate the advantages of CatBoost in selecting input feature data, it is compared with the common method of using correlation coefficients for feature selection. Spearman correlation coefficients have the advantages of being able to deal with nonlinear monotonic relationships and being suitable for capturing nonlinear trends, so it is used to select the input features by calculating Spearman correlation between the feature data and the load data. If the absolute value of the correlation coefficient of a feature data is greater than or equal to 0.2, the correlation is considered high, and the feature can be selected as the input data. The Spearman correlation coefficient heatmap at 1:30 is shown in Fig. 21. Based on this analysis, the selected input feature data are temperature and day of the year. When replacing the CatBoost stage with the Spearman correlation coefficient, the forecasting results are shown in Fig. 22 and Table XII. As observed in Table XII, after replacing the input feature data, the forecasting accuracy decreases, with MAPE increasing from 0.584% to 2.923% and RMSE increasing from 103.714 MW to 502.252 MW. As shown in Fig. 22, the forecasting curve exhibits more pronounced fluctuations compared to when CatBoost is used to determine the input feature data, with more forecasting points deviating significantly from the actual load, and the deviations being larger. A comparison of the input features reveals that the number of features identified by the Spearman correlation coefficient differs from that identified by CatBoost, with CatBoost selecting a larger number of features. Moreover, only one feature, "day of the year," is common between the two methods, while the other features differ. Because CatBoost assigns scores to features based on their impact on load forecasting during the forecasting process. Higher scores indicate a more significant impact, and these features are considered more critical. In other words, CatBoost selects input features by prioritizing their contribution to forecasting accuracy. In contrast, the Spearman correlation coefficient evaluates the linear and nonlinear relationships between feature data and load data, selecting features with strong correlations. Comparing the two methods, it is evident that CatBoost has a strong advantage in selecting input feature data.

The forecasting results of the ablation experiment and Spearman correlation coefficient selection input feature method were subjected to Gaussian smoothing. The curve of the smoothing results is shown in Fig. 22. It indicates that the curve after Gaussian smoothing aligns more closely with the actual electricity load curve. The forecasting indexes after smoothing are presented in Table XIII, showing a reduction to some extent compared to the unsmoothed results. Among the ablation experiment, the MAPE of group A, group B and group D decreased by 0.035% (0.584%, 0.549%), 0.428% (2.270%, 1.842%), 0.018% (2.769%, 2.751%) respectively, while no significant change in MAPE was observed in group C. The RMSE of group A, group B, group C, and group D decreased by 7.283 MW (103.714 MW, 96.431 MW) respectively, 70.549 MW (421.567 MW, 351.018 MW), 3.1 MW (242.906 MW, 239.806 MW), and 39.222 MW (432.282 MW, 393.060 MW), respectively; and the MAPE for the Spearman correlation coefficient selection of input features method decreased by 0.218% (2.705%, 2.923%) and RMSE decreased by 84.625 MW (417.627 MW, 502.252 MW).

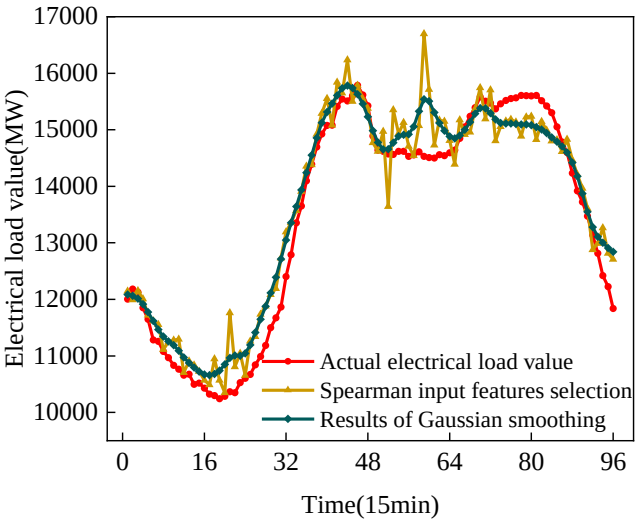


Fig. 22. Forecasting results for the input features selection method using Spearman correlation coefficient and its Gaussian smoothing results.

TABLE XII
FORECASTING INDICATORS WITH THE USE OF TWO FEATURE SELECTION METHODS

Method for selecting input features	Input features	MAPE/%	RMSE/MW
CatBoost	Day of the year Month Day of the month	0.584	103.714
Spearman	Temperature Day of the year	2.923	502.252

TABLE XIII
FORECASTING METRICS AFTER GAUSSIAN SMOOTHING USING THE FORECASTING UNDER THE TWO FEATURE SELECTION METHODS

Method for selecting input features	Input features	MAPE/%	RMSE/MW
CatBoost	Day of the year Month Day of the month	0.549	96.431
Spearman	Temperature Day of the year	2.705	417.627

The method proposed in this study achieves better results on both the Panama public dataset and the regional electric load dataset. This demonstrates that modifying the word embedding and location encoding in the input part of the Transformer into a module suitable for time-series data can improve the accuracy of short-term load forecasting. Furthermore, through two ablation experiments, it was verified that the modification to the Transformer model's input layer effectively and reliably improves load forecasting accuracy. It was also confirmed that using PACF to select the optimal input load data sequence, CatBoost to select the best input features, and the proposed hybrid forecasting strategy are all rational and effective.

Moreover, after completing the forecasting and ablation experiments on the two datasets in this study, Gaussian smoothing is applied to the forecasting results to mitigate the jagged jitter phenomenon. This approach improves the forecasting accuracy to some extent, demonstrating that Gaussian smoothing of the forecasting results is both effective and reasonable.

V. Conclusion

This paper proposes a short-term power load forecasting method based on a Transformer neural network enhanced with fused CNN-BiGRU. Under the conditions of selecting the optimal input data sequence and the best input features, the method achieves high-precision forecasting by incorporating a hybrid forecasting strategy. For input data processing, the optimal input data sequence is determined through PACF analysis of load data at each moment and daily load data in the dataset. Additionally, the best input features are selected based on the feature importance ranking results of the CatBoost model. In the forecasting network, the input layer of the Transformer is modified by replacing the word embedding and positional encoding with a CNN-BiGRU neural network. When using the Transformer fused with CNN-BiGRU for training and forecasting, a hybrid forecasting strategy is applied. This strategy involves personalized and independent model training for each time point's data, incorporating hybrid elements into the forecasting process. The final forecasting results are obtained by ordering the forecast values chronologically. Validation on two datasets shows that the modifications to the input layer of Transformer in this study are effective, and the proposed short-term power load forecasting model with Transformer fused with CNN-BiGRU is able to improve the forecasting accuracy. The results of the two ablation experiments demonstrate that the modification to the input layer of the Transformer model is both effective and reliable in improving the accuracy of load forecasting. Additionally, the PACF selecting the best input data sequence and the CatBoost model selecting the best input features, as well as the hybrid forecasting strategy are able to improve forecasting accuracy, and the model possesses the highest forecasting accuracy when both are used. In this study, Gaussian smoothing was also used to process the forecasting results after the forecasting was finished in order to improve the jagged shaped jitter phenomenon that exists in the forecasting results, which is helpful in improving the accuracy of the forecasting. The load correlation factors used in this study mainly consider meteorological factors, time factors and load's own serial correlation factors. In subsequent research, more relevant factors, such as time-of-use tariffs, can be considered for inclusion. Additionally, different forecasting strategies, as well as data and feature selection methods, can be explored to further improve model performance. Furthermore, the forecasting strategy can be optimized to eliminate the sawtooth-shaped jitter phenomenon in the forecast curve without relying on smoothing algorithms.

References

[1] Guangqi Zhang, Chuyuan Wei, Changfeng Jing and Yanxue Wang, "Short-Term Electrical Load Forecasting Based on Time Augmented Transformer," International Journal of Computational Intelligence Systems, vol. 15, no.1, pp67, 2022

[2] N. T. Dung and N. T. Phuong, "Short-term electric load forecasting using standardized load profile (SLP) and support vector regression (SVR)," Engineering, Technology & Applied Science Research, vol. 9, no.4, pp4548-4553, 2019

[3] Jian Luo, Yukai Zheng, Tao Hong, An Luo and Xueqi Yang, "Fuzzy support vector regressions for short-term load forecasting," Fuzzy Optimization and Decision Making, pp1-23, 2024

[4] Renyin Cheng, Junqi Yu, Min Zhang, Chunyong Feng and Wanhu Zhang, "Short-term hybrid forecasting model of ice storage air-conditioning based on improved SVR," Journal of Building Engineering, vol. 50, pp104194, 2022

[5] Rachna Jain, *et al.*, "A modified fuzzy logic relation-based approach for electricity consumption forecasting in India," International journal of fuzzy systems, vol. 22, pp461-475, 2020

[6] Chengdong Li, Minjia Tang, Guiqing Zhang, Ruiqi Wang and Chongyi Tian, "A hybrid short-term building electrical load forecasting model combining the periodic pattern, fuzzy system, and wavelet transform," International journal of fuzzy systems, vol. 22, pp156-171, 2020

[7] Xin Zhao, *et al.*, "Research on ultra-short-term load forecasting based on real-time electricity price and window-based XGBoost model," Energies, vol. 15, no.19, pp7367, 2022

[8] Lijie Zhang and Dominik Jánošík, "Enhanced short-term load forecasting with hybrid machine learning models: CatBoost and XGBoost approaches," Expert Systems with Applications, vol. 241, pp122686, 2024

[9] Marie Bessec and Julien Fouquau, "Short-run electricity load forecasting with combinations of stationary wavelet transforms," European Journal of Operational Research, vol. 264, no.1, pp149-164, 2018

[10] Li-Ling Peng, Guo-Feng Fan, Meng Yu, Yu-Chen Chang and Wei-Chiang Hong, "Electric load forecasting based on wavelet transform and random forest," Advanced Theory and Simulations, vol. 4, no.12, pp2100334, 2021

[11] Bo-Sung Kwon, Rae-Jun Park and Kyung-Bin Song, "Short-term load forecasting based on deep neural networks using LSTM layer," Journal of Electrical Engineering & Technology, vol. 15, pp1501-1509, 2020

[12] Mingju Gong, *et al.*, "Load forecasting of district heating system based on Informer," Energy, vol. 253, pp124179, 2022

[13] Lei Xu, Shengwei Wang and Rui Tang, "Probabilistic load forecasting for buildings considering weather forecasting uncertainty and uncertain peak load," Applied energy, vol. 237, pp180-195, 2019

[14] A. S. Khwaja, A. Anpalagan, Muhammad Naeem and Bala Venkatesh, "Joint bagged-boosted artificial neural networks: Using ensemble machine learning to improve short-term electricity load forecasting," Electric Power Systems Research, vol. 179, pp106080, 2020

[15] N. A. Mohammed and Ammar Al-Bazi, "An adaptive backpropagation algorithm for long-term electricity load forecasting," Neural Computing and Applications, vol. 34, no.1, pp477-491, 2022

[16] Jialun Zhang, Yi Wang and Gabriela Hug, "Cost-oriented load forecasting," Electric Power Systems Research, vol. 205, pp107723, 2022

[17] Yuting Lu, Gaocai Wang, Xianfei Huang, Shuqiang Huang and Man Wu, "Probabilistic load forecasting based on quantile regression parallel CNN and BiGRU networks," Applied Intelligence, vol. pp1-22, 2024

[18] Ziyu Sheng, Zeyu An, Huiwei Wang, Guo Chen and Kun Tian, "Residual LSTM based short-term load forecasting," Applied Soft Computing, vol. 144, pp110461, 2023

[19] Shizhan Liu, *et al.*, "Pyraformer: Low-complexity pyramidal attention for long-range time series modeling and forecasting," International conference on learning representations, 2021.
[20] Qin Yan, *et al.*, "An improved feature-time Transformer encoder-Bi-LSTM for short-term forecasting of user-level integrated energy loads," Energy and Buildings, vol. 297, pp113396, 2023
[21] Peng Ran, Kun Dong, Xu Liu and Jing Wang, "Short-term load forecasting based on CEEMDAN and Transformer," Electric Power Systems Research, vol. 214, pp108885, 2023
[22] P. B. Weerakody, K. W. Wong, Guanjin Wang and Wendell Ela, "A review of irregular time series data handling with gated recurrent neural networks," Neurocomputing, vol. 441, pp161-178, 2021
[23] S. Ray, R. Ray, M. H. Khondekar and K. Ghosh, "Scaling analysis and model estimation of solar corona index," Advances in Space Research, vol. 61, no.8, pp2214-2226, 2018
[24] C. H. Weiß, B. Aleksandrov, M. Faymonville and C. Jentsch, "Partial autocorrelation diagnostics for count time series," Entropy, vol. 25, no.1, pp105, 2023
[25] Zong Yuan, Taotao Zhou, Jie Liu, Changhe Zhang and Yong Liu, "Fault diagnosis approach for rotating machinery based on feature importance ranking and selection," Shock and Vibration, vol. 2021, no.1, pp8899188, 2021
[26] B. Dhananjay and J. Sivaraman, "Analysis and classification of heart rate using CatBoost feature ranking model," Biomedical Signal Processing and Control, vol. 68, pp102610, 2021
[27] Chu Zhang, Tian Peng and M. S. Nazir, "A novel integrated photovoltaic power forecasting model based on variational mode decomposition and CNN-BiGRU considering meteorological variables," Electric Power Systems Research, vol. 213, pp108796, 2022
[28] Xuechen Li, *et al.*, "Time-series production forecasting method based on the integration of Bidirectional Gated Recurrent Unit (Bi-GRU) network and Sparrow Search Algorithm (SSA)," Journal of Petroleum Science and Engineering, vol. 208, pp109309, 2022
[29] Dongxiao Niu, Min Yu, Lijie Sun, Tian Gao and Keke Wang, "Short-term multi-energy load forecasting for integrated energy systems based on CNN-BiGRU optimized by attention mechanism," Applied energy, vol. 313, pp118801, 2022
[30] Wenfeng Zheng, *et al.*, "Design of a modified transformer architecture based on relative position coding," International Journal of Computational Intelligence Systems, vol. 16, no.1, pp168, 2023
[31] Sandeep Kumar and Arun Solanki, "An abstractive text summarization technique using transformer model with self-attention mechanism," Neural Computing and Applications, vol. 35, no.25, pp18603-18622, 2023
[32] Bao-zhong Ti, Geng-yin Li and Zhao-yuan Wu, "A short-term load forecasting method based on recurrent and dilated mechanism of ConvGRU-transformer," Journal of North China Electric Power University, vol. 49, no.03, pp34-43, 2022
[33] Adam Kisvari, Zi Lin and Xiaolei Liu, "Wind power forecasting–A data-driven method along with gated recurrent neural network," Renewable Energy, vol. 163, pp1895-1909, 2021
[34] Keyu Song, *et al.*, "Short-term load forecasting based on CEEMDAN and dendritic deep learning," Knowledge-Based Systems, vol. 294, pp111729, 2024
[35] Bibi Ibrahim, Luis Rabelo, Edgar Gutierrez-Franco and Nicolas Clavijo-Buritica, "Machine learning for short-term load forecasting in smart grids," Energies, vol. 15, no.21, pp8079, 2022
[36] Chen Xiaoqiang, Chen Xinhao, Gu Shaowu and Xie Lei, "Research on power load forecasting method based on VBLA model," Manufacturing Automation, vol. 45, no.11, pp178-184, 2023