IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

SPECIAL SECTION ON KEY ENABLING TECHNOLOGIES FOR PROSUMER ENERGY MANAGEMENT

Received December 16, 2020, accepted December 30, 2020, date of publication January 13, 2021, date of current version May 14, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3051337

Multi-Model Fusion Short-Term Load Forecasting Based on Random Forest Feature Selection and Hybrid Neural Network

YI XUAN[1], WEIGUO SI[1], JIONG ZHU[1], ZHIQING SUN[1], JIAN ZHAO[2], (Member, IEEE), MINGJIE XU[2], AND SHOULIANG XU[2]

[1]Hangzhou Power Supply Company, State Grid Zhejiang Electric Power Company Ltd., Hangzhou 310007, China
[2]Electric Power Engineering, Shanghai University of Electric Power, Shanghai 200090, China

Corresponding author: Jian Zhao (zhaojianee@foxmail.com)

ABSTRACT In an increasingly open electricity market environment, short-term load forecasting (STLF) can ensure the power grid to operate safely and stably, reduce resource waste, power dispatching, and provide technical support for demand-side response. Recently, with the rapid development of demand side response, accurate load forecasting can better provide demand side incentive for regional load of prosumer groups. Traditional machine learning prediction and time series prediction based on statistics failed to consider the non-linear relationship between various input features, resulting in the inability to accurately predict load changes. Recently, with the rapid development of deep learning, extensive research has been carried out in the field of load forecasting. On this basis, a feature selection algorithm based on random forest is first used in this paper to provide a basis for the selection of the input features of the load forecasting model. After the input features are selected, a hybrid neural network STLF algorithm based on multi-model fusion is proposed, of which the main structure of the hybrid neural network is composed of convolutional neural network and bidirectional gated recurrent unit (CNN-BiGRU). The input data is obtained by using long sliding time windows of different steps, then multiple CNN-BiGRU models are trained respectively. The forecasting results of multiple models are averaged to get the final forecasting load value. The load datasets come from a region in New Zealand and a region in Zhejiang, China, are used as load forecast examples. Finally, a variety of load forecasting algorithms are introduced for comparison. The experimental results show that our method has a higher accuracy than comparison models.

INDEX TERMS Short-term load forecasting, prosumer, random forest algorithm, convolutional neural network, bidirectional GRU, multi-model fusion.

I. INTRODUCTION

The smart grid is a key component of future sustainable strategy, and high accuracy of power load forecasting is not only the precondition of the efficient operation of the smart grid, but also decrease the cost of distribution network, energy loss, At the same time, STLF is also the basis of the electric power dispatching department work and guides the economic operation of power system. High-precision short-term load forecasting is significant to realize the balance of demand of power system and reduce the waste of resource [1]. Therefore, load forecasting technology has been a hot field in the past decades. With the continuous construction and development of advanced measurement system, massive load data are collected by intelligent measurement terminals, and how to make full use of massive multi-dimensional measurement data for load forecasting becomes a new problem. At the same time, with the rapid development of calculating machine, deep learning has been widely used in the field of image processing, natural language processing topology identification [2] and load forecasting.

Load forecasting techniques mainly include parametric and non-parametric methods. Parametric methods include linear regression [3], autoregressive integral moving average, synthetic exponential technique and chaotic time series technique. These methods have simple structure, but they require high stability of original data processing and time series, and are difficult to reflect the influence of nonlinear factors [4],

The associate editor coordinating the review of this manuscript and approving it for publication was Bin Zhou.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

so they are suitable for situations with few influencing factors. The non-parametric load forecasting technology is mainly artificial neural network (ANN) [5]. ANN has the ability of fault-tolerant ability and nonlinear map ability, and can fully solve the nonlinear problems that existing in large-scale load data, so it is widely used in prediction problem. Reference [6] introduces an ELU excitation function, multi-layered deep Neural network (DNN) is used for load prediction. However, researchers find that ANN is difficult to learn the correlation of serial data, and usually used historical load data to artificially extract time features and establish the relationship between input and output variables. Load series is characterized by nonlinear, unstable and dynamic changes, the output results of the model is related to the current input and the past input [7]. With the extensive research on deep learning and the rapid development of computer technology, more and more complex neural networks, which is based on deep learning, are used for processing time series data. CNN model is used to extract the characteristics of input data and capture the seasonal cycle to carry out load prediction [8], [9], so they can learn the nonlinear relationship between input feature data and achieve higher accuracy of load prediction. RNN introduces a cyclic structure in the network, which enables it to process time series data better than other neural networks. Among them, the variant structure LSTM of RNN has been widely studied in forecasting filed. A STLF model based on multi-layer bidirectional circulation neural network is proposed in [7]. In the forecasting process, the past and future state information of the forecast points are fully considered, and the deep learning model is used to receive a more accurate scores compared with the traditional forecasting model. A LSTM recurrent neural network-based structure was proposed to predict the load of residential customers in [10], which is also tested on the open real dataset, and its forecasting performance reach the best in this field. Paper [11] also uses RNN neural network to forecast resident load, and introduces the concept of load pool, which greatly improves the accuracy of resident load forecasting. However, compared with LSTM network [12], GRU network reduces the number of gated loop units from three to two. GRU neural network uses gated cycle unit instead of traditional cycle unit to solve the gradient disappearance problem that is prone to occur in RNN [13]. Several studies have shown that LSTM and GRU networks can effectively improve the accuracy.

Load information is a kind of high-dimensional time series data, which is not only affected by the weather, temperature and humidity, but affected by the actual load of electricity and the price of electricity. However, the existing algorithms only focus on the single factor or specific factors to study the power load prediction technology, and the design of multi-load factor high-precision forecasting method is scarce. In order to fully excavate the multiple factors of given load data and get a better predicting result, the main contributions of this paper are as follows. First, random forest algorithm is used to select the input features of power system load dataset for a more accurate forecasting results. Then, multi-model fusion CNN-BiGRU hybrid neural network is introduced to predict short-term load after feature selection. Compared with traditional neural network prediction, short-term load prediction accuracy is greatly improved by introducing feature selection and hybrid neural network.

The paper is organized as follows. Section II presents the feature selection algorithm based on random forest model. Section III presents some principles of deep learning model, including CNN model and BiGRU model. Section IV presents our proposed load forecasting model, including feature selection, multi-time sliding window data processing and integrated hybrid neural network prediction model. Section V presents our algorithm forecasting results. The paper concludes in section VI with a discussion of the results.

II. FEATURE SELECTION BASED ON RANDOM FOREST

A. RANDOM FOREST MODEL

Breiman invented the classification tree algorithm in the 1980s, which greatly reduced the amount of computation by classifying or regressing binary data. In 2001 Breiman combined the classification tree into a random forest by randomizing the use of variables and data to generate many classification trees and then summarizing the results of the classification tree. Random forest improves the prediction accuracy without significant increase in computation [14]. Random forests are not sensitive to multivariate common linearity, and the results are robust to missing data and unbalanced data. They can well predict the role of up to thousands of explanatory variables, which is known as one of the best algorithms at present.

The random forest regression model is composed of multiple regression trees, and there is no correlation between each decision tree in the random forest. The final output of the model is determined by each decision tree in the random forest. After the random forest model is obtained, each decision tree in the random forest is judged separately when the new sample enters. Random forests can deal with the amount of discrete values, such as ID3 algorithm, or the amount of continuous values, such as C4.5 algorithm. In addition, random forests can also be used for unsupervised learning clustering and outlier detection.

Its algorithm principle is as follows:

a) Randomly extract m sample points from the training sample set S, get a new $S_1, \ldots, S_n$ sub-training set;
b) Sub-training set is used to train a CART regression tree (decision tree). In the training process, the segmentation rule for each node is to randomly select k features from all features, and then select the optimal segmentation point from these k features to divide the left and right sub-trees. (The decision obtained here are binary trees);
c) Through the second step, many CART regression tree models can be generated;
d) The final prediction results of each CART regression tree are the mean values of leaf nodes from the sample point;

e) The final prediction result of random forest is the mean of all CART regression tree prediction results.

B. RANDOM FOREST FEATURE SELECTION ALGORITHM

Random forest is based on random subspace theory and bootstrap method [15], the vector (X, Y) is randomly selected and the tree is grown into a decision tree. The optimal classification result is given for each tree, and the final result is the choice with the most votes among the k trees.

Assumed that the training data (X, Y) contains n samples, randomly put back and extract b subsets to build a regression tree. When the $i(i \in b)$ subset is extracted, the other samples are out-of-bag (OOB) data. In addition, the vector with a fixed dimension of $m(m < M)$ is selected from the M-dimensional vector as the input variable, and the feature space of the regression tree is constructed. The splitting variable is selected by the minimum variance criterion during the splitting growth process.

As follows:

$$I = \min_{s} \frac{\sum_{s=1}^{n} (X_s - X_s')^2}{n} \tag{1}$$

where I is the optimal splitting variable, and S is the embedded sample dimension. X_s, X_s' respectively represent the value and average value of variables. After tree growth is completed, a random forest is formed, and then the influence of data outside the bag on the model is calculated:

$$MSE = \frac{\sum_{i=1}^{n} (y_i - y_i')^2}{n} \tag{2}$$

where MSE is mean square residual. n denotes the sample size of data outside the bag; y_i, y_i' respectively represent the true and predicted charges of i set of samples respectively. The RF model reduces the importance of the indicator input variables by means of mean square residuals, generates K regression trees with out-of-pocket data to obtain the rank of mean square residuals in $[MSE_1, MSE_2, \ldots, MSE_K]$, and generates K new regression trees after multiple sampling to form the out-of-pocket data residuals matrix. For the m input variable, its importance is measured as follows:

$$Vim = \frac{\sum_{j=1}^{k} (MSE_j - MSE_{kj})}{k \cdot S_E} \tag{3}$$

where Vim is the importance score of the variable, $k(j \in k)$ is the number of decision trees, S_E is the standard error of K decision trees.

III. DEEP LEARNING MODEL PRINCIPLE

A. CNN MODEL

Convolutional neural network (CNN) is an important algorithm of deep learning. The CNN is an improvement on BP neural network [16]. CNN has been used in many fields, such as pattern classification, object detection. It is a front collapse neural network. It can extract its topology from a two-dimensional image. It uses back-propagation algorithm to optimize the network structure and solve the unknown parameters in the network. In addition, because CNN has fewer parameters, it can avoid over fitting phenomenon.

CNN model uses local connection and weight sharing to process the raw data at a higher level and more abstract, which can extract the deep nonlinear features of the processed data. The internal neural network layer is mainly composed of convolution layer, sampling layer (pool layer) and full connection layer. This structure has less number of weights and accelerates the convergence speed. By using convolution layer and pool layer to obtain effective information, the model can extract the feature vectors that hidden in the input data through three different network layer depths, which enhances the ability of feature extraction of data, and reduces the computational cost and complexity of the whole model. At the same time, by automatically extracting feature vectors from data, the complexity of feature extraction and data reconstruction is effectively reduced, and the quality of data features is improved.

The CNN network structure regards the input as an image, and by compiling specific features into the convolution structure, the efficiency of the forward transfer function is improved and the number of parameters in the network is reduced [17], [18]. The one-dimensional convolutional layer can realize the feature extraction of the time axis, and its output is:

$$y_i = \sigma(\sum_{j=1}^{k} w_j \cdot x_{i-j-k} + b) \tag{4}$$

where y_i is the output of convolutional layer; σ is the excitation function. k is the number of convolution kernels; w_j is the convolution kernel weight matrix; $x_{(i-j+k)}$ is the input time series and $1 \leq i - j + k \leq n$. b is the deviation value.

By setting the convolution layer and pooling of alternating structure realize effective local characteristics for data acquisition, and the output characteristic vector, the input data for the load data of the proposed model, by using CNN network structural characteristics of the local connection and multi-level structure, tell the partial correlation between loads, increase the low-level features. As CNN networks deepens, the low-level feature combination for multi-level characteristics, may guide the subsequent model for such characteristics of the learning and adjustment.

B. BIGRU MODEL

Gated Recurrent Unit (GRU), which is similar to LSTM, but GRU is more simple and efficient than the LSTM model in some specific problem. LSTM can capture long-term dependence and is suitable for analyzing time series data. However, the complex internal structure leads to longer training time. GRU has two gates, namely, an update gate and a reset gate [19]. Intuitively, the update gate is responsible for controlling the input of the memory information extracted by the front neurons into the current time step, and at the same time, it can decide what information to discard and what new information to add. The reset gate uses its internal activation function to determine the degree of discarding the previous

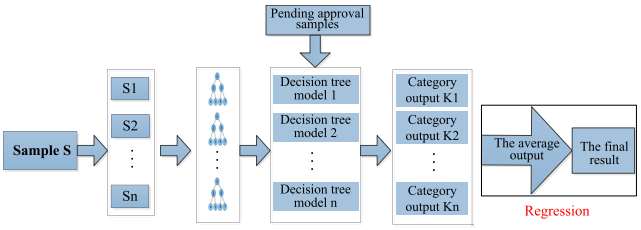


FIGURE 1. Framework of random forest algorithm.

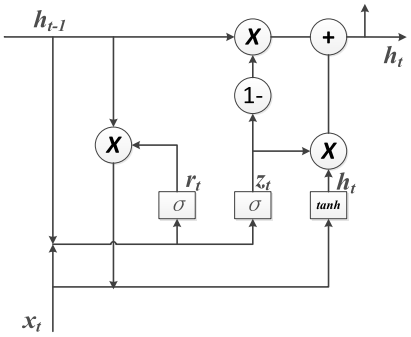


FIGURE 2. A structure of GRU neural network circulation.

information. GRU neural network only needs two gated loop units to solve the gradient problem, which saves memory and speeds up the running speed to a certain extent. At the same time, the cost of calculation time and computational resources is reduced greatly. A basic GRU structure is shown in Figure 2:

The direction indicated by the arrow is the direction of data flow, X is the multiplication of the matrix, *tanh* is the activation function, and 1- means that the data propagated forward by the link is $1 - zt$. h_{t-1} is the hidden layer state at the previous moment, r_t is the reset gate, z_t is the update gate, σ is the sigmoid activation function [20], [21]. x_t is the input information and h_t is the output data of the hidden layer. The GRU based unit calculates h_t by the following formula:

$$z_t = \sigma\left(W^{(z)}x_t + U^{(z)}h_{t-1}\right) \tag{5}$$

$$r_t = \sigma\left(W^{(r)}x_t + U^{(r)}h_{t-1}\right) \tag{6}$$

$$h_t = \tanh(r_t \odot Uh_{t-1} + Wx_t) \tag{7}$$

$$h_t = (1 - z_t) \odot h_t + z_t \odot h_{t-1} \tag{8}$$

where $\tilde{h}_t$ is the summary of input x_t and the past hidden layer state h_{t-1}, $U^{(z)}$, $W^{(z)}$, $U^{(r)}$, $W^{(r)}$, U and W is a trainable parameter matrix.

However, both the GRU and the LSTM network can only consider the information of the past time of the prediction point, but cannot consider the state of the future time, so the prediction accuracy cannot be further improved [22], [23]. BiGRU adds a hidden layer on the basis of bidirectional GRU neural network, divides the prediction process into two directions: forward prediction and backward prediction, and determines the output result jointly by the hidden layer of the two directions. The structure of BiGRU is shown in Figure 3. where W_f is the weight of the input layer to the hidden layer

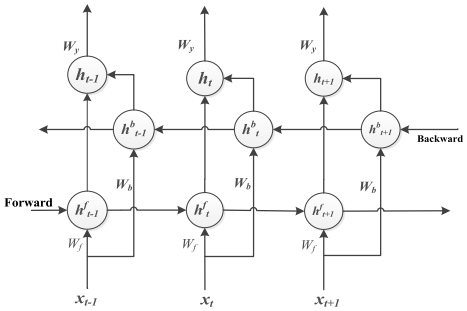


FIGURE 3. Structure of BiGRU neural network.

in the forward prediction process, W_b is the weight of input layer to hidden layer in the backward prediction process, h_t^f is the hidden layer state in the forward prediction process, and h_t^b is the hidden layer state in the backward prediction process.

As shown in figure 3, BiGRU prediction process consists of two directions, one is the forward process, the other is backward process. Forward process is a traditional one-way GRU neural network, and the backward process is a prediction process in the opposite direction corresponding to the forward process. The backward prediction process can consider the influence of the data at the back of the prediction point on the current hidden layer, and the hidden layer state of the prediction result is determined by the hidden layer in both directions, so the BiGRU neural network combining forecasting point in the past and the future state of hidden layer, which improves the prediction accuracy efficiently.

IV. LOAD FORECASTING MODEL

A. FEATURE SELECTION BASED ON RANDOM FOREST ALGORITHM

Before load forecasting of given data, the input characteristic data of load data set is first screened to prevent the overfitting of prediction model. At the same time, reducing the redundancy of features and removing attribute data which have little effect on actual load data can get a more accurate result and reduce the computational cost of algorithm convergence. Assume the load data set's characteristic attribute data $\{x^{(1)}, x^{(2)}, x^{(3)\cdots}x^{(n)}, y^{load}\}$, where $x^{(i)}$ is the characteristic data related to the actual load, such as temperature, humidity, etc., and y^{load} represents the actual load data. For the input data set containing multiple features, a simple forecasting model, which is based on the random forest machine learning, is directly established to obtain the mapping relationship between the input feature data and the target predictive load value, and then the input data are screened by the feature selection algorithm based on the random forest algorithm.

B. MULTI-TIME SLIDING WINDOW DATA PROCESSING

Weather information, electricity price information, load historical data and holiday information are all independent of each other in time series, and are also important feature information affecting load changes [16]. Therefore, on the basis of

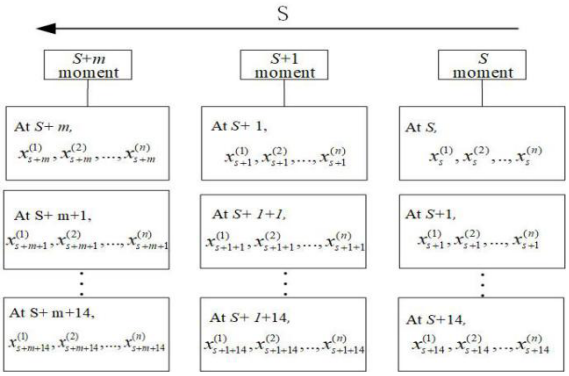


FIGURE 4. Stacking-LSTM network model input data structure.

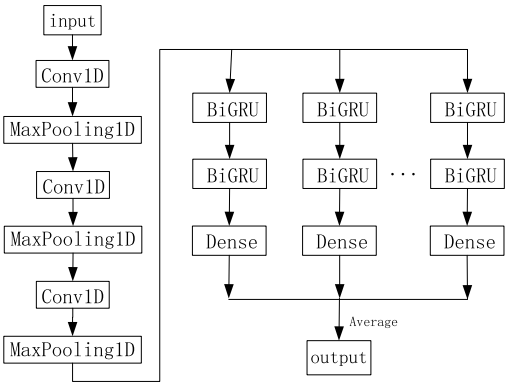


FIGURE 5. Schematic diagram of integrated hybrid neural network.

above analysis, the time sequence characteristics map as input of neural network hybrid model, with reference to the vector representation of words in a natural language processing, a time of load value through series for vector expression and its associated characteristics, creating a new time series data, achieve the coupling impact load characteristic information. The input time series data are generated by using sliding window of different step size. Assuming that the width of the sliding window is l and the original feature set of the input load data set is $\{x^{(1)}, x^{(2)}, x^{(3)\cdots}x^{(n)}\}$, among them $x^{(i)} = \{x_{(i)\,1}, x_2^{(i)}, \ldots, x_{t-1}^{(i)}, x_t^{(i)}\}$ and t is time measurement point, the input feature map is also sorted according to time series, as shown in Figure 4:

A specific time with an interval of 15 min is expressed as S, Time a certain time of S backward m 15min interval is represented by $(S + m)$, and the coordinate axis is represented by S, when time is used as the scale.

C. INTEGRATED HYBRID NEURAL NETWORK PREDICTION MODEL

After feature screening based on random forest algorithm, the data input after screening and time window processing is integrated into the prediction model. The integrated hybrid neural network prediction model is mainly composed of multiple CNN-BiGRU sub-neural network modules. The neural network architecture of each CNN-BiGRU prediction module is the same, but the input data of each module is not the same. The input data of each module is the input data processed by sliding time windows of different widths.

Sub-prediction modules consist of Conv1d module (one-dimensional CNN), one-dimensional MaxPooling module and BiGRU module. Conv1d and MaxPooling are mainly used for feature extraction, while BiGRU network is used for load forecasting. After inputting the data, the Conv1d is firstly used to extract the features of the input time series. The one-dimensional convolutional neural network can well identify the change rule of time series data and hidden information. One dimensional convolutional neural network has been well applied in the time series analysis, time series regression and time series classification. Besides, it can also be used in the analysis of signal data. After feature extraction in the convolutional layer, the one-dimensional pooling layer is used for data sampling. The biggest benefit of pooling layer is that the data can be dimensioned, redundant information can be removed, and features can be compressed to simplify the network complexity, thus reducing computation and memory consumption, etc. After multiple convolutional layers are pooled, the extracted feature data are fed into the bidirectional GRU network. GRU has gated cyclic structure. However, compared with LSTM structure, GRU has fewer training parameters and maintains the prediction effect of LSTM. When a sub-forecasting module completes the load forecasting, the forecast results will be output. After all sub-forecasting modules complete the forecast, the prediction results of all sub-forecasting modules will be averaged to get the final prediction result. The specific principle is shown in Figure 5:

V. CASE STUDY

In order to verify the accuracy of our algorithm, the electric load data sets of a certain region in New England and a certain region in Zhejiang, China, are used for experimental analysis. The New England data set includes a total of 365 days of power load data in 2017, including 13 load influencing factors such as real-time electricity price, temperature, humidity and wind speed. The data is collected every one hour and a total of 8,760 data samples are named as dataset 1. China's Zhejiang dataset includes data from January 1, 2018 to May 31, 2019, collected in every 30 minutes. The data features and attributes are similar to those of New England, and are named dataset 2. The hardware platform of this example experiment is a dell high-performance workstation with i7-8700K processor, 64GB memory, 1TB solid-state disk capacity, and RTX2080TI GPU graphics card. The software platform is Jupyter Notebook, and the main development library is Sklearn machine learning library and Keras deep learning library based on Tensorflow. Keras is programmed by Python, which is an end-to-end deep learning platform, providing high-level neural network programming interface, with simple and flexible architecture, supporting the free combination of neural network models and layer upon layer superposition.

A. DATA PREPROCESSING

If the data is not normalized, the proposed model will need a longer training time. Therefore, after acquiring the load data, the values of multiple features will be normalized. Through min-max normalization, the original data are converted into data falling between (0, 1), and the magnitude of the data value is normalized, so that the prediction model training can achieve faster convergence [24]. The normalization formula is:

$$x^* = \frac{x - \min(x)}{\max(x) - \min(x)} \tag{9}$$

x denotes each characteristic data in the load data set, x^* is the normalized data. In order to evaluate the accuracy of model prediction, we select the average absolute percentage error (*MAPE*) and root mean square error (*RMSE*) as evaluation metric, and their expressions are respectively:

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\frac{|y_i - \overline{y_i}|}{y_i} * 100\% \tag{10}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \overline{y_i})^2} \tag{11}$$

B. FEATURE SELECTION AND MODEL TRAINING

After completing data preprocessing, before the feature selection, in order to consider each feature's time lags, we need to shift the input features. Assuming the time sequencing of feature i is $\{x_{(i)\,1}, x_2^{(i)}, \ldots, x_{t-1,}^{(i)} x_t^{(i)}\}$, $1 \le t \le 8760$. The features of backward translation n time units, the characteristics of the amount of time sequences into $\{x_d^{(i)}, x_{d+1}^{(i)}, \ldots, x_{t-d}^{(i)}\}$. When completing the shift operation of selected feature, we could find that the length of the translation features of new generation is not equal to the original time series length, and new features need to adjust the input data, to consider the consistency of dimension of input data, this paper directly uses the shortest length of feature vector dimensions as random forest input data dimensions after multiple time shift of feature vector dimensions, removes excess portion of the value. The translated feature sequences are recombined to form a new load dataset.

The transformed load data set is used for feature selection using the random forest algorithm, the actual load data is taken as the target value, other load-related data are taken as input variables, the random forest regression is used for forecasting, and the importance degree of each feature after prediction is visualized. The visualization of dataset 1 is shown in Figure 6. Here, the feature attribute whose characteristic contribution degree is in the top 6 is selected.

For dataset 1, the top 6 of its feature contributions are temperature, temperature-shfit-1, electricity price, humidity, humidity-shift-1, date type. The feature with shift in the label is the feature data after translation processing. The visualization of each feature contribution of dataset 2 is shown in Figure 7. Here, the feature attribute whose feature contribution is in the top 6 is selected. By the feature selection of random forest algorithm, the features with high contribution are selected. Random forest algorithm ranks by the contribution degree of features, and the features with high contribution often have an important impact on the accuracy of prediction.

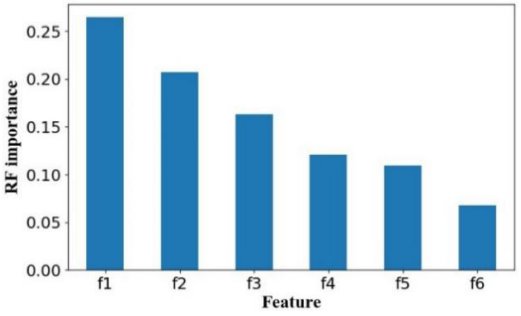


FIGURE 6. Ranking of feature contribution of dataset 1.

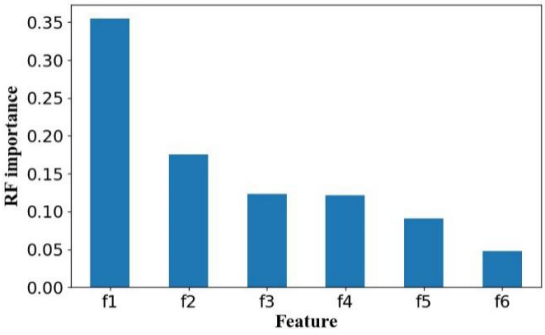


FIGURE 7. Ranking of feature contribution of dataset 2.

TABLE 1. Comparison of different timesteps in dataset 1.

Timestep	5	10	20	50	Stack
MAPE(%)	4.31	4.28	4.32	4.32	4.22
RMSE(kw)	14.63	14.01	14.93	15.41	14.17

For dataset 2, the top 6 feature contributions are temperature, temperature-shfit-3, temperature-shfit-1, humidity, humidi-ty-shift-3, date type.

According to the ranking of feature contributions, feature data and load data ranked in the top six of feature contributions are selected as input data in the next part. Using these data can greatly reduce the dimension of input data, reduce feature redundancy, and speed up the calculation at the same time.

After filtering the data, before model training, we need to split the dataset into train dataset and test dataset. The data set was split by a ratio 9:1, and the large one is train set. The sliding time window method with different widths and synchronously long lengths is used to carry out feature transformation respectively, so as to make the input data meet the input requirements of CNN-BiGRU. Time windows with sliding window width of 5, 10, 20, 50 are selected respectively to resampling the input load data set, and the sampled data with different time Windows of different widths are trained with different sub-forecasting modules, and then the average value of all the predicted results was taken for training. The performance of forecasting results is shown in Table 1 and Table 2:

TABLE 2. Comparison of different timesteps in dataset 2.

Timestep	5	10	20	50	Stack
MAPE(%)	5.12	5.14	5.11	5.15	5.08
RMSE(kw)	5.48	5.39	5.40	5.27	5.11

TABLE 3. Comparison of different models in dataset 1.

	Mertic	CNN-NN	CNN-GRU	LSTM	CNN-LSTM	Our model
Dataset1	MAPE(%)	5.82	4.53	5.21	4.49	4.22
	RMSE(kw)	22.46	15.28	19.64	16.56	14.17

TABLE 4. Comparison of different models in dataset 2.

	Mertic	CNN-NN	CNN-GRU	LSTM	CNN-LSTM	Our model
Dataset2	MAPE(%)	6.96	5.53	6.20	5.69	5.08
	RMSE(kw)	7.87	6.15	6.95	5.89	5.11

It can be concluded from Table 1 and Table 2 that when the input data of the CNN-BiGRU model adopts sliding time Windows with different widens, the predicted results are different. For Dataset 1, when the sliding window's step is 10, the prediction accuracy is the highest; for dataset 2, when selecting 20 as the window size, the prediction accuracy is the highest. However, for all two datasets, the prediction accuracy of stack model is higher than that of any single model, which proves the superiority of the integrated CNN-BiGRU model over the single CNN-BiGRU model. Besides, in order to compare the algorithm performance of proposed model and other prediction models, this paper also introduces CNN-NN, CNN-GRU [24], LSTM [25] and CNN-LSTM for comparison. The results of different metric are shown in Table 3 and Table 4:

The forecasting accuracy of our method is the highest compared with the other four methods. For dataset 1, the first metric MAPE decreased by 1.6%, 0.31%, 0.99%, 0.27%, at the same time, the other metric RMSE decreased by 36.9%, 7.26%, 27.85%, and 14.43% when proposed model is compared with another four models. For dataset 2, the first metric MAPE decreased by 1.88%, 0.25%, 1.12%, and 0.61%, at the same time, the other metric RMSE decreased by 35.06%, 16.91%, 26.47%, and 13.24%, when proposed model is compared with another four models. Based on the comprehensive analysis, both two metrics decrease significantly when using the method presented in this paper, indicating that the precision of the whole volume prediction and the performance of the model are greatly improved in the process of prediction.

To verify the accuracy of multiple CNN-BiGRU model fusion is better than that of a single CNN-BiGRU model, we compare the forecast results of different datasets. For dataset 1, we choose the real load data of November 2017 to compare with the load data predicted by the model. The results are shown in figure 8:

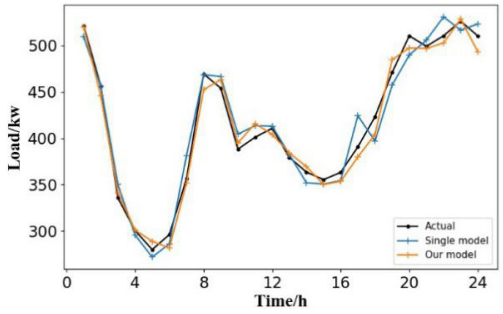


FIGURE 8. Comparison of forecasting curve and actual load curve of data set 1 single model and multi-model fusion.

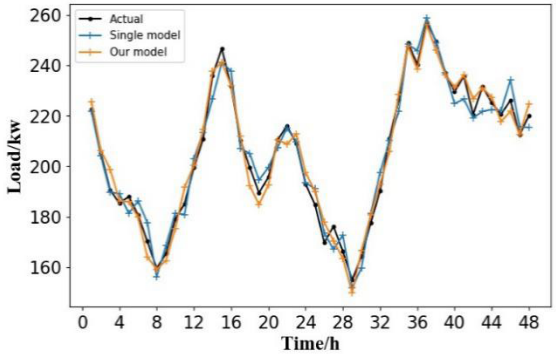


FIGURE 9. Comparison of forecasting curve and actual load curve of data set 2 single model and multi-model fusion.

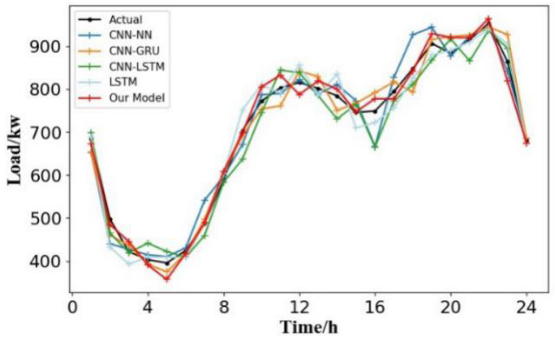


FIGURE 10. Comparison of forecast results on December 31, 2017.

For dataset 2, this paper selects the real load data from April 2018 to compare with the load data predicted by the model. The results are shown in figure 9:

By comparison, it can be found that the forecasting load value of the CNN-BiGRU model of a single model deviates seriously from the actual value in some periods, while the multi-model fusion prediction network uses multiple length time windows for feature processing. The average results of multiple prediction models have a higher accuracy than a single model.

In order to present the load forecasting results more clearly and intuitively, Fig. 10 is the comparison diagram of the real load value of dataset 1 on December 31, 2017 and the short-term load forecasting curves of different methods. Fig. 11 compares the real load value of dataset 2 on May 31, 2019 with the short-term load forecasting curves of different

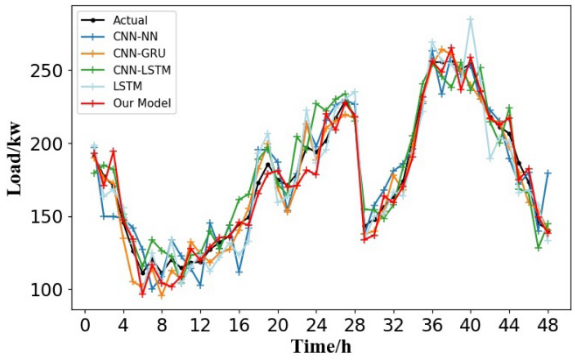


FIGURE 11. Comparison of forecast results on May 31, 2019.

methods. It can be seen from Fig. 10 and Fig. 11 that our model has a better performance and a higher accuracy. For dataset 1, the load curve is relatively smooth, so all the forecasting curves have a small error. However, compared with other model, our model has the smallest error and accurate prediction of the curve trend. For dataset 2, the daily load changes greatly, and the load value changes significantly in the morning and noon periods. Compared with other models, the method in this paper can predict the load value more accurately and smoothly at this period, and can better capture the load change rule in the relatively gentle period.

VI. CONCLUSION

In this paper, a feature selection method based on random forest algorithm is proposed, and a multi-model integrated prediction algorithm based on CNN-BiGRU hybrid neural network is proposed based on feature selection. The algorithm uses various width sliding time windows to obtain different data, trains a sub-forecasting model separately, and finally takes the average value of the forecasting results of multiple sub-models to obtain the final forecasting results. Finally, through data verification, compared with a single CNN-BiGRU prediction model, the proposal model achieves accurate load forecasting results on both MAPE and RMSE. Moreover, compared with other deep learning prediction models, the proposal model also performs better than the other four comparison models in terms of MAPE and RMSE. In future studies, this method will continue to be applied to ultra-short-term power load forecasting, and further adjust and optimize this method through data sets such as power load data and meteorological data of more regions.

REFERENCES

[1] K. Chen, K. Chen, Q. Wang, Z. He, J. Hu, and J. He, "Short-term load forecasting with deep residual networks," *IEEE Trans. Smart Grid*, vol. 10, no. 4, pp. 3943–3952, Jul. 2019, doi: 10.1109/TSG.2018.2844307.

[2] G. Dudek, "Pattern-based local linear regression models for short-term load forecasting," *Electr. Power Syst. Res.*, vol. 130, pp. 139–147, Jan. 2016.

[3] J. Zhao, L. Li, Z. Xu, X. Wang, H. Wang, and X. Shao, "Full-scale distribution system topology identification using Markov random field," *IEEE Trans. Smart Grid*, vol. 11, no. 6, pp. 4714–4726, Nov. 2020, doi: 10.1109/TSG.2020.2995164.

[4] Y. Wang, Q. Chen, N. Zhang, and Y. Wang, "Conditional residual modeling for probabilistic load forecasting," *IEEE Trans. Power Syst.*, vol. 33, no. 6, pp. 7327–7330, Nov. 2018, doi: 10.1109/TPWRS.2018.2868167.

[5] H. Wang, Y. Liu, B. Zhou, C. Li, G. Cao, N. Voropai, and E. Barakhtenko, "Taxonomy research of artificial intelligence for deterministic solar power forecasting," *Energy Convers. Manage.*, vol. 214, Jun. 2020, Art. no. 112909.

[6] T. Hossen, S. J. Plathottam, R. K. Angamuthu, P. Ranganathan, and H. Salehfar, "Short-term load forecasting using deep neural networks (DNN)," in *Proc. North Amer. Power Symp. (NAPS)*, Sep. 2017, pp. 1–6.

[7] X. Tang, Y. Dai, T. Wang, and Y. Chen, "Short-term power load forecasting based on multi-layer bidirectional recurrent neural network," *IET Gener., Transmiss. Distrib.*, vol. 13, no. 17, pp. 3847–3854, Sep. 2019, doi: 10.1049/iet-gtd.2018.6687.

[8] H. Zang, L. Cheng, T. Ding, K. W. Cheung, Z. Liang, Z. Wei, and G. Sun, "Hybrid method for short-term photovoltaic power forecasting based on deep convolutional neural network," *IET Gener., Transmiss. Distrib.*, vol. 12, no. 20, pp. 4557–4567, Nov. 2018, doi: 10.1049/iet-gtd.2018.5847.

[9] Y. Wang, Q. Chen, D. Gan, J. Yang, D. S. Kirschen, and C. Kang, "Deep learning-based socio-demographic information identification from smart meter data," *IEEE Trans. Smart Grid*, vol. 10, no. 3, pp. 2593–2602, May 2019, doi: 10.1109/TSG.2018.2805723.

[10] W. Kong, Z. Y. Dong, Y. Jia, D. J. Hill, Y. Xu, and Y. Zhang, "Short-term residential load forecasting based on LSTM recurrent neural network," *IEEE Trans. Smart Grid*, vol. 10, no. 1, pp. 841–851, Jan. 2019, doi: 10.1109/TSG.2017.2753802.

[11] H. Shi, M. Xu, and R. Li, "Deep learning for household load forecasting—A novel pooling deep RNN," *IEEE Trans. Smart Grid*, vol. 9, no. 5, pp. 5271–5280, Sep. 2018, doi: 10.1109/TSG.2017.2686012.

[12] Y. Wang, D. Gan, M. Sun, N. Zhang, Z. Lu, and C. Kang, "Probabilistic individual load forecasting using pinball loss guided LSTM," *Appl. Energy*, vol. 235, pp. 10–20, Feb. 2019.

[13] E. Pan, X. Mei, Q. Wang, Y. Ma, and J. Ma, "Spectral-spatial classification for hyperspectral image based on a single GRU," *Neurocomputing*, vol. 387, pp. 150–160, Apr. 2020.

[14] J. Liu and Y. Li, "Study on environment-concerned short-term load forecasting model for wind power based on feature extraction and tree regression," *J. Cleaner Prod.*, vol. 264, Aug. 2020, Art. no. 121505.

[15] S. Liu, H. Li, Y. Zhang, B. Zou, and J. Zhao, "Random forest-based track initiation method," *J. Eng.*, vol. 2019, no. 19, pp. 6175–6179, Oct. 2019, doi: 10.1049/joe.2019.0180.

[16] B. K. Oh, B. Glisic, Y. Kim, and H. S. Park, "Convolutional neural network-based wind-induced response estimation model for tall buildings," *Comput.-Aided Civil Infrastruct. Eng.*, vol. 34, no. 10, pp. 843–858, Oct. 2019.

[17] Z. Deng, B. Wang, Y. Xu, T. Xu, C. Liu, and Z. Zhu, "Multi-scale convolutional neural network with time-cognition for multi-step short-term load forecasting," *IEEE Access*, vol. 7, pp. 88058–88071, 2019, doi: 10.1109/ACCESS.2019.2926137.

[18] M. Alhussein, K. Aurangzeb, and S. I. Haider, "Hybrid CNN-LSTM model for short-term individual household load forecasting," *IEEE Access*, vol. 8, pp. 180544–180557, 2020, doi: 10.1109/ACCESS.2020.3028281.

[19] M. Sajjad, Z. A. Khan, A. Ullah, T. Hussain, W. Ullah, M. Y. Lee, and S. W. Baik, "A novel CNN-GRU-based hybrid approach for short-term residential load forecasting," *IEEE Access*, vol. 8, pp. 143759–143768, 2020, doi: 10.1109/ACCESS.2020.3009537.

[20] S. Xu, J. Li, K. Liu, and L. Wu, "A parallel GRU recurrent network model and its application to multi-channel time-varying signal classification," *IEEE Access*, vol. 7, pp. 118739–118748, 2019, doi: 10.1109/ACCESS.2019.2936516.

[21] C. Xu, J. Shen, X. Du, and F. Zhang, "An intrusion detection system using a deep neural network with gated recurrent units," *IEEE Access*, vol. 6, pp. 48697–48707, 2018, doi: 10.1109/ACCESS.2018.2867564.

[22] A. Cura, H. Kucuk, E. Ergen, and I. B. Oksuzoglu, "Driver profiling using long short term memory (LSTM) and convolutional neural network (CNN) methods," *IEEE Trans. Intell. Transp. Syst.*, early access, Jun. 8, 2020, doi: 10.1109/TITS.2020.2995722.

[23] D. Li, Y. Wu, and J. Zhao, "Novel hybrid algorithm of improved CKF and GRU for GPS/INS," *IEEE Access*, vol. 8, pp. 202836–202847, 2020, doi: 10.1109/ACCESS.2020.3035653.

[24] W. Li, T. Logenthiran, and W. L. Woo, "Multi-GRU prediction system for electricity generation's planning and operation," *IET Gener., Transmiss. Distrib.*, vol. 13, no. 9, pp. 1630–1637, May 2019, doi: 10.1049/iet-gtd.2018.6081.

[25] A. Banik, C. Behera, T. V. Sarathkumar, and A. K. Goswami, "Uncertain wind power forecasting using LSTM-based prediction interval," *IET Renew. Power Gener.*, vol. 14, no. 14, pp. 2657–2667, Oct. 2020, doi: 10.1049/iet-rpg.2019.1238.