Ultra-Short Term Power Load Prediction Based on Gated Cycle Neural Network and XGBoost Models

Xiaojin Li¹, Yueyang Huang² and Yuanbo Shi¹
¹ College of Computer Communication and Engineering, Liaoning Shihua University, Fushun 113001, Liaoning, China
² School of Information and Control Engineering, Liaoning Shihua University, Fushun 113001, Liaoning, China
Email: 312953286@qq.com; yangmaokong@yeah.net; syb2011@yeah.net

Abstract. Load forecasting is an important guarantee for power system design, planning and operation. In order to further improve the accuracy of power load forecasting, a combined forecasting model based on gating recurrent neural network (GRU) and limit gradient lifting (XGBoost) is proposed. Firstly, according to the load data and the input structure of GRU and XGBoost models, the data are preprocessed, and the preprocessed data are input into the corresponding model respectively. Then, the model is weighted by entropy weight method to obtain the predicted value of the final combination model. Finally, the first mock exam is used to compare the combination forecasting model with the common forecasting models. The results show that the proposed method can effectively combine the advantages of the two models, and take into account the continuous time series and discontinuous feature variables, which is more accurate than the single model and the common forecasting models.

Keywords. GRU; XGBoost; combination model.

1. Introduction

Power load forecasting is a very important part of the power energy system. The balance between power generation and load demand must be maintained in the best state to avoid fatal interference to the grid due to overload [1]. From classic time series analysis [2] to recent machine learning algorithms, there are many research methods on load forecasting. Commonly used forecasting methods include forecasting methods based on time series, gray forecasting methods [3], regression analysis and so on. All of these methods have their limitations, and they are unable to dig deeper into the data, which affects the further improvement of the accuracy of the prediction results of their models.

Another branch of load forecasting is the use of artificial neural networks (ANN) models [4]. artificial neural networks have a better fitting effect on nonlinear data [5]. The commonly used network structure in ANN is multi-layer perceptron (MLP), and many documents have adopted the method of multi-layer perceptron, and achieved good prediction results. For example, in reference [6], an hourly load forecasting model based on ANN is proposed for household electricity consumption, and [7] uses a deep confidence network to predict the spatial load of the distribution system, which improves the accuracy of spatial load forecasting. The above models are prone to under-fitting in the load forecasting of long-term series. In recent years, long-short-term memory neural networks (LSTMs) are often used for model predictions with time series attributes because of the internal storage units. Reference [8] combines a deep neural network with LSTM to perform better than a single neural network. As a variant of LSTM, GRU can obtain similar effects as LSTM, but it uses fewer algorithm resources and is faster.
As an improved algorithm of GBDT, XGBoost has achieved very good results in many competitions [9]. In this paper, GRU and XGBoost are combined through entropy method, and the combined model is compared with a single model, and the prediction accuracy has been greatly improved.

2. Based on GRU Network Load Forecasting Model

2.1. GRU Network Model

Recurrent Neural Network (RNN) is easily affected by the weight too much [10]. In a model with a long-time span, it is prone to overfitting or underfitting. To solve this problem, LSTM introduces a logic control gate unit, to solve the problem of RNN gradient disappearing or exploding. LSTM has more parameters and gating units, so it takes a lot of time to adjust the parameters. The Gated Recurrent Unit (GRU) is optimized on this basis, combining the input gate and forget gate of LSTM into one update gate, and GRU does not have a cell structure like LSTM, which can take less time and save algorithm resources to achieve the same Similar effects of LSTM [11].

![Figure 1. GRU cell structure diagram.](image)

Figure 1 is the structure diagram of the GRU network, where $r_t$ is the update gate in the network structure, and the update gate controls the previous state input $h_{t-1}$, the weight of the state that is passed into the current state [12]. $z_t$ is the reset gate in the network structure. The update gate controls the reset gate to indicate how much the previous state is inserted into the current state candidate set $h_t$. As shown in equation (1) is the propagation formula of the update gate:

$$r_t = \sigma(W_r \cdot [h_{t-1}, x_t])$$  \hspace{1cm} (1)

where $\sigma$ is the activation function sigmoid, and $W_r$ is the learned parameter, $W_z$ is the weight of the reset gate input data. The propagation formula of the reset gate is as follows:

$$z_t = \sigma(W_z \cdot [h_{t-1}, x_t])$$  \hspace{1cm} (2)

The calculation formula of the final result:

$$y_t = \sigma(W_o \cdot h_t)$$  \hspace{1cm} (3)

2.2. Construction of Load Forecasting Model Based on GRU Network

Load changes are mainly affected by weather and time factors. Among them, the meteorological factors specifically include temperature and humidity, and the time characteristic factors are mainly holiday or rest days. This article takes into account the time and meteorological factors, and takes the 24 hourly load values of the highest temperature, lowest temperature, average temperature, relative humidity, and date type as the characteristics of a day, and enters all the characteristics of the previous $n$ days from $t-1$ to $t-n$. Value and load value on the $t$th day, the highest temperature, the lowest temperature, the average temperature, the relative humidity, and the 5 characteristic values of the date type are input together, a total of $r$ characteristic values are used as input, and the 24 hourly load data values on the $t$th day are predicted.
**Step1:** In order to reduce the error and training time, the load and temperature data need to be normalized:

$$x_i' = \frac{x_i - x_{\text{min}}}{x_{\text{max}} - x_{\text{min}}}$$  \hspace{1cm} \text{(4)}$$

$x_i$ and $x_i'$ are the values before and after normalization respectively, $x_{\text{min}}$ and $x_{\text{max}}$ are the minimum and maximum values of corresponding feature data respectively.

**Step2:** Through data analysis, it is found that the load data changes greatly during the rest days. This article finally quantifies the working days as 0 and the rest days as 1.

**Step3:** The humidity of the original data is relative data, so this article directly uses the relative humidity in the original data, and the range is between (0,1).

After the data preprocessing is completed, this paper constructs a multi-layer GRU network model as shown in figure 2, which mainly contains a three-layer network structure. The model inputs the feature variables into the model through the input layer, and then trains the model through the hidden layer of the model. The hidden layer contains several GRU network layers, and finally predicts that the data will be output through the fully connected layer Dense layer.

![GRU neural network model](image)

Figure 2. GRU neural network model.

### 3. XGBoost Load Forecasting Model

**3.1. XGBoost Algorithm**

XGBoost is a gradient enhancement framework based on decision trees. It is a portable and efficient platform. For a given data set with $n$ samples and $m$ features $D=(x_i,y_i)$, In the data set $|D|=n,x_i \in R^m$, $y_i \in R$. The integrated output model of the tree is shown below:

$$\hat{y}_i = \sum_{k=1}^{K} f_k(x_i), f_k \in F, F = \{ f_k(x_i) = w_q(x)\}(q : R^n \rightarrow T, w \in R^q)$$  \hspace{1cm} \text{(5)}$$

where, $F$ is the space of CART (regression tree), $x_i$ represents the feature vector of the $i$th data point, and $q$ represents the sample of each decision tree Structure, $T$ represents the leaf node on each tree, and each data sample is mapped to $T$ through the $q$ decision tree structure. $w$ represents all the scores obtained by each leaf node in the model scoring function, $w_q(x)$ is the scoring function of the sample $x$.

$$\hat{y}_i^{(t)} = \sum_{k=1}^{K} f_k(x_i) = \hat{y}_i^{(t-1)} + f_t(x_i)$$  \hspace{1cm} \text{(6)}$$

The above formula is the formula for fitting residual, $\hat{y}_i^{(t)}$ is the predicted value of the $i$th sample at the $t$th step, keep the previous prediction and add the new function $f_t(x_i)$.

The XGBoost model needs to optimize the objective function and continuously optimize the model to obtain the output model. The objective function is as follows:
\[
\text{Obj}(\theta) = \sum_{i} l(y_i, \hat{y}_i) + \sum_{k} \Omega(f_k)
\]  
(7)

\[
\Omega(f) = \gamma T + \frac{1}{2} \| w \|^2
\]  
(8)

In the above formula, \( l(y_i, \hat{y}_i) \) is the loss scoring function, \( \Omega(f_k) \) represents the regularization term of the model, where \( \gamma, \lambda \) is the penalty coefficient of the regularization term.

The iterative function of XGBoost based on the addition method is:

\[
\text{Obj}_{i+1}^{(t)} = \sum_{i} l(y_i, \hat{y}_i^{(t-1)}) + f_i(x_i) + \Omega(f_i) + \text{const} \tan t
\]  
(9)

Remove the constant term from the function to get a standard target function:

\[
\sum_{i} [g_i f_i(x_i) + \frac{1}{2} h_i f_i^2(x_i)] + \Omega(f_i)
\]  
(10)

3.2. XGBoost Predictive Model Construction

The model structure of the XGBoost model is different from that of the GRU neural network. It cannot output the load data of 24 hours a day at a time. Therefore, it is necessary to predict each prediction point in turn through the model. This paper selects the number of forecast points of the day, the load observations of the previous N days, weather data, day type and weather data of the day, whether it is a holiday as the input of the model, and the 24 hours load forecast value of the day as the output. Finally build the XGBoost model [13].

4. Combined Forecasting Model

4.1. Entropy Method to Determine the Optimal Weighted Combination

After obtaining the prediction results of the GRU neural network and the XGBoost model, we need to assign different weights according to the prediction accuracy of the two models at each moment [14]. The specific method used is the entropy method:

(1) Calculate the relative error weight of the \( i \)th model at time \( t \).

\[
p_{ei} = \frac{e_{ei}}{\sum_{i=1}^{n} e_{it}}, i = 1, 2, \ldots, T
\]  
(11)

where \( e_{it} \) represents the relative error of the \( i \)th model at time \( t \), and its calculation method is as follows:

\[
e_{it} = \begin{cases} 1, 0 \leq |(x_i - \hat{x}_i) / x_i| < 1 \\ 0, |(x_i - \hat{x}_i) / x_i| \geq 1 \\
\end{cases}
\]  
(12)

In the formula, \( x_i \) is the true value sequence at time \( t \), and \( \hat{x}_i \) is the predicted value sequence of the \( i \)th model at time \( t \).

(2) Calculate the information entropy value of the relative error of the \( i \)th model prediction:

\[
h_i = -k \sum_{t=1}^{T} p_{ei} \ln p_{ei}, i = 1, 2, \ldots, n, k = 1 / \ln T
\]  
(13)

(3) Calculate the coefficient of variation between the predicted value and the true value of each model separately:

\[
d_i = 1 - h_i, i = 1, 2, \ldots, n
\]  
(14)

(4) Calculate the weighting coefficient of each model separately:
\[ l_i = \frac{1}{n-1} (1 - \frac{d_i}{\sum_{i=1}^{n} d_i}), i = 1, 2, \ldots, n \]  
\[ (15) \]

(5) According to the weighting coefficient, the prediction results are weighted and combined to calculate the combined prediction value:
\[ \hat{x}_t = \sum_{i=1}^{n} l_i x_i \]  
\[ (16) \]

4.2. Combination Forecasting Process
The flow chart of the combined prediction is shown in figure 3. First, the original data is removed by data preprocessing, the data set is divided, and the training set and the test set are divided. The GRU neural network model and the XGBoost model are respectively established through the previously given model construction method, the range of hyperparameters is set, and the parameters are continuously adjusted according to the returned training results through the continuous iterative training of the model to achieve the best model prediction effect.

![Flow chart of combined forecast](image)

**Figure 3.** Flow chart of combined forecast.

5. Case Analysis

5.1. Experimental Data and Platform
The experiment uses a certain region of my country from January 1, 2012 to January 1, 2015 as the
experimental data set, and uses the maximum temperature, minimum temperature, average temperature, relative humidity, daily type and actual hourly load as input sample characteristics. This article is divided into training set and test set according to the ratio of 7:3.

This article is implemented in Python language. Deep learning neural network models such as GRU use the Keras framework. XGBoost uses the py-xgboost framework.

5.2. Experimental Evaluation Index
This paper selects the average absolute percentage error $E_{MAPE}$ as the main evaluation index, and chooses the root mean square error $E_{RMSE}$ as the auxiliary evaluation index. The formulas of $E_{MAPE}$ and $E_{RMSE}$ are as follows:

$$E_{MAPE} = \frac{1}{N} \sum_{i=1}^{N} \left| \frac{y_i - \hat{y}_i}{y_i} \right| \times 100\%$$

(17)

$$E_{RMSE} = \sqrt{\frac{1}{N} \sum_{i=1}^{N} (y_i - \hat{y}_i)^2}$$

(18)

Among them, $y_i$ represents the real load value at time t, and $\hat{y}_i$ represents the predicted value of the model at time t. The smaller the value of $E_{MAPE}$ and $E_{RMSE}$, the better the prediction effect of the model.

5.3. Selection of Model Parameters
This paper uses the grid search method to find the optimal parameter combination. The hidden layer of the GRU neural network model contains two layers, the learning rate is set to 0.01, the number of iterations is 100, and the optimization algorithm is Adam algorithm. The choice of XGBoost promotes its choice of gbtree, with a learning rate of 0.06; the maximum height of the tree is 5; the sample sampling ratio is 0.2; the leaf tree is 50; the number of iterations is 1500.

5.4. Different Input Sequence Length Affects Model Accuracy
By changing the length of the input sequence, analyze the impact of different input sequence length on the accuracy of the model, and the results are shown in table 1.

| Input sequence length (GRU,XGBoost) | $E_{MAPE}$ (%) | XGBoost | GRU-XGBoost |
|------------------------------------|----------------|---------|-------------|
| (34,34)                            | 3.46           | 2.78    | 2.63        |
| (92,34)                            | 3.32           | 2.78    | 2.57        |
| (208,34)                           | 2.85           | 2.78    | 2.36        |
| (208,92)                           | 2.85           | 2.85    | 2.45        |
| (208,208)                          | 2.85           | 2.89    | 2.48        |

It can be seen from table 3 that as the length of the input sequence increases, the accuracy of the prediction is improved, indicating that the gated recurrent neural network (GRU) has a better ability to mine time series information [15]. However, as the length of the input sequence increases, the error of the XGBoost model increases, indicating that the XGBoost model does not perform well in the prediction of long-term series and cannot fully explore the relationship between time series and data. Therefore, combining the prediction results of the two models can complement the advantages of the two models to a certain extent. The load data is greatly affected by the characteristic variables of whether it is a holiday or not, so the influence of whether it is a holiday on the forecast results is analyzed.
Table 2. Forecast error of different day types.

| Data type | Error analysis | GRU | XGBoost | GRU-XGBoost |
|-----------|----------------|-----|---------|-------------|
| holiday   | MAPE (%)       | 3.89| 2.96    | 2.87        |
|           | RMSE (kW)      | 185.25| 163.12 | 159.36     |
| workday   | MAPE (%)       | 2.56| 2.48    | 2.12        |
|           | RMSE (kW)      | 134.56| 130.5  | 124.23     |

It can be seen from table 4 that the prediction results of the GRU cyclic neural network on rest days are not good, with an average absolute percentage error of 3.89%, indicating that the GRU cyclic neural network is not effective in processing non-continuous time series data.

5.5. Analysis of Prediction Results of Different Models
This paper uses GRU neural network model, XGBoost model and GRU-XGBoost model to predict the load of one day (June 10, 2014) and a continuous week (June 16, 2014 to June 22, 2014). The experimental The prediction results are compared as shown in the figure:

Figure 4. Comparison of one day forecast results.

Figure 5. Comparison of one week forecast results.

It can be seen from figures 4 and 5 that the forecasting effect of the GRU-XGBoost combined model is better than the single forecasting model in one day and one week, indicating that compared with single model forecasting, the entropy method combination model can effectively improve the prediction accuracy of the model [16].

In order to better verify the combined model, this article chooses commonly used prediction models, such as multi-layer perceptron (MLP), BP neural network, and recurrent neural network (RNN) to make predictions. The prediction results are combined with the combined model (GRU-XGBoost) for comparison, the prediction results are as follows:
Figure 6. Comparison of prediction results of different models.

It can be seen from figure 6 that the predicted value curve of the GRU-XGBoost combined model is closest to the true value, and the prediction effect of the GRU-XGBoost combined model is better than that of other classic models. The prediction effect of the combined model can be seen through the above experiment. It is better than a single prediction model and commonly used prediction models. The following table shows the prediction errors of each model:

Table 3. Prediction error of different models.

| Prediction model | MAPE (%) | RMSE (kW) |
|------------------|----------|-----------|
| GRU              | 2.85     | 159.12    |
| XGBoost          | 2.78     | 156.73    |
| GRU-XGBoost      | 2.36     | 126.67    |
| BP               | 4.65     | 278.63    |
| NLP              | 4.23     | 254.35    |
| RNN              | 3.85     | 225.68    |

It can also be seen from the error score analysis table of the prediction results in Table 3 that the average percentage error $E_{MAPE}$ of the BP neural network is up to 4.65, and the $E_{MAPE}$ of the GRU and XGBoost models are similar, but the $E_{MAPE}$ of XGBoost is lower than that of GRU, indicating the accuracy of the XGBoost model. Better than GRU neural network model. However, the accuracy of the GRU-XGBoost model after combining the two through the entropy method is better than that of the single model, which verifies that the prediction effect of the GRU-XGBoost combined model is better than that of the single model.

6. Conclusion
This paper proposes a GRU-XGBoost model prediction method, which combines the characteristics of GRU suitable for long-term sequence prediction and the high prediction accuracy of XGBoost discontinuous features, which avoids the extreme errors of a single prediction model. In the subsequent forecasting situation, more attention should be paid to the specific application of load forecasting in those fields, such as industry, agriculture and commerce, and more useful characteristic information can be mined through specific backgrounds. In the future, it can be considered on the basis of big data platforms.

References
[1] Esteves G R T, Bastos B Q, Cyrino F L, et al. 2015 Long term electricity forecast: A systematic review Procedia Computer Science 55 549-558.
[2] Hagan M T and Behr S M 1987 The time series approach to short term load forecasting *IEEE Transactions on Power Systems* 2 (3) 785-791.

[3] Wang Y L, Wu M K, Zhou Z Q, et al. 2017 Quantitative analysis model of power load influencing factors based on improved grey correlation degree *Powergrid Technology* (06) 76-82.

[4] Park S, Ryu S, Choi Y, et al. 2015 Data-driven baseline estimation of residential buildings for demand response *Energies* 8 (9) 10239-1025.

[5] Song Z W 2020 *Research on the Method of Power Load Forecasting Based on Deep Learning and Wavelet Transform* (Zhejiang University).

[6] Rodrigues F, Cardeira C and Calado J M F 2014 The daily and hourly energy consumption and load forecasting using artificial neural network method: A case study using a set of 93 households in Portugal *Energy Procedia* 62 220-229.

[7] Liang R, Yang B, Ma R Z, et al. 2018 Spatial load forecasting of distribution system using multi-source information and deep confidence neural network *Power Construction* 39 (10) 21-28.

[8] Zhang X Y, Niu P H and Gao F 2018 VAD algorithm based on DNN-LSTM *Journal of Tsinghua University (Natural Science Edition)* 58 (5) 509-515.

[9] Chen J Q, Yang J J and Lou Z B 2019 Study on a new short-term load forecasting model based on XGboost algorithm *Electrical Measurement and Instrumentation* 56 (21) 23-29.

[10] Li R C, Zhu F, Zhu Y L, et al. 2018 Short term load forecasting of power system based on recurrent neural network combined with constrained Boltzmann machine *Power System Protection and Control* 46 (17) 92-97.

[11] Kong W, Dong Z Y, Jia Y, et al. 2019 Short-term residential load forecasting based on LSTM recurrent neural network *IEEE Transactions on Smart Grid* (S1949-3053) I (10) 841-851.

[12] Wang Z, Zhao B, Ji W, Gao X and Li X 2019 Short-term load forecasting method based on GRU-NN model *Automation of Electric Power Systems* 43 (05) 53-58.

[13] Chen Z Y, Liu J B, Li C, et al. 2020 Short term power load forecasting based on combination model of LSTM and XGboost *Grid Technology* 44 (2).

[14] Jia Y L, Gong Q W, Zhan J S and Li J X 2016 Short term power load combination forecasting method based on efficiency and relative entropy *Journal of Electric Power Science and Technology* 31 (02) 3-9.

[15] Wei N, Hu Z J, Yue J P, Du Y X and Qi Q 2020 Short term load forecasting based on long-term memory network and lightGBM combination model *J/OL. Power System Automation* 1-9.

[16] Tan M, Yuan S, Li S, et al. 2020 Ultra-short-term industrial power demand forecasting using LSTM based hybrid ensemble learning *IEEE Transactions on Power Systems* (S0885-8950) 35 (4) 2937-2948.