

Application of SRNN-GRU in Photovoltaic power Forecasting

Shiyan Liu¹, Xiaoguang Hao¹, Zhengji Meng¹, Jianfeng Li¹, Tongfei Cui¹, Le Wei²

¹ Electric Power Research Institute of State Grid Hebei Electric power Company, Shijiazhuang, Hebei, 050051, China

² Xian WuChang Power Technology company, Shanxi, 710048, China

Abstract. Short-term photovoltaic power forecasting is of great significance for maintaining the security and stability of the power grid and coordinating the utilization of resources. As one of the Deep Learning Methods, Recurrent Neural Network (RNN) is widely used in time series prediction but lacks the ability of parallel computing. With good prediction effect, RNN is faced with the problem of long training time. In this paper, Sliced Recurrent Neural Network (SRNN) is applied to PV power prediction to guarantee the ability of parallel computing. The research result shows that compared to other commonly used models, SRNN can greatly speed up the training of Deep Learning Network with over 4 times higher training speed of the application of PV power prediction than that of ordinary RNN structure like LSTM and GRU. The accuracy of SRNN model is also improved by 0.1102 mae, which is significantly ahead of the others, as its parallel structure causes the more efficient parameter update, thus achieving ideal effect in PV prediction.

1 Introduction

Photovoltaic power generation is a sustainable and renewable clean energy source [1]. Since 2007, the photovoltaic market has been developing continuously, and many factors including its power generation, power transmission process, maintenance and loss have become an indispensable part of the smart grid construction[2]. With the increase of PV power generation, the stability and security of the power grid are challenged [3]. Therefore, accurate prediction of PV power is of great significance to the stability of photovoltaic generation [4].

Wavelet neural network is established in Reference [5]. Moreover, after applying grey system, the prediction accuracy is further improved. At present, deep learning is one of the most popular research fields, which has been applied in the field of PV power prediction and achieved remarkable results [6]. To predict photovoltaic power, the DBN (Deep Believe Network) model is established in reference [7]. LSTM (Long short term memory), as a cyclic unit of deep learning cyclic neural networks, is often applied in related works [8]. Reference [9] uses the isolated forest method to process the data, and the processed data is input into LSTM to predict the photovoltaic power. The CNN (Convolutional Neural Network) is concatenated with the cyclic neural network RNN (Recurrent Neural Networks) structure in reference [10]. The hybrid model achieves ideal prediction accuracy, and the cyclic unit of that is LSTM. RNN is a very effective network structure for dealing with photovoltaic power data [11]. The QRNN, proposed by combining the rapidity of CNN with RNN ability of memory is 16 times faster than the simple RNN [12]. The presentation of recurrent cycle unit SRU makes RNN training faster, 5-10 times faster than LSTM [13].

While many studies try to speed up RNN, but not change the structure. RNN still requires step by step operation and lacks parallel capabilities. However, the main problem of RNN is that it can not do parallel problem processing, and the sequence information can not be processed before the completion of the processing in previous sequence. The SRNN (Sliced Recurrent Neural Networks) structure proposed in [14] solves the problem that RNN can not be parallel, and greatly accelerates the training speed of the model.

This paper attempts to build a SRNN model and evaluate its performance by comparing with some methods like simple RNN, LSTM and GRU. The second section shows the introduction of the related Algorithm, including GRU and SRNN. And the data slice method is given in section 3. Forecast model building is in section 4. The results are discussed in section 5.

2 Introduction to Algorithm

2.1 Gate Recurrent Unit

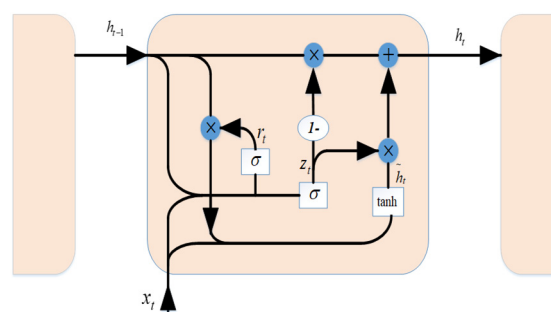


Figure 1. GRU Structure

Corresponding author's e-mail: liusy9@he.sgcc.com.cn

GRU (Gate Recurrent Unit) is one of the CNN (Recurrent Neural Network). GRU structure is shown in Figure 1. As a kind of recurrent unit that solves gradient explosion and gradient disappearance, it is more simplified than LSTM and has fewer parameters, so it has faster training speed, but works as well as LSTM.

The updated expression is:

$$h_t = (1 - z_t) \otimes h_{t-1} + z_t \otimes \tilde{h}_t \quad (1)$$

$$\tilde{h}_t = \tanh(W \otimes [r_t \otimes h_{t-1}, x_t]) \quad (2)$$

$$z_t = \sigma(W_z \otimes [h_{t-1}, x_t]) \quad (3)$$

$$r_t = \sigma(W_r \otimes [h_{t-1}, x_t]) \quad (4)$$

\otimes is matrix corresponding elements multiplication, and $+$ is matrix addition. W is the network weights, x_t is the input value of the GRU unit, and h_t is the output value of the unit. σ is sigmoid function.

2.2 Sliced Recurrent Neural Networks

A new structure ‘‘Sliced Recurrent Neural Networks’’ is proposed in paper [9] ‘‘Sliced Recurrent Neural Networks’’. After adding parallel structures to the RNN, the restriction that RNN needs to wait for the next step is broken up, and SRNN structure is used to achieve parallelization of photovoltaic power prediction of RNN. The greatest desirability of SRNN lies in the parallel computing power, and the longer the sequence, the more obvious the advantages.

SRNN structure diagram is shown in Figure 2. By dividing the whole sequence and computing the segmentation sequence separately, the parallel processing sequence is realized, instead of training the whole sequence directly as input as RNN.

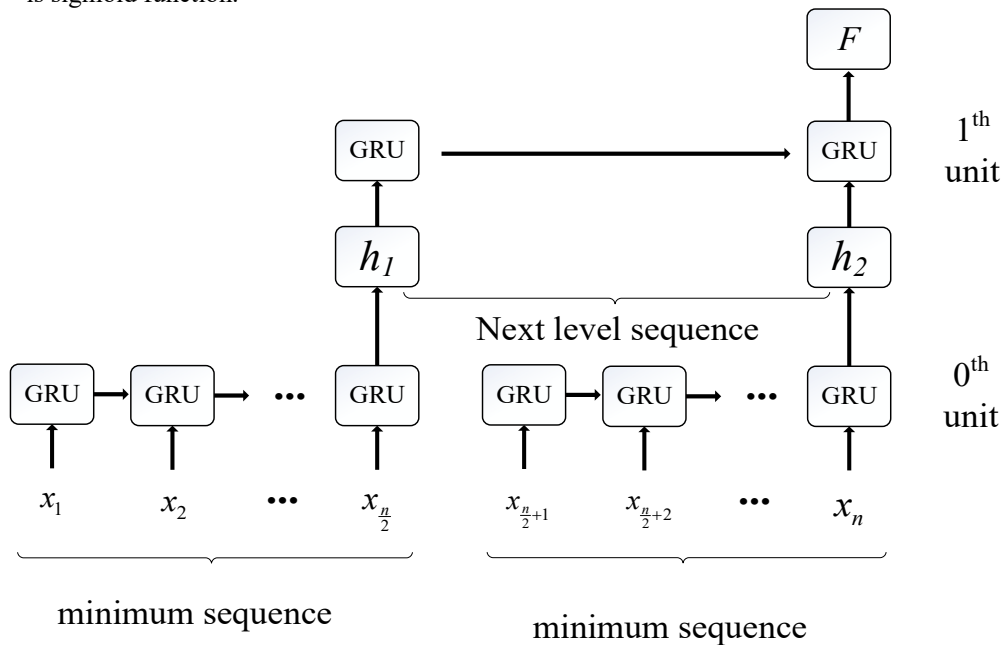


Figure 2. SRNN Structure

the total sequence is:

$$X = [x_1, x_2, \dots, x_n] \quad (5)$$

The sequence is divided according to the set slicing number N . N , the slicing number, is equal fraction of the sequence. K is the number of slicing operations, that is to operate the sequence several times with the same number cuts. When the number of cuts is K and the number of GRU layers is $K+1$, the number of the minimum sequence after segmentation which is the sequence groups pass into layer 0 cycle units is:

$$\begin{aligned} GRU(input) = & [GRU(X_1), \\ & GRU(X_2), \dots, GRU(X_{N^K})] \end{aligned} \quad (6)$$

In the figure.2 N equals to two, K equals to 1. X_i is the minimum sequence of the input GRU units. The other sub-sequences are passed into GRU in the same way as the

minimum sequence with the length of $\frac{n}{N^K}$. The number of cuts can be more than once, when the number of slicing operations K increase once, that is to say, every subsequence, such as X_1, X_2 to X_N , which has been slice once, is treated as the original sequence to perform the same slicing operation again.

Subsequence X_i for each incoming GRU is:

$$\begin{aligned} GRU(X_i) = & GRU([x_{(i-1) \times \frac{n}{N^K} + 1}, \\ & x_{(i-1) \times \frac{n}{N^K} + 2}, \dots, x_{i \times \frac{n}{N^K}}]) \end{aligned} \quad (7)$$

Partitioned sequences pass layer by layer, the next layer GRU recurrent unit passes to the next layer with h which is the output sequence of the previous layer as input. The first layer GRU unit is the layer zero, the hidden state

sequence group which is accepted by GRU unit in layer M is:

$$\begin{aligned} H^M &= [H_1^M, H_2^M, \dots, H_{N^{K-M}}^M] \\ &= [h_1^M, h_2^M, \dots, h_{\frac{n}{N^{K-M}}}^M] \end{aligned} \quad (8)$$

h^M represents that the hidden state h is received by GRU in layer M, H^M represents the hidden state sequence received by GRU in layer M. Finally through the pyramid structure to the final layer GRU unit, the final hidden state F is obtained.

$$\begin{aligned} F &= GRU^K(H^K) \\ &= GRU^K([h_1^K, h_2^K, \dots, h_n^K]) \end{aligned} \quad (9)$$

Its nested parallel network structure can be highly parallelized.

3Data processing

3.1Photovoltaic data pre-treatment feature data processing

The data of a northwest power station with 5-minute interval from 2018 are experimental data, including meteorological data and power data. The covariance correlation coefficient between meteorological data and power is shown in Table I. Expression of covariance correlation coefficient is:

$$r = \frac{\frac{1}{m} \sum_{i=1}^m (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\delta_x} \sqrt{\delta_y}} \quad (10)$$

respectively represents the meteorological data selected according to the variance of x and y , including wind speed, irradiance, humidity, etc.

Table 1. Correlation coefficient between each photovoltaic parameters

Correlation coefficient	air temperature	Relative humidity	Air pressure	Wind speed	wind direction	Direct radiation	Total radiation	Scattered radiation	Component temperature
power	0.3167	-0.3042	-0.0972	0.3125	0.2337	0.0704	0.9903	0.8358	0.4067

Table 1 shows the correlation coefficients between each factor and the power when the power is generated during the day (delete the data with the zero power). It can be seen from Table that irradiance is the most related factor to power, but in order to make the model more accurate, all data needs to be input.

Outliers need to be replaced and the data with more outlier is eliminated to improve prediction accuracy by iforest, a data preprocessing method provided by python Sklearn Toolkit. Because of the large numerical difference between the data, the data should be standardized for each input to equalize the influence of each input on the model. Scaling the data in proportion increases the training speed and converges faster. Preprocessing and StandardScaler are used to standardize the data so that they all fall on the [-1, 1] interval.

3.2Data segmentation

The data is sliced to fit the network due to the fact that there is data can not be divided with no remainder and data can not be evenly divided. This paper uses N^{K+1} data before the t moment as an input data window. As time t increases, the data window slides backwards over time. The data in the data window is sliced each time to adapt to

the network structure, and the power value of the t time is taken as the output of the data window at the t moment. Data segmentation makes the original sequential and coherent input time series into batch sequential input network, which is essentially the localization of the entire network and multiple applications to the sequence. For example, as shown in Figure 3, a time series with a length of 12 can be divided into 3 parts, each of which is a minimum sub-sequence with a length of 4, and each minimum sub-sequence is regarded as a divided minimum sequence one by one Input to the SRNN network. The output value of the network continues to be trained in the SRNN network as the next-generation minimum sequence, but the network size gradually decreases layer by layer. When the final output result is only one, the training is completed once.

For example, $N = 3 K = 2$, that is to say, 27 sequences before the t moment are used as inputs to predict the output of the t moment. The data is sliced by two times by time dimension. Input data changes from the original (27, 10) two-dimensional to (3, 3, 3, 10) four-dimensional, and the minimum sequence is a two-dimensional input of (3, 10), which is passed in parallel to the zero in the three-layer GRU unit, and then the data window slides with the increase of t until the last moment.

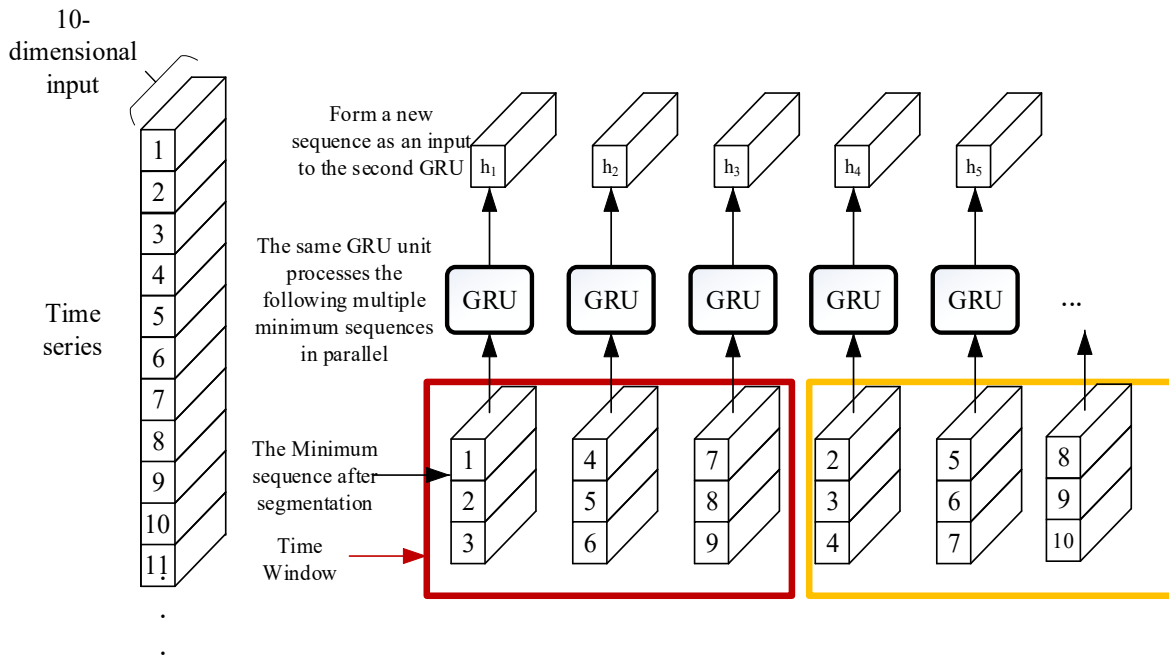


Figure 3. The Schematic diagram of data input network

4 Photovoltaic prediction model based on SRNN-GRU

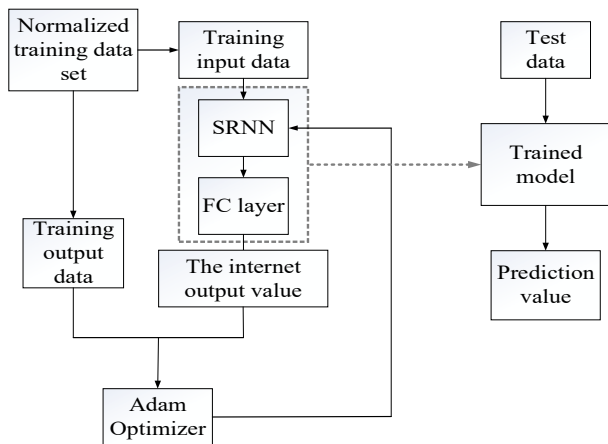


Figure 4. Flow chart of the SRNN model.

Figure 4 is the flow chart of the model. First, the network structure should be determined. Because there are no specific methods and theories, the structure of the model network is generally determined by experience and

approximate testing. The SRNN network in which $N=3$, $k=2$ is finally selected after testing and then connect to a FC fully-connected layer. The training data set is then processed to adapt to the network structure as an input that can be applied to the network. Iterative algorithm is Adam algorithm and its network weights are continuously iteratively trained. Finally, epochs number is 20.

The following uses data from a northwestern power station at intervals of 5 minutes from 2018 for validation. The first 80% of the data is used as the training set, the middle 10% is used as the validation set, the last 10% is used as the test set. The data is sliced in a $N = 3$, $K = 2$ way. Because $K = 2$, the GRU has a three-layer structure. The number of GRU units in the three layers is 300, and the last two layers are the FC fully connected layers with the activation function ReLU. The activation function of the cycle time step is sigmoid, and the optimizer selects the Adam optimizer. As a comparison, the RNN structure adopts a three-layer recurrent unit in series, and other configurations are the same as SRNN.

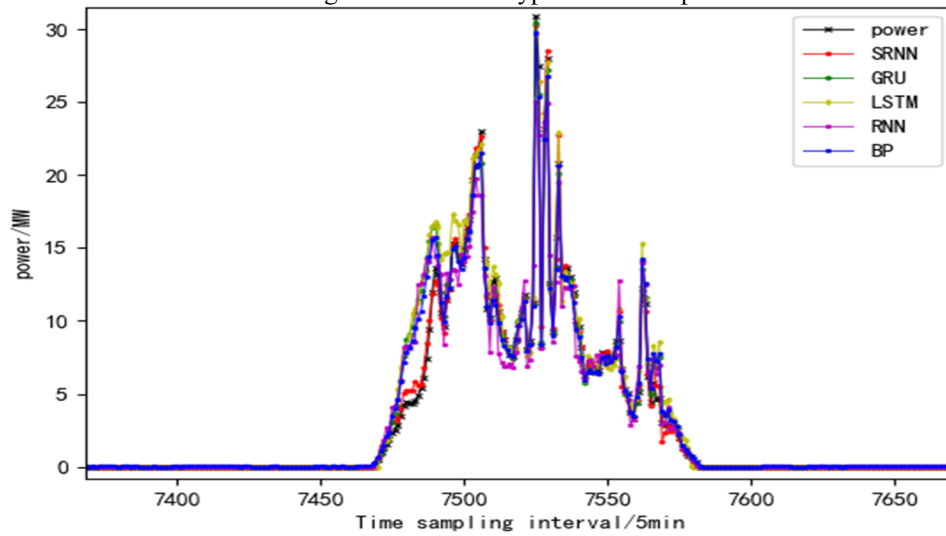
5 Experimental Simulation and Results Analysis

Table 2. Performance comparison between various models

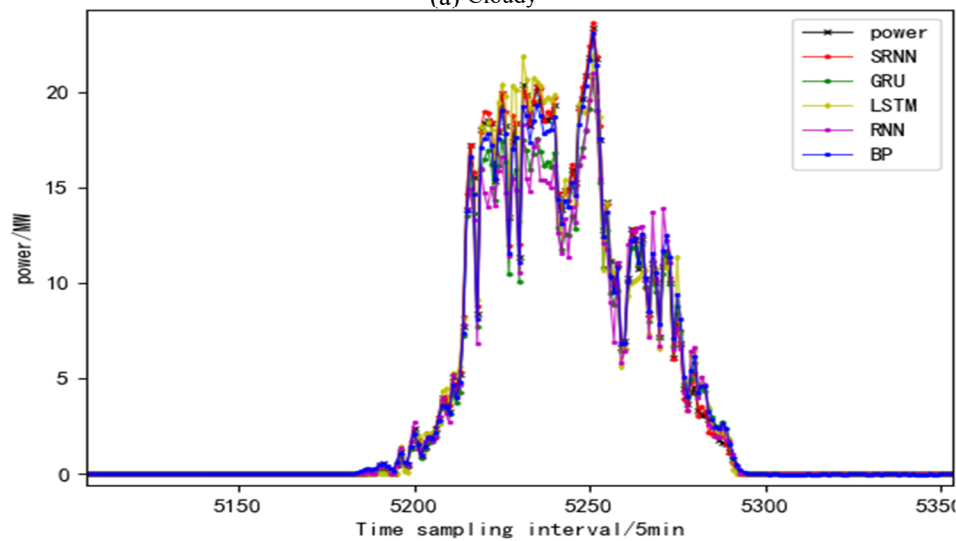
Model	Epochs/ time	Mae	Mse
BP	2s	0.4552	0.9955
RNN	60s	0.5673	2.3149
LSTM	117s	0.3485	0.6037
GRU	100s	0.3813	0.6939
SRNN-GRU	27s	0.1102	0.1145

The experimental results are shown in Table 2. RNN has a fast training speed, but the network effect is the worst. Due to its own structural defects, it cannot transmit the information of the long sequences information. The BP neural network has the fastest speed due to its simple structure. Although the BP neural network performs better than RNN, its accuracy can't match others like GRU and LSTM with memory forgetting function. Since LSTM has one more gate than GRU, the result would be better, but with the addition of network parameters, the training process would be slower than GRU. Although the same

GRU unit, the GRU units using the SRNN structure not only improves speed, but also improves the accuracy of the model. This is because the pure GRU network is a series connection of GRU units, and the network parameter update needs to be similar to the RNN structure. It is transmitted layer by layer, but the GRU network using the SRNN structure is a cascade of GRU units, so the network parameters do not need to be updated layer by layer, so the speed accuracy of the model is very objective. This shows the superiority of SRNN. Figure 5 shows some typical weather prediction.



(a) Cloudy



(b) Rainy

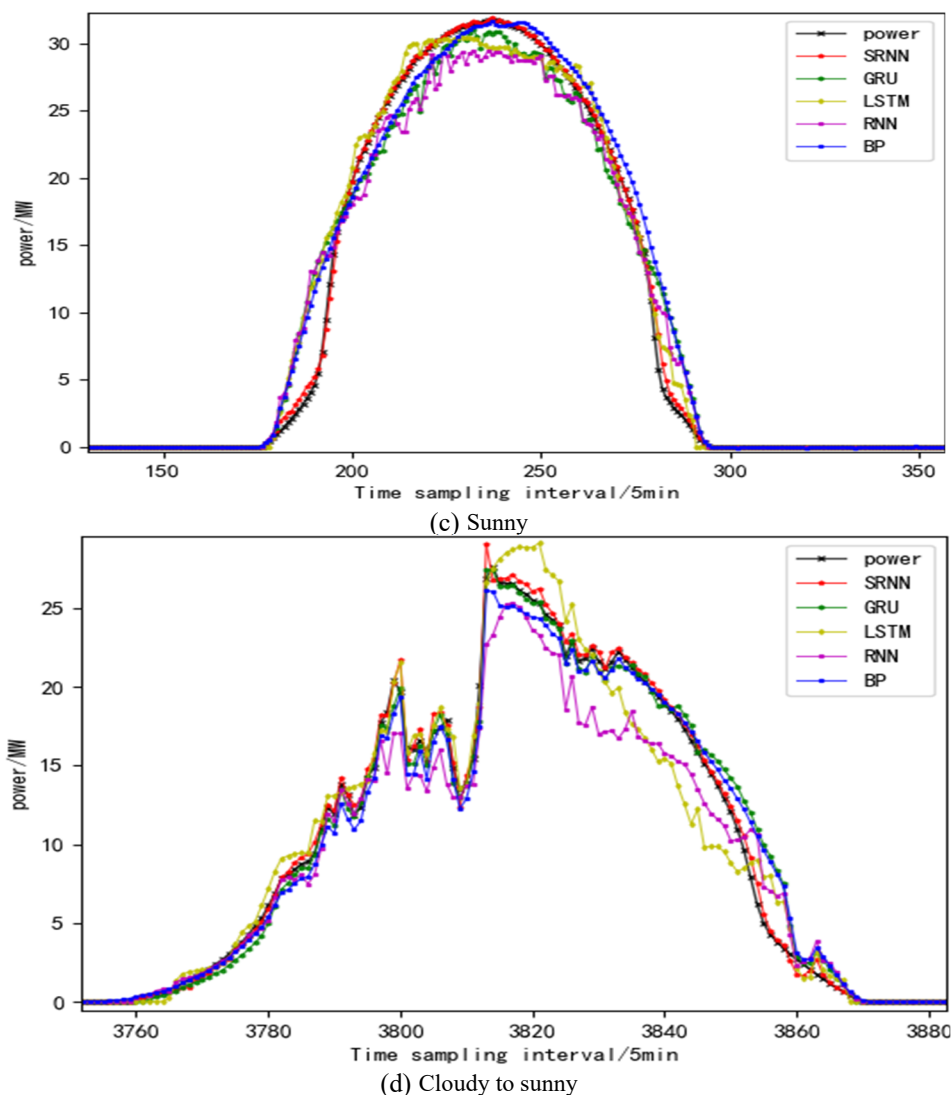


Figure 5. The predict result of each model.

It can be seen from Figure 5 that the SRNN model fits best. Due to the slow training speed, slow convergence, and stagnation of parameter update, the LSTM model has poor model accuracy. Compared with the SRNN model, the GRU model also has the same problem, but because its training speed is slightly faster than the LSTM and the GRU cycle unit is also used, its model accuracy is acceptable. The RNN network has no memory unit and the sequence length is too long, and the RNN cannot remember too much sequence information, so its model performs the worst. The shape of the predicted value of the BP neural network model is similar with the original power value, but the predicted value is overall biased.

6 conclusion

This paper considers the characteristics of short-term photovoltaic power generation and seasonal photovoltaic power generation, then proposes a prediction model based on SRNN structure. In this paper, the recurrent neural network with SRNN structure is used to predict the photovoltaic power. Due to the parallel feature of the SRNN structure, this model not only has faster training

speed than other models, but also has a much better prediction accuracy than the RNN structure models. Therefore, SRNN achieves good expectations and provides a solution for rapid training model.

References

1. Li Jianlin, Ji Tianming, Kong Lingda, et al. Span Determining of Photovoltaic Generation Data Mining[J]. TRANSACTIONS OF CHINA ELECTROTECHNICAL SOCIETY, 2015, 30(14): 450-456(in Chinese).
2. S. C. Kaushik, V. S. Reddy, S. K. Tyagi Energy and exergy analyses of thermal power plants: a review Renew. Sustain. Energy Rev., 15 (4) (2011), pp. 1857-1872.
3. Gong Yingfei, Lu Zongxiang, Qiao Ying, et al. An Overview of Photovoltaic Energy System Output Forecasting Technology[J]. Automation of Electric Power Systems, 2016, 40(04): 140-151(in Chinese).
4. A. Saez-de-Ibarra, A. Milo, H. Gaztañaga, V. Debusschere, and S. Bacha, "Co-optimization of storage system sizing and control strategy for

- intelligent photovoltaic power plants market integration, ” *IEEE Trans. Sustain. Energy*, vol. 7, no. 4, pp. 1749–1761, Oct. 2016.
5. Ding M, Liu Z, Bi R, Zhu W. Photovoltaic Output Prediction Based on Grey System Correction-Wavelet Neural Network[J]. *Power System Technology*, 2015, 39(09):2438-2443(in Chinese).
 6. H. Wang, H. Yi, J. Peng, G. Wang, Y. Liu, H. Jiang, and W. Liu, “Deterministic and probabilistic forecasting of photovoltaic power based on deep convolutional neural network, ” *Energy Convers. Manage.*, vol. 153, pp. 409–422, Dec. 2017.
 7. F. Xu, Y. Tian, Z. Wang and J. Li, "One-day Ahead Forecast of PV Output Based on Deep Belief Network and Weather Classification, " 2018 Chinese Automation Congress (CAC), Xi'an, China, 2018, pp. 412-417, doi: 10.1109/CAC.2018.8623312.
 8. K. Yan, X. Wang, Y. Du, N. Jin, H. Huang, and H. Zhou, “Multi-step shortterm power consumption forecasting with a hybrid deep learning strategy, ” *Energies*, vol. 11, no. 11, p. 3089, Nov. 2018.
 9. Zhang Q, Ma Y, Li G, Ma J, Ding J. Applications of Frequency Domain Decomposition and Deep Learning Algorithms in Short-term Load and Photovoltaic Power Forecasting[J]. *Proceedings of the CSEE*, 2019, 39(08):2221-2230+5(in Chinese).
 10. Jian X, Gu H, Wang Z. A short-term photovoltaic power prediction method based on dual-channel CNN and LSTM[J]. *Electric Power Science and Engineering*, 2019, 35(05):7-11.
 11. Atsushi Yona, Tomonobu Senjyu, Toshihisa Funabashi. Decision Technique of Solar Radiation Prediction Applying Recurrent Neural Network for Short-Term Ahead Power Output of Photovoltaic System[J]. *Smart Grid and Renewable Energy*, 2013, 04(6):32-38.
 12. James Bradbury, Stephen Merity, Caiming Xiong, and Richard Socher. 2017. Quasi-recurrent neural networks. In *Proceedings of the International Conference on Learning Representations*.
 13. Tao Lei, Yu Zhang, and Yoav Artzi. 2017. Training RNNs as Fast as CNNs. arXiv preprint arXiv:1709.02755.
 14. Yu, Zeping, and Gongshen Liu. "Sliced Recurrent Neural Networks. " *Proceedings of the 27th International Conference on Computational Linguistics*. 2018.