An Ultra-Short-Term and Short-Term Wind Power Forecasting Approach Based on Optimized Artificial Neural Network with Time Series Reconstruction

2022 4th International Conference on Smart Power & Internet Energy Systems (SPIES) | 978-1-6654-8957-7/22/$31.00 ©2022 IEEE | DOI: 10.1109/SPIES55999.2022.10082352

Lihan Zha
WeiYang College
Tsinghua University
Beijing, China
clh20@mails.tsinghua.edu.cn

DongXiang Jiang
Department of Energy and Power Engineering
Tsinghua University
Beijing, China
jiangdx@tsinghua.edu.cn

***Abstract*—Wind power forecasting (WPF) is crucial for grid dispatch and effective collection of wind power. In recent years, deep learning such as multi-layer perception (MLP) has been widely adopted in WPF. This paper proposes a novel ultra-short-term and short-term WPF approach based on optimized artificial neural network (ANN) with time series reconstruction (TSR) method. Z-scored method is adopted to preprocess the dataset measured from a wind farm. TSR is proposed and used to generate model inputs. An optimized ANN-based model with MLP architecture is introduced, and momentum algorithm and decaying learning rate are proposed for model optimization. Repeated trainings are performed to obtain the optimal model. The proposed approach is validated on the test case of a wind farm in China. The results obtained prove that the optimized ANN-based model and TSR can both effectively improve the accuracy of ultra-short-term and short-term WPF.**

Keywords—artificial neural network, multi-layer perception, time series reconstruction, momentum algorithm, wind power forecasting

I. Introduction

Wind power (WP) is considered one of the most appealing renewable energy sources today in face of the global effort to combat global warming and energy shortage. In many European and Asian countries such as Denmark, England, China, and India, wind power has undergone an impressive pace of development over recent years [1]. It is estimated that the combined capacity of WP worldwide has reached 94GW, corresponding to a growth rate of 12% [2].

Despite its low carbon footprint, one important drawback of WP roots from its intermittent and fluctuating nature, which imposes instability and heavy load to the electricity grid. It is thus very important to carry out accurate wind power forecasting (WPF). The goal is to formulate the complex functional relationship between WP and other parameters including wind speed, wind direction, weather conditions, wake effect, wind turbine characteristics, etc. [3], so that WP in a near future can be predicted through measured parameters and the power network can be dispatched ahead of time. Among all the parameters, wind speed is considered as the most significant [4]. In this paper, we consider wind speed as the key parameter to achieve reduced model complexity while maintaining good performance.

The many current approaches for WPF include various artificial intelligence (AI) methods and statistical methods [5], and notably the development of AI has greatly benefited the advancement of WPF [6]. Support vector machines (SVM) [7], artificial neural network (ANN) [8], genetic algorithm (GA) [9], and evolutionary algorithms (EA) [10] are representative artificial intelligence methods. Probabilistic forecasting [11], persistence method (PM) [12], Kalman filtering (KF) [13], auto-regressive integrated moving average (ARIMA) [14] are representative statistical methods. Among the various methods, ANN is proved to have good performance in short-term predictions [15].

Compared to traditional WPF methods with ANN, this paper introduces deep learning techniques and proposes a novel optimized ANN model with multi-layer perceptron (MLP) architecture. This work, compared to prior study, adopts momentum algorithm to optimize stochastic gradient descent (SGD), and introduces decaying learning rate during training process to improve the network's performance. Time series reconstruction (TSR) is implemented to generate inputs of the model. Both optimized ANN-based model and TSR are proved to effectively improve the forecasting accuracy on a 1-month dataset.

This paper is organized as follows. Section II presents detailed methodology of the study. Section III & IV introduces the procedure to determine MLP architecture and parameters for ultra-short-term WPF and short-term WPF, and discusses results obtained by different optimizing methods. Section V outlines main conclusions of this paper.

II. Methodology

A. Description and Preprocessing of Data

The data used is acquired from wind farms in Jiangsu, China. In the dataset, provided data are wind power (WP) and wind speed (WS) measurement data from 2015/10/1 0:00 to 2015/10/31 23:59. Both WP and WS are measured every 30 seconds in time series format, and each gives 74881 measurements in total neglecting missing observations. The WP and WS data have been normalized to range [0, 1], and the original data is shown in Fig. 1.

Z-score method is first utilized to eliminate anomalous measurements as a preprocessing step. The range [0, 1) is divided into n equal intervals of length $1/n$:

$$Intervals = \{[x_i, x_{i+1}) \mid x_i = \frac{i}{n}, i \in \{0, 1, \ldots, n-1\}\} \quad (1)$$

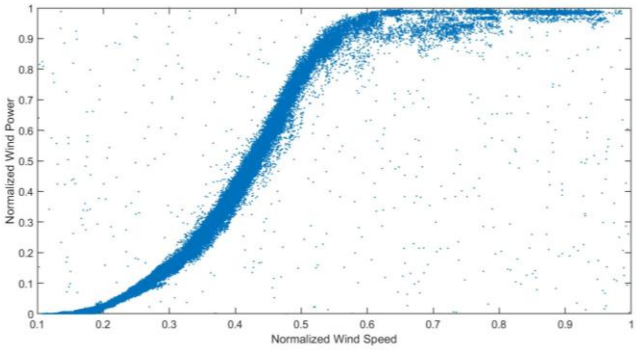

Fig. 1. Wind power and wind speed data samples before preprocessing.

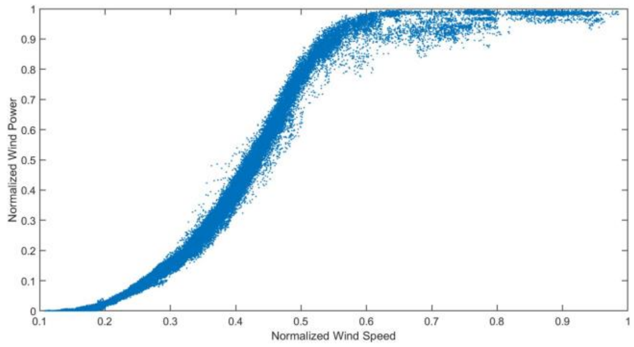

Fig. 2. Wind power and wind speed samples after preprocessing.

The normalized wind speed (NWS) of the samples are then binned into the intervals and form corresponding data sets:

$$X_i = \{x_{i,k} \mid x_{i,k} \in [x_{i-1}, x_i]\} \tag{2}$$

The corresponding samples of normalized wind power (NWP) subsequently form the sets as:

$$Y_i = \{y_{i,k} \mid f^{-1}(y_{i,k}) \in [x_{i-1}, x_i]\} \tag{3}$$

where $f^{-1}(\cdot)$ describes the inverse function that gives the corresponding $x_{i,k}$ for each $y_{i,k}$. Then, the mean μ_i and standard deviation σ_i of each set Y_i are calculated. According to Z-score theorem,

$$z_{i,k} = (y_{i,k} - \mu_i)/\sigma_i \tag{4}$$

If $z_{i,k}$ is larger than an adjustable constant c_i, sample set $(x_{i,k}, y_{i,k})$ should be eliminated from the original data set. Traversing the domain of i with multiple trials and adjustments, a cleaner data set can be obtained, as shown in Fig. 2.

B. *Principles of ANN-Based Model and Algorithm*

Artificial neural network (ANN), as it was so named, imitates the process of signal transmission between neurons in a biological nervous system to process information. In this paper, MLP architecture is utilized as a class of ANN. Fig. 3 depicts general architecture of MLP. It is composed of an input layer, a number of hidden layers, and an output layer, and each layer contains multiple neurons. Neurons of two adjacent layers are fully connected.

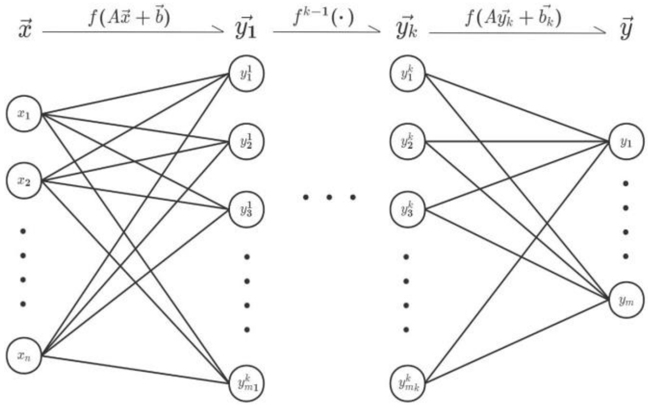

Fig. 3. General architecture of multi-layer perception model based on ANN

Signals are passed in the forward direction, which means that neurons in the next layer receive all signals from neurons in the previous layer. All signals are processed during transmission according to function f:

$$\vec{y} = f(A\vec{x} + \vec{b}) \tag{5}$$

where $\vec{y} = [y_1, y_2, \dots, y_n]^T$ are the signals received by the k^{th} layer, $\vec{x} = [x_1, x_2, \dots, x_m]^T$ are the signals released by the $(k-1)^{th}$ layer, $\vec{b} = [b_1, b_2, \dots, b_m]^T$ is the bias, $A = [\overrightarrow{w_1}, \overrightarrow{w_2}, \dots, \overrightarrow{w_m}]$ is the weight matrix, and $f(\cdot)$ is a non-linear activation function. Here, we use

$$f(x) = 1/(1 + \exp(-x)) \tag{6}$$

also known as the Sigmoid function. The advantage of Sigmoid function is that it changes rapidly in range [0, 1] so that it can readily distinguish data with values between 0 and 1. As both NWP and NWS values are normalized to [0, 1], utilizing Sigmoid function can amplify the differences between the samples and allow better characterization of the data.

ANN is powerful for information processing because it is a combination of non-linear and linear transformations with massive number of adjustable parameters. Its flexible structure and universal approximation capability in theory allows it to determine any functional relationship between the outputs and inputs.

The core of using ANN is parameter tuning. Weights and biases are initially set as random values and then tuned with certain algorithm through the training process. Different algorithms have been proposed, including cascade correlation (CC) and backward propagation (BP) [16]. The goal of the tuning process is to adjust weights and biases to reduce the discrepancy between prediction outputs and targets. The algorithm employed here is BP with optimized stochastic gradient descent (SGD) method. In each training epoch, the network calculates an output based on input and current parameters. By comparing the output with the correct answer, namely target, we can calculate the error, or loss, defined as:

$$E = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(output - target)^2} \tag{7}$$

where n is the number of output neurons. The goal of the model is to make predictions as accurate as possible, that is to minimize E. The weights and biases are then altered according to the following equations to reduce the loss E:

$$w_k = w_{k-1} - \eta \frac{\partial E}{\partial w_{k-1}} \tag{8}$$

$$b_k = b_{k-1} - \eta \frac{\partial E}{\partial b_{k-1}} \tag{9}$$

where w_k, b_k refers to a weight and bias component in training epoch k, and η is called learning rate. This algorithm is called traditional SGD.

Here we propose an optimized SGD with momentum algorithm, given by following equations:

$$v_{w,k} = \beta v_{w,k-1} + \frac{\partial E}{\partial w_{k-1}} \tag{10}$$

$$v_{b,k} = \beta v_{b,k-1} + \frac{\partial E}{\partial b_{k-1}} \tag{11}$$

$$w_k = w_{k-1} - \eta(k) v_{w,k-1} \tag{12}$$

$$b_k = b_{k-1} - \eta(k) v_{b,k-1} \tag{13}$$

$v_{w,k}$ and $v_{b,k}$ are introduced to calculate the exponential average of w_k and b_k,

$$v_{w,k} = \sum_{i=0}^{k-1} \beta^i w^i \tag{14}$$

$$v_{b,k} = \sum_{i=0}^{k-1} \beta^i b^i \tag{15}$$

The advantage of this momentum algorithm versus traditional SGD is that utilizing exponential average accelerates the change of w_k and b_k and reduces the possibility of being trapped in local minima and saddle points instead of the global minimum. This has been proved to improve network's performance on a satisfactory level [17]. By adjusting momentum factor β, the proportion of previous weights and biases can be altered.

The introduction of decaying learning rate $\eta(k)$ aims to converge to the minimum value more efficiently and accurately. In later stages of training, reducing the learning rate can reduce the changes of w_k and b_k, allowing faster convergence to the minimum instead of oscillating on an undesired scale. In this paper, different milestones are set for ultra-short-term WPF and short-term WPF. Learning rate $\eta(k)$ only decays at the milestones at a constant speed $c \in (0, 1)$ and remains the same otherwise, as in (16):

$$\eta(k) = \begin{cases} c\eta, & k \in milestones \\ \eta, & otherwise \end{cases} \tag{16}$$

The proposal of momentum algorithm and decaying learning rate greatly optimizes the performance of the ANN-based model, which will be proved in Section III & IV.

C. Time Series Reconstruction of Inputs

Input of the network is real WP and WS data in period $[t, t+T]$, and the output of the network is WPF values at a fixed time interval of 15 minutes in period $[t+T, t+2T]$. Different sets of inputs are evenly spaced in time series, constructing the input set:

$$I = \{I(\mathrm{t}_0), I(\mathrm{t}_0 + \Delta t), I(\mathrm{t}_0 + 2\Delta t), \dots, I(\mathrm{t}_{final})\} \tag{17}$$

Typically, Δt is chosen to be T. However, this convention is not beneficial for network training because each sample point is studied only once in one epoch, making it harder for the network to learn the characteristics of data.

With time series reconstruction (TSR), the input set gets denser as we choose $\Delta t < T$, so that samples in period $[\mathrm{t}_i + \Delta t, \mathrm{t}_i + T]$ are repeatedly trained, allowing multiple training procedures of the same sample. In view of the limitation of time series dataset, time series reconstruction enables larger and more detailed training set, allowing more efficient network training and better network performance.

D. Evaluation of the Prediction Accuracy

This paper utilizes RMSE and qualified rate (Q) as error indicators, as defined in equations (18) and (19), respectively. y_i refers to measured WP values, $\widehat{y_i}$ refers to forecasted values of y_i, and n is the number of samples. Both y_i and $\widehat{y_i}$ have already been normalized. RMSE indicates the average error between targets and forecasted outputs, Q indicates the ratio of qualified forecasted outputs, and the criterion of qualified forecasted outputs is defined in equation (20). RMSE and Q constitute different measures of the forecasting accuracy.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \widehat{y_i})^2} \tag{18}$$

$$Q = \frac{1}{n}\sum_{i=1}^{n} B_i \times 100\% \tag{19}$$

$$B_i = \begin{cases} 1, & 1 - |y_i - \widehat{y_i}| \geq 0.85 \\ 0, & 1 - |y_i - \widehat{y_i}| < 0.85 \end{cases} \tag{20}$$

III. Ultra-Short-Term Wind Power Forecasting

A. Optimal MLP Architecture and Model Parameters

The original dataset is obtained from a wind farm in Jiangsu, China. The preprocessed dataset is divided into a training set and a test set, comprising 83% and 17% of the original data, respectively. The training set is used for calibrating weights and biases of the neurons in MLP through the learning process [18]. The test set is used for evaluating the performance of the trained model. Based on the best performing model on the test set, the best MLP architecture is obtained.

For ultra-short-term WPF, the input is composed of WP and WS data samples in a period of 4 hours prior to the forecast. Since WP and WS measurements are taken every 30 seconds, the input layer contains 960 neurons. The output is composed of forecasted WP results at a time interval of 15 minutes in the future 4 hours, so the output layer contains 16 neurons. Based on time series reconstruction (TSR), four different time intervals between adjacent input vectors are chosen:

$$\Delta t_1 = 5,\ \Delta t_2 = 15,\ \Delta t_3 = 60,\ \Delta t_4 = 240 \tag{21}$$

with minute as the unit. Each time interval corresponds to an optimal MLP architecture with optimal model parameters, namely MLP_1, MLP_2, MLP_3, and MLP_4. Particularly, MLP_4 corresponds to a model without time series reconstruction because $\Delta t_4 = T$. Given the sizes of the input and output vectors, the numbers of hidden layers, numbers of neurons in each hidden layer, the momentum factor β, decaying speed c of learning rate, and milestones need to be subsequently determined.

Through trial and error, it is first found that if the number of training epochs is fixed, milestones can be set the same for three MLP architectures, and decaying speed c can be set as 0.5 with an initial learning rate of 0.02. Through the second trial and error analysis, two-hidden-layer layout is adopted. Different numbers of layout neurons and different momentum factors β are tested, and Table I presents RMSE and Q of examples of the three MLP architectures with different parameters. Balancing between performance and complexity of the model, MLP_1 adopts 110 and 40 neurons in the two hidden layers with $\beta = 0.9$, MLP_2 adopts 100 and 40 neurons in the two hidden layers with $\beta = 0.9$, MLP_3 adopts 100 and 40 neurons in the two hidden layers with $\beta = 0.8$, and MLP_4 adopts 110 and 40 neurons in the two hidden layers with $\beta = 0.85$.

B. Ultra-Short-Term Wind Power Forecasting Results

Based on the optimal models of MLP_1, MLP_2, MLP_3, and MLP_4, ultra-short-term WPF is carried out with four different intervals of time series construction and different optimizing algorithms.

Fig 4. shows examples of historic data, WPF and real WP. The blue curve is historic data over 24-h prior to the forecast, the red curve is real WP values measured, and the yellow curve is WPF. Fig. 4 (a)-(d) show data forms of MLP_1, MLP_2, MLP_3, and MLP_4, respectively.

As discussed in Section II-D, RMSE and Q are evaluated on the test set. Based on time series reconstruction, one time point is given by a total number of $240/\Delta t$ forecast values, so RMSE and Q of the forecasted result are evaluated with an averaged value, i.e., for time point i, forecasted WP value is defined as (22), and RMSE is defined as (23):

$$y_i^{average} = \frac{1}{k}\sum_{m=1}^{240/\Delta t} y_i^m \tag{22}$$

$$RMSE_i = |y_i^{average} - \hat{y}_i| \tag{23}$$

Subsequently, RMSE and Q of all points are defined as (24)-(26):

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(RMSE_i)^2} \tag{24}$$

$$Q = \frac{1}{n}\sum_{i=1}^{n} B_i \times 100\% \tag{25}$$

$$B_i = \begin{cases} 0, & RMSE_i \geq 0.15 \\ 1, & RMSE_i < 0.15 \end{cases} \tag{26}$$

As shown in Table I, MLP_2 gives the best forecasting accuracy while MLP_4 gives the poorest performance. Since MLP_4 corresponds to a model without TSR, it is proved that TSR can greatly improve forecast accuracy based on limited data samples, reducing RMSE by 0.066 and increasing qualified rate by about 11%. Furthermore, comparison between MLP_1, MLP_2 and MLP_3 proves that shorter time intervals of TSR do not necessarily lead to higher accuracy.

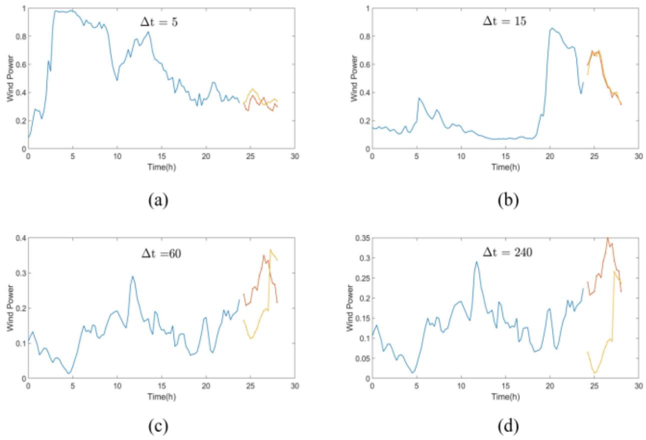

Fig. 4. Data forms of input, output, and target of four models. (a) MLP_1. (b) MLP_2. (c) MLP_3. (d) MLP_4. Blue line is historic wind power curve in previous 24 hours, red line is real wind power curve 4-hours-ahead, and yellow line is forecasted wind power curve 4-hours-ahead.

TABLE I. EXAMPLES OF RMSE AND QUALIFIED RATE OF MODELS

Model	(MLP Layout, β)	RMSE	Q(%)
MLP_1	**$(960 \times 110 \times 40 \times 16, 0.9)$**	**0.133**	**89.3**
	$(960 \times 110 \times 40 \times 16, 0.7)$	0.138	87.9
	$(960 \times 200 \times 80 \times 16, 0.9)$	0.193	79.8
MLP_2	**$(960 \times 100 \times 40 \times 16, 0.9)$**	**0.121**	**92.3**
	$(960 \times 100 \times 40 \times 16, 0.7)$	0.139	88.0
	$(960 \times 100 \times 50 \times 16, 0.9)$	0.186	81.2
MLP_3	**$(960 \times 100 \times 40 \times 16, 0.8)$**	**0.142**	**87.1**
	$(960 \times 110 \times 40 \times 16, 0.8)$	0.147	85.9
	$(960 \times 100 \times 50 \times 16, 0.85)$	0.208	74.9
MLP_4	**$(960 \times 110 \times 40 \times 16, 0.85)$**	**0.187**	**81.2**
	$(960 \times 110 \times 40 \times 16, 0.8)$	0.190	80.3
	$(960 \times 100 \times 40 \times 16, 0.9)$	0.203	76.1

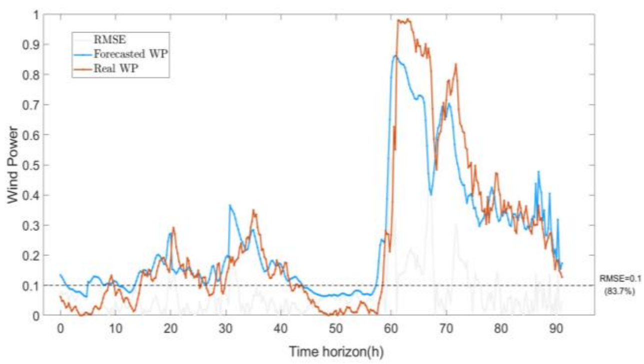

Fig. 5. Data forms of input, output, and target for four models. (a) MLP_1. (b) MLP_2. (c) MLP_3. (d) MLP_4. Blue line is historic wind power curve in previous 24 hours, red line is real wind power curve 4-hours-ahead, and yellow line is forecasted wind power curve 4-hours-ahead

TABLE II. RMSE AND QUALIFIED RATE COMPARISON OF DIFFERENT OPTIMIZATION ALGORITHMS

Method			RMSE	Q(%)
With TSR (MLP_2)	with momentum	with decaying learning rate	0.121	92.3
		w/o decaying learning rate	0.122	92.0
	w/o momentum	with decaying learning rate	0.130	90.0
		w/o decaying learning rate	0.130	90.1
W/O TSR (MLP_4)	with momentum	with decaying learning rate	0.187	81.2
		w/o decaying learning rate	0.190	80.2
	w/o momentum	with decaying learning rate	0.193	79.9
		w/o decaying learning rate	0.192	80.1

The model may easily converge to local minima and get over-fitted if prior data samples are studied for too many times. In this paper, choosing 15 minutes as the time interval of TSR gives the optimal model for ultra-short-term WPF. Fig 5. presents the entire forecast result by MLP_2. It should be clarified that RMSE in the figure refers to RMSE of each time point i defined in (23). The overall forecast accuracy is satisfactory. Few point forecasts show excessive deviation, illustrating good forecasting performance with RMSE ≤ 0.1 for 83.7% point forecasts. The forecast accuracy decreases with the increase of time interval as it exceeds 15 minutes, as shown in Table I.

The performance of different optimizing methods is presented in Table II and Fig. 6. Obviously, TSR leads to better model performance as discussed above. Momentum algorithm improves the model accuracy on a satisfactory extent, increasing qualified rate by 2.3% and reducing RMSE by about 0.01. Introducing decaying learning rate has no significant influence on model's accuracy, but enables shorter training time, which is desirable for practical applications. Fig. 5. shows that applying Z-scored method to preprocess the data leads to better model performance, reducing RMSE by about 0.01. It is noted that data utilized in this paper is sufficiently dense, so that the preprocessing step only requires eliminating the abnormal measurements. Therefore, Z-scored method is suitable here.

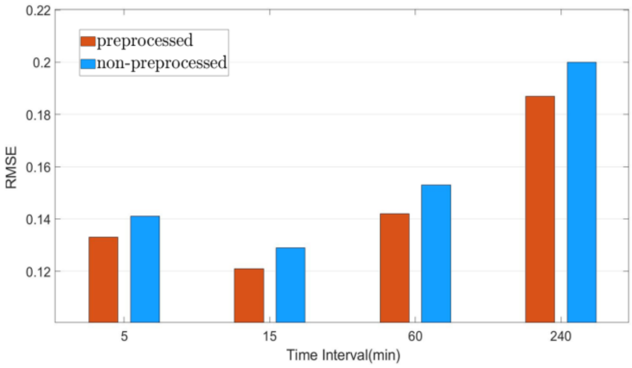

Fig. 6. Comparison between RMSE of two kinds of dataset. The blue bar represents the RMSE of wind power forecasting using non-preprocessed data as dataset, and the red bar represents RMSE of wind power forecasting using preprocessed data as dataset. The x-axis refers to different time intervals adopted by time series reconstruction.

IV. SHORT-TERM WIND POWER FORECASTING

A. Optimal Models and Result Analysis

Based on the method and results discussed in Section III, three different time intervals of TSR are chosen:

$$\Delta t_5 = 60,\ \Delta t_6 = 240,\ \Delta t_7 = 1440 \tag{27}$$

with minute being the unit. This corresponds to three optimal models MLP_5, MLP_6 and MLP_7. Particularly, MLP_7 is the model without TSR since $\Delta t_7 = T'$.

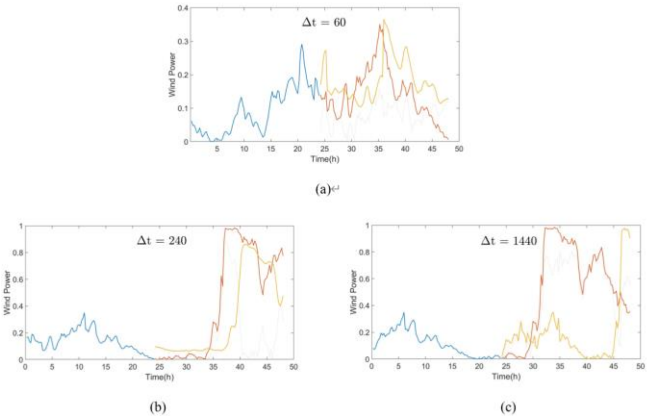

Fig. 7. Data forms of input, output, and target of three models. (a) MLP_5. (b) MLP_6. (c) MLP_7. Blue line is historic wind power curve in previous 24 hours, red line is real wind power curve 24-hours-ahead, yellow line is forecasted wind power curve 24-hours-ahead, and grey line is RMSE between real wind power curve and forecasted wind power curve 24-hours ahead.

TABLE III. COMPARISON BETWEEN MODELS OF SHORT-TERM WPF

Model	MLP Layout	Algorithm	RMSE	Q(%)
MLP_5 ($\Delta t_5 = 60$)	$5760 \times 700 \times 60 \times 96$	with momentum	0.154	85.2
		w/o momentum	0.159	83.8
MLP_6 ($\Delta t_6 = 240$)	$5760 \times 700 \times 55 \times 96$	with momentum	0.207	74.2
		w/o momentum	0.209	73.3
MLP_7 ($\Delta t_7 = 1440$)	$5760 \times 700 \times 55 \times 96$	with momentum	0.343	25.7
		w/o momentum	0.342	25.7

For short-term WPF, the input vector is composed of real WP and WS samples in the 24 hours prior, so that the input layer contains 5760 neurons. The output is composed of forecasted WP results at a time interval of 15 minutes in the future 24 hours, so the output layer contains 96 neurons. Fig 7. presents historic data, the forecast and real WP values with RMSE. The blue curve represents historic data input, the red curve is real WP, the yellow curve is forecasted WP, and the grey curve is RMSE.

Through trial and error, parameters and layouts of the three models are determined, as presented in Table III. Comparing MLP_5 with MLP_7, it can be proved that TSR can increase qualified rate by around 60% and reduce RMSE by around 0.2, thus dramatically improving the forecast accuracy by creating more training samples based on very limited data. Adopting momentum algorithm brings higher accuracy when the model is acceptable, while making no significant difference when the model performs poorly. Obviously, the method proposed in this paper is more suitable for ultra-short-term WPF than short-term WPF. With only WP and WS data accessible and no weather data, it is difficult to make satisfying 24-h-ahead forecasts.

V. CONCLUSIONS

The main contribution of this paper includes the introduction of time series reconstruction, optimized ANN-based model with MLP architecture, momentum algorithm and decaying learning rate to ultra-short-term and short-term wind power forecasting. Optimal MLP architecture and model parameters are determined through trial-and-error. Compared to traditional ANN method, the approach proposed can effectively increase the accuracy of both ultra-short-term (4-h) and short-term (24-h) WPF. For ultra-short-term WPF, TSR can increase qualified rate of forecasting by 11% and reduce RMSE by about 0.07, and momentum algorithm can reduce RMSE by about 0.01. For short-term WPF, TSR can dramatically increase qualified rate by about 60% and reduce RMSE by about 0.2. This paper also demonstrates that 24-h-ahead WPF is difficult to implement without weather data.

On the other hand, challenges still remain. Firstly, the optimal model is solely obtained from a trail-and-error procedure, thus lacking adaptability. Secondly, there is no solid physical reasoning for time series reconstruction method and momentum algorithm, so the generality of the model is not guaranteed.

REFERENCES

[1] Jyothi Varanasi, M.M. Tripathi, “Artificial Neural Network Based Wind Power Forecasting in Belgium,” 2016 IEEE 7th Power India International Conference (PIICON).

[2] Global Wind Energy Council (GWEC), Global Wind Report, 2022.

[3] L. Li, Y. Mei Wang, and Y. Qian Liu, “Impact of wake effect on wind power prediction,” in Proc. 2nd IET Renew. Power Gener. Conf.(RPG’13), Sep. 2013, pp. 1–4.

[4] Khawaja Masood Ahmed, Muhammad Arshad Khan, Imran Siddiqui, Siraj Khan, M. Shoaib, Ibrahim Zia, “Wind Speed Prediction from Site Meteorological Data Using Artificial Neural Network,” 2022 Global Conference on Wireless and Optical Technologies (GCWOT)

[5] M. Khalid and A. Savkin, “Closure to discussion on ‘a method for short-term wind power prediction with multiple observation points’,” IEEE Trans. Power Syst., vol. 28, no. 2, pp. 1898–1899, May 2013.

[6] Tao Hong, Pierre Pinson, Yi Wang, et.al., “Energy Forecasting: A Review and Outlook”, IEEE Open Access Journal of Power and Energy, vol. 7, pp. 376-388, Oct. 2020.

[7] J. Zeng and W. Qiao, “Short-term wind power prediction using a wavelet support vector machine,” IEEE Trans. Sustain. Energy, vol. 3, no. 2, pp. 255–264, Apr. 2012.

[8] H. Quan, D. Srinivasan, and A. Khosravi, “Short-term load and wind power forecasting using neural network-based prediction intervals,” IEEE Trans. Neural Netw. Learn. Syst., vol. 25, no. 2, pp. 303–315, Feb. 2014.

[9] Yicong Wang, “Short-term wind power forecasting by genetic algorithm of wavelet neural network,” 2014 International Conference on Information Science, Electronics and Electrical Engineering.

[10] E. Elattar, “Short term wind power prediction using evolutionary optimized local support vector regression,” in Proc. 2nd IEEE PES Int. Conf. Exhib. Innovative Smart Grid Technol. (ISGT Europe), Dec. 2011, pp. 1–7.

[11] Amandine Pierrot, Pierre Pinson, “Adaptive Generalized Logit-Normal Distributions for Wind Power Short-Term Forecasting”, 2021 IEEE Madrid PowerTech, Jul.2021.

[12] H. Bludszuweit, J. Dominguez-Navarro, and A. Llombart, “Statistical analysis of wind power forecast error,” IEEE Trans. Power Syst., vol. 23, no. 3, pp. 983–991, Aug. 2008.

[13] P. Louka,et al., “Improvements in wind speed forecasts for wind power prediction purposes using kalman filtering,” J. Wind Eng. Ind. Aerodyn., vol. 96, no. 12, pp. 2348–2362, 2008.

[14] P. Chen, T. Pedersen, B. Bak-Jensen, and Z. Chen, “Arima-based time series model of stochastic wind power generation,” IEEE Trans. Power Syst., vol. 25, no. 2, pp. 667–676, May 2010.

[15] Qianyao Xu, Dawei He, Ning Zhang, Chongqing Kang, et al., “A Short-Term Wind Power Forecasting Approach With Adjustment of Numerical Weather Prediction Input by Data Mining,” IEEE Trans. Sustain. Energy, vol. 6, no. 4, Oct. 2015.4.

[16] R. Dar, P. Winzer, “On the Limits of Digital Back-Propagation in Fully Loaded WDM Systems,” IEEE Photonics Technology Letters, vol. 28, issue. 11, Jun. 2016.1.

[17] Navjot Singh, Deepesh Data, Jemin George, et al., “SQuARM-SGD: Communication-Efficient Momentum SGD for Decentralized Optimization,” IEEE Journal on Selected Areas in Information Theory, vol. 2, issue. 3, Sep. 2021.

[18] Christos S. Ioakimidis, Konstantinos N. Genikomsakis, Panagiotis I. Dallas, et al., “Short-term wind speed forecasting model based on ANN with statistical feature parameters,” IECON 2015 - 41st Annual Conference of the IEEE Industrial Electronics Society.