IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received February 23, 2021, accepted March 4, 2021, date of publication March 10, 2021, date of current version March 17, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3065502

Wind Power Forecasting Using Attention-Based Recurrent Neural Networks: A Comparative Study

BIN HUANG[1], YUYING LIANG[1], AND XIAOLIN QIU[2]
[1]Department of Electrical and Electronic Engineering, Nanchang Institute of Technology, Nanchang 330003, China
[2]Department of Energy and Environmental Engineering, Nanchang Institute of Technology, Nanchang 330003, China

Corresponding author: Yuying Liang (yuying_liang68@163.com)

ABSTRACT Wind power is one of the most efficient renewable resources without emissions. Nonetheless, it is difficult to exactly forecast wind power generation given historical power and wind speed information, the failure of which may cost the risk of large-scale outages. This article takes a close look at the artificial recurrent neural network framework in the application of wind power forecasting. More intelligent mechanisms using attention to capture spatial-temporal patterns within historical data are emphasized in this work and are shown to be state-of-the-art for short-term wind power forecasting. Our experiments at a wind farm in southeast Australia using only the historical wind power generation and wind speed records from ambient weather stations show that, e.g., 7.4750% in mean absolute error (MAE) and 0.3345 in the coefficient of variation in the root mean squared error (CV-RMSE) for half-hour-ahead prediction. To interpret how the three models under consideration—the long- and short-term time-series network (LSTNet), the temporal pattern attention-based long short-term memory (TPA-LSTM) and the dual-stage attention-based recurrent neural network (DA-RNN)—work, we visualize and analyze the details of the models so that further improvement can be made by combining the advantageous components of the models.

INDEX TERMS Wind power forecast, time-series forecast, recurrent neural network, attention, deep learning, DA-RNN, LSTNet, TPA-LSTM.

I. INTRODUCTION

As one of the most promising renewable resources of power without emissions, wind power has gained enormous attention from investors and governments around the world. Despite the unpredictable impacts of COVID-19 on the global energy market, the Global Wind Energy Council (GWEC) expects that over 355 GW of new wind power capacity will be added worldwide before 2025, which is more than half of the 651 GW installed capacity by the end of 2019 [1]. The growing permeability of highly volatile renewable resources poses challenges to the flexibility of power grids, with the large-scale outage on 9 August 2019 in the UK striking a wake-up call to us [2]. Techniques such as intelligent control provide potential solutions to the issue, where strategies of accurately forecasting wind power generation are among the key difficulties.

Nevertheless, our knowledge of efficient and reliable wind power forecasting techniques is still limited compared to the rapid expansion of the global wind power market. Unlike other renewable resources with strong seasonal patterns such as solar energy, trends and fluctuations in generated wind power are notoriously elusive to grasp. Traditionally, physical approaches using power curves provided by the turbine manufacturer to convert wind speed to power suffer severely from ambient noise and outliers [3]–[5], and accurately forecasting the exact wind speed at the desired location and turbine hub height is usually an intractable task.

Data-driven methods, especially methods derived from advanced artificial intelligence techniques, have become much more powerful alternatives to address this problem. In particular, recurrent neural networks (RNNs) [6], having prevailed in the field of natural language processing (NLP) due to their deep recurrent design to learn highly nonlinear temporal dynamics from sequences, can be readily implanted to time-series forecasting tasks [7]–[10]. Abundant works to improve the overall performance of RNN conditioned on various application settings contribute to the boom in the field, including the two popular RNN variants, namely, the long short-term memory (LSTM) [11] and gated recurrent

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

unit (GRU) [12], among others. These variants can capture temporal patterns of relatively longer-term as well as short-term dependencies in the input signals.

Drawbacks of applying LSTM and GRU directly to wind power forecasting, however, lie in their limited capacity to learn complicated temporal and spatial patterns from multivariate time series as a whole. Indeed, along the *temporal* axis, very long-term dependencies may be easily lost due to vanishing gradient [13]; along the *spatial* axis, historical wind speed measurements from surrounding meteorological stations interplay with the power generated by the targeted wind farm chaotically and should be modeled effectively. Both temporary and spatial patterns can be captured more intelligently with the aid of a so-called *attention* mechanism [14], [15] to be elaborated in this article. We shall review the wind power forecasting literature in greater length in the next subsection.

A. RELATED WORKS

The development of wind power forecasting has witnessed an increasingly important role played by big data theories and techniques within recent decades. Methods inspired by statistical learning and data mining dominated the field in the first ten years of the 21st century, successfully breaking through the paradigm of the autoregressive integrated moving average model (ARIMA), which is powerful in exploiting autocovariance information but depends highly on the stationary, linear and homoscedastic assumptions of the time series. Machine learning algorithms such as support vector regression (SVR) [16]–[18], multilayer perceptron (MLP) [16], [19], artificial neural network (ANN) [20], [21], random forest (RF) [17], [22], [23] and extreme learning machine (ELM) [24]–[27] are demonstrated to be among the most efficient models selected. However, these algorithms were originally created for general regression problems, and the temporal/ordinal nature inherent in the time-series data is therefore neglected.

Apart from the options of modern machine learning algorithms equipped with the capability of learning nonlinear, complex interdependencies within the data, adaptive strategies are shown to be equally crucial for elevating the prediction accuracy even more. One of the major purposes of these adaptions, in summary, is to encode as much of the spatial-temporal information from the historical wind power and exogenous (typically wind speed and wind direction) data as possible while at the same time excluding any random noise or disturbance (see, e.g., [24], [25]).

One classical routine for achieving this end is to decompose the historical time series into additive subseries of different frequencies. Various forecasting algorithms can then be applied to the denoised subseries separately, and the final prediction is obtained by summing these results. The rationale behind the decomposition-based approach rests on the simpler seasonal patterns of the subseries that can be modeled by regression algorithms with less difficulty. Wavelet decomposition (WD) [28], [29], empirical mode decomposition (EMD) [30], [31] and variational mode decomposition (VMD) [25], [32] are popular decomposition approaches. For a thorough discussion of decomposition-based hybrid models for wind power forecasting, we refer interested readers to [33].

Additionally, feature selection, or feature engineering in machine learning jargon, is a stage of data preprocessing commonly used along with the decomposition stage in hybrid models for wind power forecasting. It either filters out unrelated features or maps the original input into a subspace before training. The most straightforward method of selecting a proper time window from historical data is via a (partial) autocorrelation function (PACF, ACF). Other methods derived from unsupervised learning, such as principal component analysis (PCA) and Gram-Schmidt orthogonalization (GSO) [24], [34], can be used to select both temporal and spatial features but still encode only linear relationships within the inputs. The mean impact value (MIV) [35] method can dynamically optimize the selected feature set according to some objective function and hence is a widely used nonlinear criterion. However, the hard thresholds of the feature selection procedure may be less flexible than soft thresholds used in attention-based methods.

The application of deep learning methods and techniques initiates a new paradigm for wind power forecasting. Recurrent neural networks are designed specifically for learning temporal representations of a series, which differs greatly from the classical regression algorithms borrowed from the statistical learning and data mining community. In recent years, researchers have started employing an echo state network (ESN), an instance of RNN with a sparsely connected hidden layer and randomly assigned weights, to improve the accuracy of wind power forecasting [36], [37]. Long short-term memory (LSTM) is also favored by an increasing number of researchers [38]–[40], and quite a few works are devoted to combining it with existing wind power forecasting strategies such as VMD [41], [42], Gaussian mixture model [43] and ESN [37].

Deep neural networks also offer brand new methods for encoding spatial-temporal information from historical wind data. In fact, when adopting an encoder-decoder model of RNN, a better representation of raw input is inferred in the encoding stage, whereas the decoding stage mainly focuses on the regression job. Researchers from machine translation are among the first to notice that encoding a sentence to a fixed-length vector representation often causes information loss when dealing with long texts. The attention mechanism, proposed in [15] and [14], solves this problem by assigning different weights to the intermediate hidden states according to their relevance to the specific target states. Combining the weighted hidden states yields the context vectors of the target states to be fed into the decoder. The attention mechanism is embedded in the deep neural architecture so that it can be learned along with other parameters in the entire network.

Although it originated in machine learning, the attention mechanism has been adopted by time-series analysts.

TABLE 1. Summary of wind power forecasting paradigm transition with methods for capturing spatial-temporal patterns.

paradigm	model	to capture spatial-temporal patterns	method
statistics	ARIMA	linear	
machine learning	SVR, MLP, ANN, RF, ELM	decomposition - denoising	WD, EMD, VMD
		feature selection	PACF, ACF, PCA, GSO, MIV
deep learning	ESN, LSTM, GRU	decomposition - denoising	WD, EMD, VMD
		attention	attention-RNN, **LSTNet, TPA-LSTM, DA-RNN**

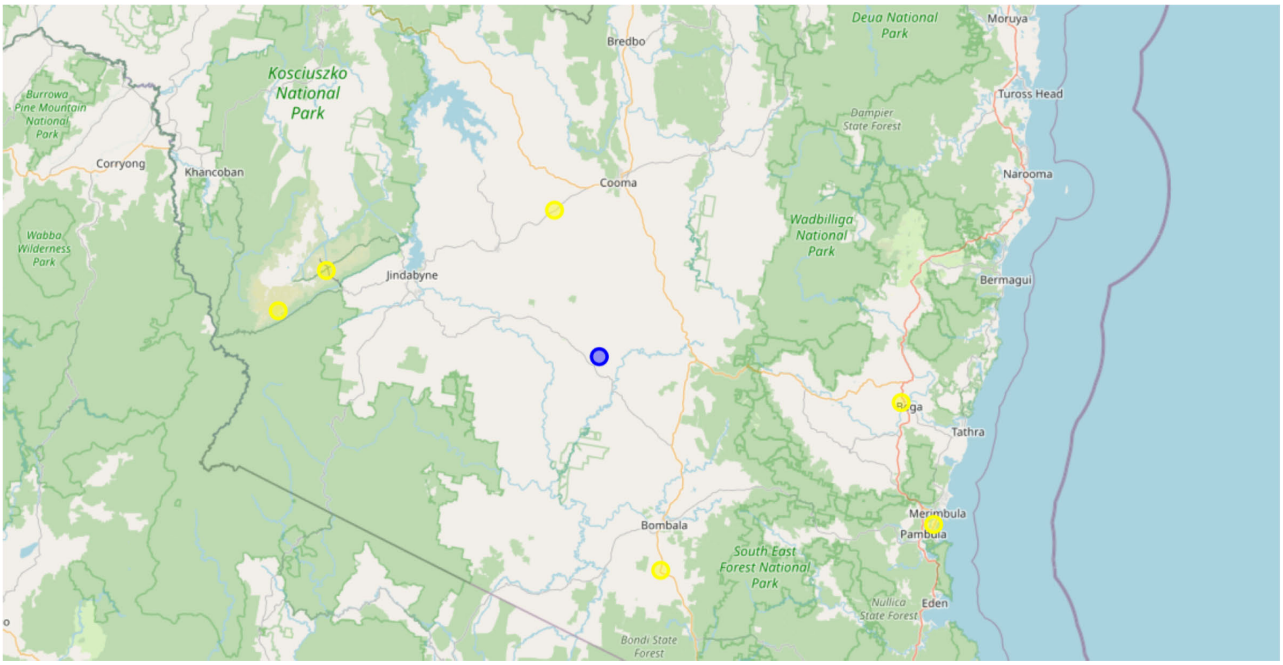


FIGURE 1. Map of Boco Rock Wind Farm (blue circle) in New South Wales, Australia and 6 nearest weather stations (yellow circles).

The most successful examples that incorporate attention into the framework of recurrent neural networks have been proposed in recent years, including LSTNet [13], TPA-LSTM [44] and DA-RNN [45]. We delay the theoretical elaboration of these models to section II. Table 1 summarizes the paradigm transition in the wind power forecasting literature, along with working methods to capture temporal and spatial patterns, as we discussed above.

B. OUR CONTRIBUTIONS

We dedicate this article to the investigation of three state-of-the-art recurrent neural network variants using an attention mechanism to forecast wind power generation: LSTNet [13], TPA-LSTM [44] and DA-RNN [45]. To the best of our knowledge, this work is the first to fill the gap between the wind power forecasting community and the rapidly developing attention-based RNN among time-series analysts. We test the models on a wind farm in New South Wales, Australia. The wind power data along with the meteorological observations in the surrounding weather stations are collected for training and inference. Fig. 1 shows the geographical distribution of the wind farm and weather stations. Based on our comprehensive experiments, an in-depth analysis of how attention helps to capture temporal and spatial patterns from the raw inputs is performed in this article.

The article is organized as follows. The architecture and mathematical foundations of LSTNet, TPA-LSTM and DA-RNN are introduced in section II. We present thorough experimental results in section III and delay the visualization and interpretation of the attention mechanisms of the models to section IV. Section V concludes the entire study with a discussion on the pros and cons of the three models so that future improvement could be made upon our work.

II. ATTENTION-BASED RECURRENT NEURAL NETWORKS

A. BUILDING BLOCKS

Before addressing the abstraction of the wind power forecasting problem, we remind our audience of the most basic components in the deep learning library here. By assembling these following components as needed, we can build the three state-of-the-art attention-based RNNs.

1) RNN AND RECURRENT SKIP

Consider a sequence $\{x_1, x_2, \ldots, x_T\}$, where x_t can be a scalar or vector according to the setting of the specific problem. A recurrent neural network calculates hidden states h_t based

on a recurrent function as follows:

$$h_t = RNN(h_{t-1}, x_t). \tag{1}$$

The hidden states encode the information in the passing input entries that is most relevant to producing the targeted outputs. The parameters of the recurrent function are shared along the sequence indexed by t and can be trained using backpropagation. However, since conventional RNN cells are incapable of learning very long-term interdependency due to gradient vanishing, one can use the recurrent skip to leverage the periodic pattern within the input data:

$$h_t = RNN(h_{t-p}, x_t), \tag{2}$$

where the period value p is determined by the input data as prior knowledge. Here, we adopted the formulation of recurrent skip in [13]. Note that *RNN* represents a mapping that can be embodied by either LSTM or GRU, which are to be defined explicitly next.

2) LSTM

Proposed in [11], the long short-term memory (LSTM) cell is a popular instance of RNN. Its recurrent function is realized as follows:

$$f_t = \sigma(W_f[h_{t-1}; x_t] + b_f) \tag{3a}$$
$$i_t = \sigma(W_i[h_{t-1}; x_t] + b_i) \tag{3b}$$
$$o_t = \sigma(W_o[h_{t-1}; x_t] + b_o) \tag{3c}$$
$$s_t = f_t \odot s_{t-1} + i_t \odot tanh(W_s[h_{t-1}; x_t] + b_s) \tag{3d}$$
$$h_t = o_t \odot tanh(s_t), \tag{3e}$$

where $\odot$ denotes the elementwise multiplication and σ is the logistic function defined elementwise by

$$\sigma(x) = \frac{1}{1 + e^{-x}}, \tag{4}$$

and the hyperbolic tangent function *tanh* is also applied elementwise to its inputs:

$$tanh(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}}. \tag{5}$$

Notation $[h_{t-1}; x_t]$ is the concatenation of the previous hidden state and the current input vector. W_f, W_i, W_o are matrices, and b_f, b_i, b_o are vectors of corresponding shapes that should be clear in context.

The key innovation of LSTM involves the cell state s_t. Compared to the rapid change in the hidden state h_t, the cell state can memorize a relatively longer history. Its memory is controlled by the forget gate f_t, whereas i_t is the input gate that selectively memorizes the current information. The hidden state h_t is obtained from s_t through an output gate.

3) GRU

The gated recurrent unit (GRU) was created 17 years later than LSTM in [12] but is quickly accepted by the academic and industrial communities due to its computational efficiency. Its recurrent function is updated as follows:

$$r_t = \sigma(W_r[h_{t-1}; x_t] + b_r) \tag{6a}$$
$$u_t = \sigma(W_u[h_{t-1}; x_t] + b_u) \tag{6b}$$
$$s_t = RELU(W_s[r_t \odot h_{t-1}; x_t] + b_s) \tag{6c}$$
$$h_t = (1 - u_t) \odot h_{t-1} + u_t \odot s_t. \tag{6d}$$

Most of the notations are exactly the same as in the case of LSTM, except that here we use the rectified linear unit (ReLU) to activate the current information for empirical reasons [13]. The key improvement of GRU is that it reduces the 3 gates in LSTM to only 2 gates. In fact, the forgetting and selective memory are controlled simultaneously by one update unit u_t. The unit r_t is used to reset the previous hidden state.

4) CNN

A convolutional neural network (CNN) [46], [47] is a special kind of artificial network inspired by linear filters in the signal processing domain. A kernel C whose size is small compared to the input matrix X slides across the whole X to produce a hidden matrix:

$$H = RELU(C \star X + B), \tag{7}$$

where $\star$ denotes the convolution operation (with flipped kernel):

$$[C \star X]_{i,j} = \sum_m \sum_n C_{m,n} \times X_{i+m,j+n}. \tag{8}$$

Instead of using multiple different weight matrices to produce one hidden matrix, it is advantageous to share parameters among them in a convolutional layer. The basic assumption supporting the usage of CNN is that a similar pattern should be shared throughout the input X regardless of the position of its occurrence. That is why the most well-known application of CNN rests in the field of computer vision. For time-series input, nonetheless, CNN is still helpful for capturing temporal or local motifs. Moreover, multiple kernels are usually used in one layer to learn different motifs, yielding a hidden tensor with multiple channels.

5) ATTENTION MECHANISM

Emerging as recently as 2014, the attention mechanism inevitably has multiple different types and forms, most of which were created to address very specific engineering problems. In this article, it is sufficient to consider Bahdanau attention [15] as our basic framework, whereas using the score functions proposed in Luong attention [14] for complement.

The context vector

$$c_i = \sum_j \alpha_{ij} h_j \tag{9}$$

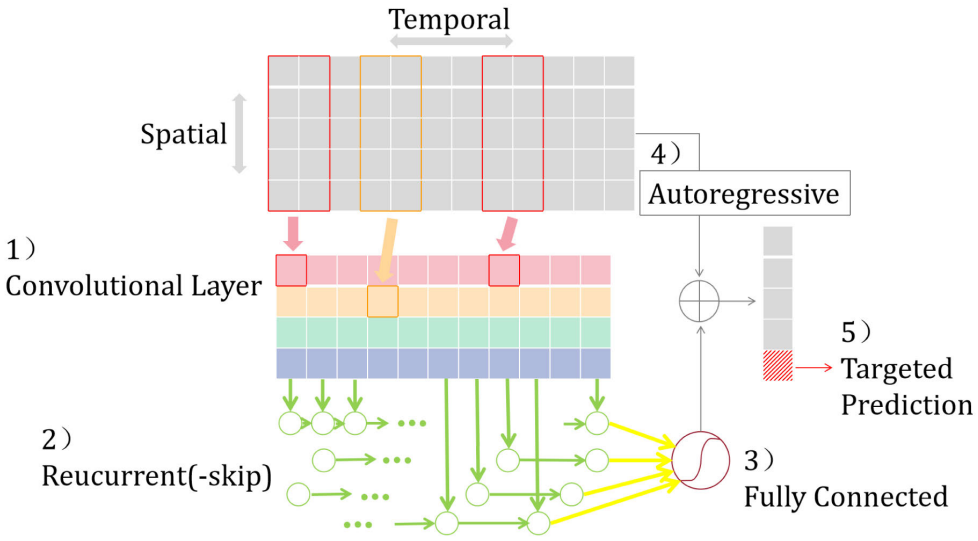


FIGURE 2. LSTNet.

is a weighted sum of existing hidden states, where the weights obtained by the softmax function

$$\alpha_{ij} = \frac{exp(e_{ij})}{\sum_k exp(e_{ik})} \tag{10}$$

or sigmoid logistic

$$\alpha_{ij} = \frac{exp(e_{ij})}{1 + exp(e_{ij})} \tag{11}$$

are called attention weights, determined by

$$e_{ij} = score(s_i, h_j). \tag{12}$$

Here, s_i is some sequence from the decoder side as in the original machine translation setting under the encoder-decoder architecture [15]. The attention mechanism is therein used to match the decoder sequences s_i with the most relevant encoder sequences h_j, where the relevance is measured by a score function. Borrowing terminology from the information retrieval domain, s_i can be seen as the *queries*, whereas in the above framework, h_j are considered both *keys* and *values* [48]. The context vector c_i compresses the information from all the previous h_j sequences that are most needed by the decoder. In practice, s_i and h_j can be replaced with other queries, keys and values according to the application.

Luong *et al.* [14] proposed several different score functions:

$$score(s_i, h_j) = \begin{cases} s_i^\top h_j \\ s_i^\top W h_j \\ v^\top tanh(W[s_i; h_j] + b) \end{cases} \tag{13}$$

namely, the dot, general and concat functions.

B. PROBLEM FORMULATION

We are now ready to address the specific time-series context. Our wind power forecast problem can be formulated as a nonlinear autoregressive exogenous model (NARX) as follows. An input sample is a T-length time window given by n exogenous/driving series $\mathbf{X}$ and a target series $\mathbf{y}$:

$$\mathbf{X} = (\mathbf{x}^1, \mathbf{x}^2, \ldots, \mathbf{x}^n)^\top = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_T) \in \mathcal{R}^{n \times T}, \tag{14a}$$

$$\mathbf{y} = (y_1, y_2, \ldots, y_T) \in \mathcal{R}^T, \tag{14b}$$

where $\mathbf{x}^k \in \mathcal{R}^T$ is a driving series, and $\mathbf{x}_t \in \mathcal{R}^n$ denotes the n driving series at time step t. The problem is to predict the value of the target series in the h time horizon:

$$\hat{y}_{T+h} = F(\mathbf{X}, \mathbf{y}). \tag{15}$$

Our job is to learn the function $F(\cdot)$ from historical wind data and then test this a learned function for inference accuracy on the test dataset.

C. LSTNET

Proposed in [13], the long- and short-term time-series network (LSTNet) is dedicated to forecasting multivariate time series, and therefore, its framework is slightly different from our NARX problem. Instead of predicting $\hat{y}_{T+h}$ immediately, it regards the concatenated matrix $\tilde{\mathbf{X}} = [\mathbf{X}; \mathbf{y}]$ as a whole and is designed to predict $\tilde{\mathbf{x}}_{T+h} = (x^1_{T+h}, x^2_{T+h}, \ldots, x^n_{T+h}, y_{T+h})^\top$. In other words, driving features such as wind speed from ambient weather stations as well as the wind power will be predicted altogether in the LSTNet model, and the targeted value can be immediately seen from the output vector.

We illustrate the working flow of LSTNet in Fig. 2.

The inference procedure of the model is summarized as follows:

1) Feed the input multivariate series $\tilde{\mathbf{X}}$ into the first layer of LSTNet: the convolutional layer aimed at extracting

TABLE 2. Comparison of the three state-of-the-art models, based on the way spatial-temporal information is extracted and how the final prediction is made.

	temporal information	spatial information	final prediction
LSTNet	recurrent skip, CNN	CNN	dense layer, ARIMA
TPA-LSTM	CNN	LSTM, attention	dense layer
DA-RNN	attention-based LSTM	attention-based LSTM	dense layer

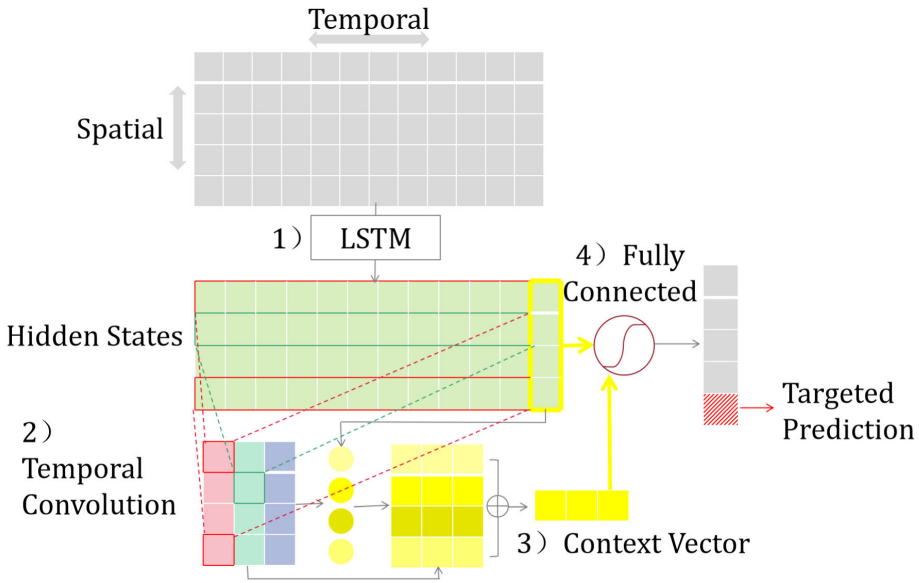


FIGURE 3. TPA-LSTM.

interdependencies among different spatial variables as well as ultrashort-term temporal patterns.

2) The output of the convolutional layer is put into the recurrent component and the recurrent-skip component simultaneously. The recurrent layer uses GRU as proposed in [13]. The value p in the recurrent components can be determined beforehand or tuned in validation.
3) A dense layer is added to combine $p+1$ values, namely, the last hidden state of the recurrent component and all the hidden states of the recurrent-skip component from the last period.
4) In parallel to the neural network, an autoregressive (AR) component is applied directly to the input series to predict the linear part.
5) The final prediction is the combination of both the neural network and AR results.

The recurrent-skip mechanism is shown to outperform conventional machine learning models in [13], as long as the periodic behavior of the time series is evident enough. In other words, LSTNet relies on step 2) to capture temporal seasons. However, the convolutional component of step 1) extracts different spatial patterns among the input variables, as well as some ultrashort-term temporal patterns, depending on the width of the kernels. Table 2 summarizes and compares how the spatial-temporal information is derived in the three models.

D. TPA-LSTM

The temporal pattern attention-based long short-term memory (TPA-LSTM) model proposed in [44] is another simple yet powerful framework for multivariate time-series forecasting equipped with the capability of recognizing spatial-temporal patterns within the data. Similar to LSTNet, it was originally dedicated to predicting $\tilde{\mathbf{x}}_{T+h}$ as a whole, yet we can easily obtain the targeted prediction from its output vector. The procedure is illustrated in Fig. 3 and summarized as follows:

1) The first layer is an LSTM network, producing a series of hidden states.
2) Given the hidden states from the first layer, we detect the temporal patterns using a convolutional layer. Specifically, kernels of shape $1 \times (T-1)$ are applied to the hidden matrix excluding the hidden state at the last time step.
3) Next, an attention mechanism is used to select relevant spatial variables from the output matrix of the convolutional layer. The rows of the output matrix correspond to the keys/values, and the query vector is the hidden state at the last time step. The attention mechanism produces a context vector that encodes both temporal and spatial information related to the last time step.
4) Finally, a dense layer is applied to the concatenation of the last hidden state and the context vector.

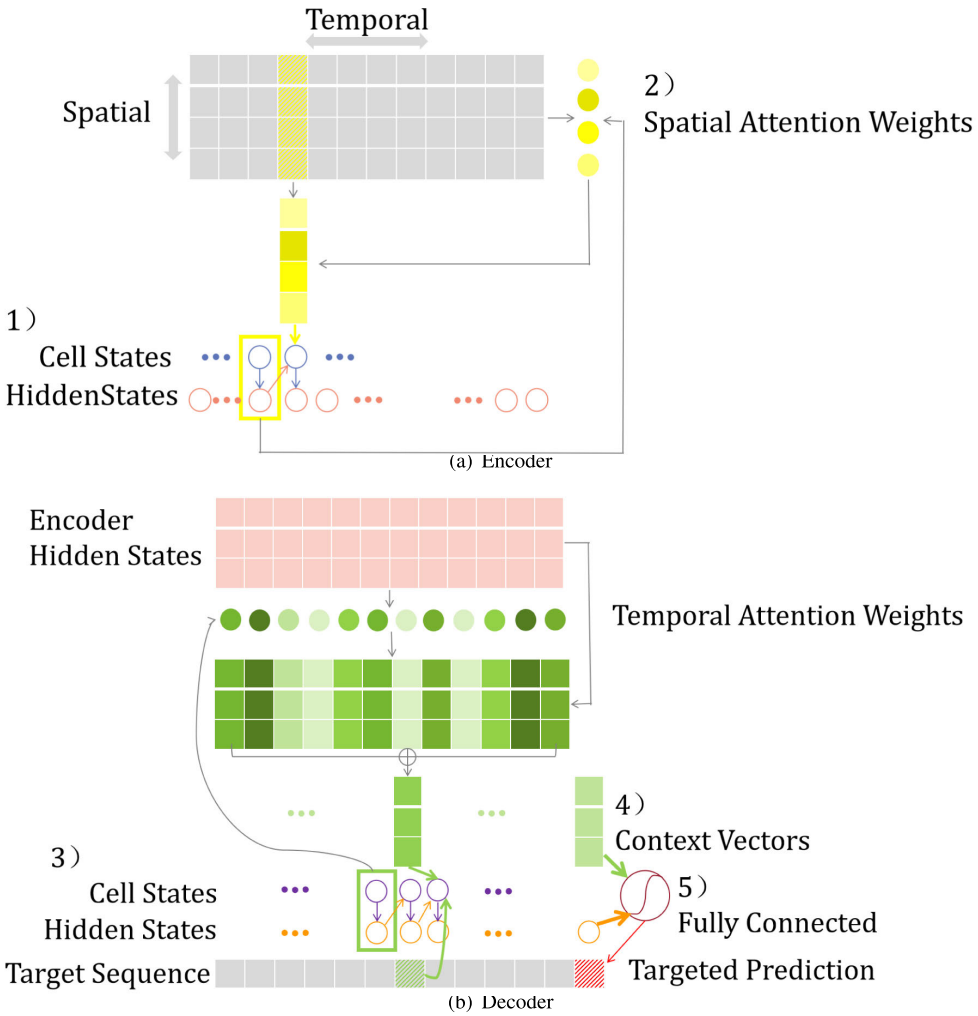


FIGURE 4. DA-RNN.

In contrast to LSTNet, where the seasonal behavior on the temporal axis is emphasized, TPA-LSTM was created to outperform classical attention by focusing on the spatial variable interdependencies. The idea is to distinguish more significant variables for forecasting from less significant variables in step 3). The temporal features are encapsulated by convolutional kernels in step 2). Please refer to Table 2 for a summary of the way spatial-temporal patterns are encoded in TPA-LSTM.

E. DA-RNN

The dual-stage attention-based recurrent neural network (DA-RNN) is slightly more complicated than the previous two models. Inspired by the two-stage mechanism of human attention, the model encompasses an encoder to adaptively select the elementary stimulus features in the driving series and then a decoder to select relevant encoder hidden states across all time steps. Despite its complexity, DA-RNN was invented to solve NARX problems directly and hence should be more pertinent to our wind power forecasting task than LSTNet and TPA-LSTM.

The working procedure is illustrated in Fig. 4. To be elaborate

1) Given the exogenous input series $\mathbf{X}$, an LSTM network is used to obtain encoder hidden states. At any time step, the input vector x_t is preprocessed by an attention mechanism described in step 2) before being fed into the LSTM. The hidden states of LSTM are thus obtained iteratively.
2) Using the encoder cell state joined with the hidden state at time step $t-1$ as the query vector, the attention weights correspond to the query's interdependency on the multiple driving sequences. Instead of calculating a context vector as in classical attention, we only rescale the current input x_t by the attention weights.
3) The decoding process takes the encoder hidden states as input and applies another LSTM network. This time, at any time step t, a context vector $\mathbf{c}_{t-1}$ derived from step 4) along with the value of target series y_{t-1} are fed into the decoder LSTM to compute the hidden state h'_t.
4) To compute a context vector $\mathbf{c}_t$, use the decoder cell state and hidden state at $t-1$ as query vector, and the

TABLE 3. Time-series division.

	starting time	ending time	percentage	time steps	days
training series	2018-01-01 00:00	2018-10-19 23:30	80	14016	292
validation series	2018-10-20 00:00	2018-11-25 11:30	10	1752	36.5
testing series	2018-11-25 12:00	2018-12-31 23:30	10	1752	36.5

encoder hidden states throughout all the time steps are the keys/values to address.

5) Finally, a fully connected layer converts the last decoder hidden state h'_{T+1} along with the last context vector from the temporal attention $\mathbf{c}_{T+1}$ to the value of the targeted prediction.

The original DA-RNN was created for the prediction in one time step only. However, it is not hard to expand the forecasting horizon to multiple time steps, just as how this is implemented in LSTNet and TPA-LSTM using the hidden states and contexts from the last time step or input period. As demonstrated in Table 2, DA-RNN extracts both temporal and spatial information adaptively using attention-based LSTM instead of the less dynamic convolutional components that appeared in the other two models.

III. EXPERIMENTAL DETAILS

A. DATA PREPROCESSING

Our collected data are composed of two parts: the amount of power generated by all the dispatchable units interconnected to the entire Australian Electricity Grid in 2018, exported from the Aneroid Energy database via [49] that were originally provided by the Australian Energy Market Operator (AEMO), and the detailed meteorological data received upon request from the Australian Bureau of Meteorology [50] during the same time period. For the sake of our comparative study for wind power forecasting, we only focus on the Boco Rock Wind Farm in New South Wales, Australia, which has 113 MW of total registered capacity of electricity. Since weather observations at the exact location of the turbines are unavailable, we choose to use the data from the six nearest weather stations surrounding the targeted wind farm, as shown in Fig. 1.

The raw data are preprocessed to fit the NARX framework of our wind power forecasting problem. The time steps are set to be half hour separated, yielding a 17,520-length series for the entire year of 2018. The target series $\mathbf{y}$ is the wind power in MW, and the $n = 6$ exogenous series $\mathbf{X}$ are the wind speed measured in the ambient weather stations in m/s. Instead of determining a specific time window T now, we choose to experiment on different values and then pick up the best one by validation. We also test over various time horizons h for the accuracy of the models in various short-term forecasting tasks.

Following the standard statistical learning practice, we divide the entire time series into training, validation and testing subseries in chronological order, according to Table 3. We normalize the seven series separately to the range of [0, 1] in the training data and then apply the trained scaler to validation and testing subseries.

TABLE 4. Prepared datasets.

	number of samples
training set	$14016 - T - h + 1$
validation set	1752
testing set	1752

To train our deep neural networks, as in any data-driven regression models, we prepare pairs of normalized input $(\mathbf{X}, \mathbf{y})$ and output y_{T+h}, obtained by moving one time step at a time across the entire series. Therefore, the actual number of training samples depends in part on the values of T and h. The samples for validation and testing immediately follow their training counterparts in time order, and hence, the sizes of the prepared datasets are settled in Table 4.

B. TRAINING DETAILS

1) PERFORMANCE EVALUATION

We use three assessment criteria to evaluate the performance of our models of interest, namely, the root mean squared error (RMSE), the mean absolute error (MAE) and the coefficient of variation in the root mean squared error (CV-RMSE). For ease of notation, consider the N samples in validation or testing sets, whose real outputs are denoted as y_i and are predicted by the trained models to be $\hat{y}_i$. The evaluation metrics measure the deviation of $\hat{y}_i$ from y_i, where $i = 1, 2 \ldots, N$. The three criteria are defined as follows:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N}(\hat{y}_i - y_i)^2}{N}} \tag{16a}$$

$$MAE = \frac{\sum_{i=1}^{N}|\hat{y}_i - y_i|}{N} \tag{16b}$$

$$CV - RMSE = \frac{RMSE}{mean(y_i)}. \tag{16c}$$

2) TRAINING METHODS

To train the parameters in an attention-based RNN, we use iterative algorithms based on gradients to optimize a loss function that measures how well the model predicts the training samples. Backpropagation (BP) is widely used in deep neural networks to compute gradients and is realized by transmitting partial gradients from the output layer back to the input layer following the computational graph of the network. Various adaptive alternatives to accelerate the convergence of gradient-based iterative algorithms are proposed in the literature, and here, we choose to use the Adam algorithm [51].

TABLE 5. Ranges of hyperparameters for selection. The best combinations validated via a grid search are in boldface.

LSTNet	
time window	**48** 96 240
kernel width	1 **3** 5
number of kernels	5 15 **40**
recurrent skip	16 **24** 48 72

TPA-LSTM	
time window	**48** 96 240
LSTM hidden units	16 32 **64** 128
number of kernels	15 40 **100**

DA-RNN	
time window	**48** 96 240
encoder hidden units	16 32 **64** 128
decoder hidden units	16 32 **64** 128

TABLE 6. Comparison of different forecasting models measured by three metrics. The best (smallest) value of each criterion is in boldface.

Horizon		1			3			6		
Metric		RMSE	MAE	CV-RMSE	RMSE	MAE	CV-RMSE	RMSE	MAE	CV-RMSE
Models	SVR	12.5796	9.1130	0.3612	20.4066	14.6791	0.5860	26.7063	19.7560	0.7669
	ELM	17.3901	12.9762	0.4993	22.8990	17.1093	0.6575	27.1661	21.0573	0.7801
	RBF	14.6849	10.4098	0.4217	22.6986	17.0870	0.6518	28.2073	22.0740	0.8100
	EMD-WT-SVR	16.8080	11.6701	0.4826	24.0986	17.6906	0.6920	31.5922	24.0957	0.9071
	vanilla LSTM	11.7987	7.6449	0.3388	19.2960	13.8563	0.5541	**24.7717**	**18.9129**	**0.7113**
	attention-LSTM	12.0381	7.8949	0.3457	20.2105	14.3660	0.5803	26.2272	19.9908	0.7531
	LSTNet	**11.6477**	7.6471	**0.3345**	**19.1159**	13.8617	**0.5489**	24.9714	19.5062	0.7170
	TPA-LSTM	11.6551	7.5637	0.3347	19.4420	13.8280	0.5583	24.8313	19.0816	0.7130
	DA-RNN	11.8496	**7.4750**	0.3403	19.5291	**13.2781**	0.5608	30.6838	26.7237	0.8811

We also use techniques such as learning rate reduction, early stopping and dropout to facilitate the training.

The loss function used in the experiment is the squared loss (L2-loss). Therefore, the objective of the training process is the following optimization problem:

$$\min_{\theta} \sum_{i=1}^{N_{train}} (\hat{y}_i - y_i)^2 \tag{17}$$

where θ denotes the trainable parameters of the model $F(\mathbf{X}, \mathbf{y})$, and the squared loss is summed over all training samples.

3) HYPERPARAMETER TUNING

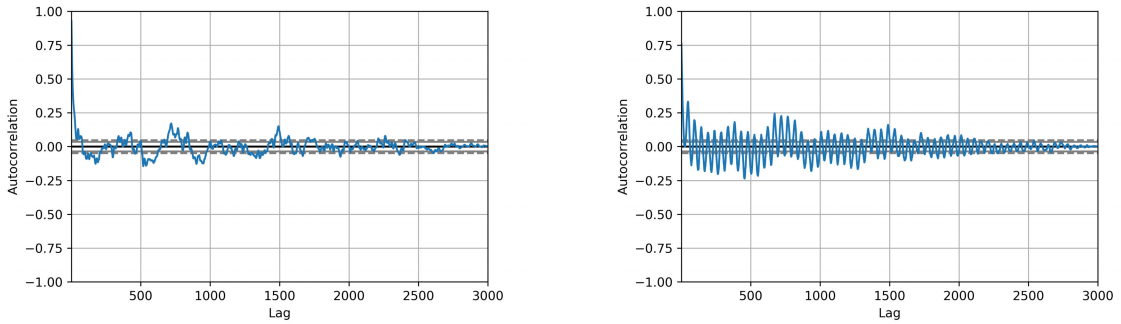
There are several hyperparameters within each of the three models whose value impacts the overall performance to a considerable extent. We conduct a grid search on the hyperparameter ranges using the validation set and determine the best combination of the hyperparameters that minimizes the validation L2-loss. The ranges and the best candidates of the hyperparameters are shown in Table 5. It can be summarized from the table that 48 time steps (24 hours) are typically sufficient for inferring a good forecast. Additionally, LSTNet and TPA-LSTM usually require a large number of kernels in their CNN components.

C. COMPARISON WITH OTHER MODELS

After successfully tuning the hyperparameters, we summarize the experimental results of the three models using the three metrics defined in section III-B1. The metrics are calculated based on the test dataset to eliminate biases. Shown in Table 6 are the reports of the related models making multiple short-term horizons ahead predictions. The effectiveness of the attention-based RNN models is compared with other established algorithms in the wind power forecasting literature.

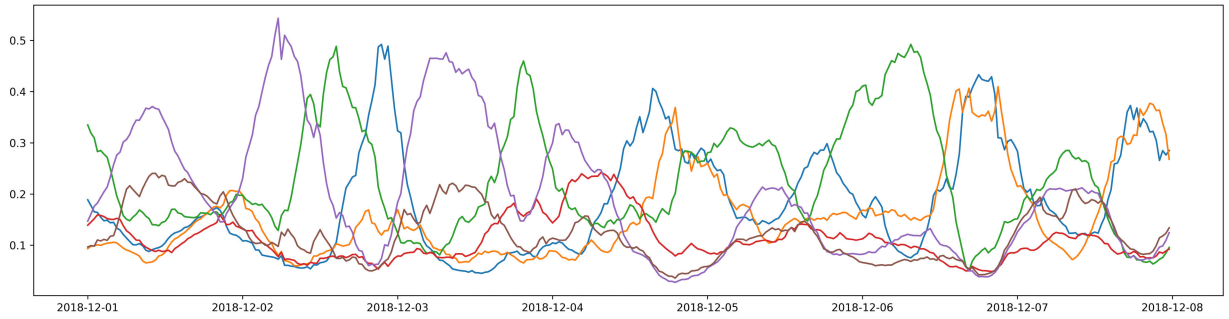
The key points summarized from the comparison are the following:

1) Since the support vector regression (SVR), extreme learning machine (ELM) and radial basis network (RBN) models only accept a vector or a single-variate sequence as input, we only use 48 steps of historical wind power for prediction. The hyperparameters are tuned by a grid search on the validation set. It is shown that the SVR performs the best among the three simple machine learning models, but its prediction error grows faster with the horizon h than the other two models.
2) The EMD-WT-SVR model attempts to capture the seasonal patterns within the wind power signal using a fixed number of denoised (by wavelet transformation) intrinsic mode functions (IMF, decomposed from the raw sequence using empirical mode decomposition). Multiple SVRs are trained independently for the multiple IMFs corresponding to the different intrinsic frequencies. To predict a test sample, the input sequence is decomposed into the same number of IMFs as required in the training process and denoised, and then their SVR results are combined to yield the final output. This model performs poorly overall, which may be due to the lack of clear periodic behavior within the wind power signal, illustrated via its autocorrelation function in Fig. 5, in contrast to the less fuzzy wind speed autocorrelation functions. Indeed, some of the most successful decomposition-based projects are dedicated to wind speed forecasting; see, e.g., [52], [53].
3) Both the vanilla LSTM and the attention-LSTM allow multivariate time series as input. The vanilla LSTM network is among the most effective models in our experiment and even achieves the best outcome when $h = 6$. The attention-LSTM is implemented by stacking a standard attention layer on top of a bidirectional layer and then another LSTM layer upon it. However,

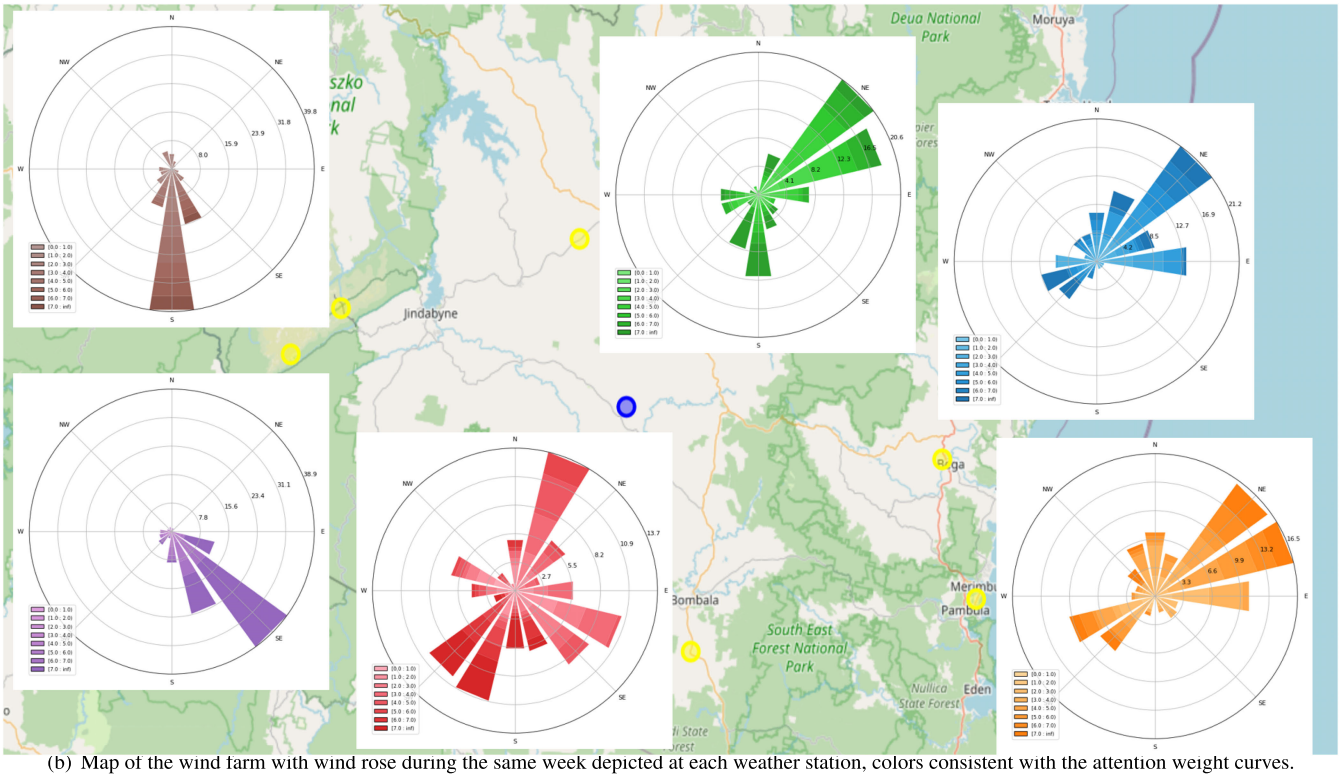


(a) ACF of the wind power. (b) ACF of the wind speed at the nearest weather station.

FIGURE 5. **Autocorrelation functions (ACF) of the wind power and speed series.**



(a) The spatial attention weights of the first week in December 2018 inferred from the learned DA-RNN encoder for 3-step ahead forecasting.



(b) Map of the wind farm with wind rose during the same week depicted at each weather station, colors consistent with the attention weight curves.

FIGURE 6. **Illustration of the spatial attention mechanism of DA-RNN.**

it is less efficient than the vanilla LSTM and grows even worse when the horizon increases. It is thus clear that adding a simple attention layer on top of an RNN is not sufficient for capturing complicated spatial-temporal information innate in the wind power forecasting task and may even jeopardize it. More complicated

FIGURE 7. The average temporal kernel from the learned TPA-LSTM for 3-step ahead forecasting. The lighter the color, the higher the attention weight is allocated.

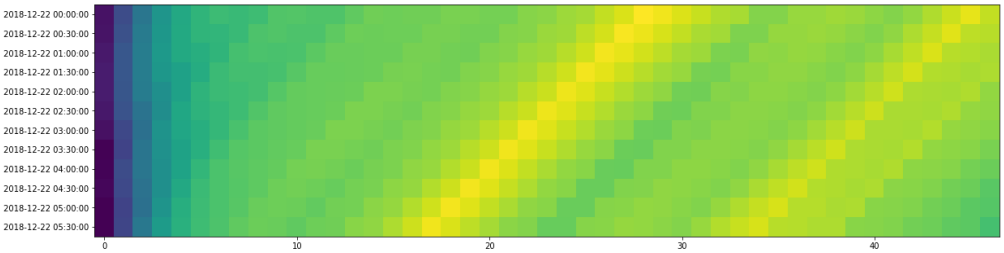


FIGURE 8. The temporal attention weights (at the last time step in each sample) tested from the first 6 hours of a day in December 2018 using the decoder of the learned 3-step ahead DA-RNN. The lighter the color, the higher the attention weight is allocated.

attention mechanisms deliberately designed for time-series forecasting are needed to improve the vanilla LSTM.

4) All three targeted models, namely, LSTNet, TPA-LSTM and DA-RNN, are among the best when forecasting shorter-term wind power. In particular, LSTNet is superior to the other models in the L2 criteria, whereas DA-RNN is the best in the L1 metrics. However, DA-RNN grows worse quickly as the horizon reaches 6, which can be the nature of the model because DA-RNN was originally designed for 1-step ahead prediction only. In addition, LSTNet and TPA-LSTM cannot outperform the vanilla LSTM for longer-term forecasting.

It can be seen from the experimental results that the existing attention-based RNN models can indeed achieve state-of-the-art wind power forecasting performance when the time horizon is small but become less competitive when h increases. In the next section, we interpret and analyze the ways these models propose to improve time-series forecasting using attention and other mechanisms via visualization and then suggest the reasons for their good and bad performance so that adjustments can be made to the existing models.

IV. SPATIAL-TEMPORAL PATTERNS: VISUALIZATION

Generally, all three attention-based RNN models extract temporal and spatial patterns in a separate manner, as summarized in Table 2. We analyze the way these mechanisms are operated by revealing the attention weights and kernels from the trained models.

A. SPATIAL PATTERNS

In contrast to the less dynamic convolutional kernels used in LSTNet and the LSTM layer (where the attention is based on the LSTM hidden space and is therefore even less interpretable) of TPA-LSTM, we find that the encoder of DA-RNN is worth noting for its flexible spatial attention mechanism. To understand how it works, the first fact we notice is that given one test sample, the spatial attention weights along the 48 time steps are almost invariant. Therefore, to observe how the weights change along the sliding samples, we can visualize their values at only the last time steps within the samples, as in the testing week: 2018-12-01 00:00 to 2018-12-07 23:30 of Fig. 6(a). There are apparent peaks and valleys along most curves, with some wind speed observations (purple, green, blue) being more significant than others in general. However, the physical rules behind the attention patterns are still unclear, even with the aid of the wind roses at each weather station in Fig. 6(b).

B. TEMPORAL PATTERNS

In terms of temporal patterns, we observe an interesting disagreement on the periodic effect of the wind series between the LSTNet model and the other two. Since the best hyperparameter combination of LSTNet confirms the superiority of the 24-skip RNN to its 16-skip counterparts, one may expect similar periodic patterns embodied in the convolutional kernels of TPA-LSTM and the decoder attention weights of DA-RNN. However, the 100 temporal kernels in TPA-LSTM are averaged to what can be seen in Fig. 7, without a clear 24-time step recurrence. For the decoder phase of DA-RNN, we find that usually in a consecutive period of time, testing samples will attend to the information of some fixed time steps in the historical series, yielding the sliding temporal attention weight vectors exemplified in Fig. 8. Other striking patterns are seen throughout the testing set, including versatile highest-attention intervals (17 time steps in this example) and diminishing attention on the beginning 1 to 3 time steps.

V. DISCUSSION AND CONCLUSION

We study and apply three state-of-the-art RNN models featuring attention mechanisms and other components to forecast short-term wind power generation. Overall, empirical results

enhance our confidence in applying the three attention-based RNN models to more complicated wind power forecasting tasks, especially under circumstances where exact wind speed prediction is not available and has to be inferred from the spatial-temporal information of the historical series.

In this article, the models are analyzed from the viewpoint of spatial-temporal pattern extraction and are tested on a wind farm in southeast Australia, when only limited wind speed information is gathered in the ambient weather stations of the targeted farm. The three models prove to exceed other competitors when the time horizon is small. Apart from that, by uncovering the attention weights and convolutional kernels of the models, some additional findings are learned so that quite a few heuristic adjustments can be made upon the existing models for the wind power forecasting task in particular.

First, although DA-RNN yields the best results in terms of MAE, we find that there can be considerable redundancy in the attention weights because they are compulsory at each of the time steps along the time window of a single input sample, even though they are almost equivalent. Second, the convolutional components of LSTNet and TPA-LSTM usually need many more kernels than the dimension of the input vector space to address the complicated nature of the time-series data when no dynamic mechanisms such as attention are involved.

To overcome these drawbacks, it may be wise to combine beneficial aspects of the three models, for instance, skipping some time steps in the spatial and temporal attentions of DA-RNN, as borrowed from the skip-RNN of LSTNet and involving temporal attentions into the TPA-LSTM, to leap beyond the CNN-only method of capturing temporal patterns. After all, the design of new variants of attention-based RNN models aimed at wind power forecasting deserves several other full-length papers, and we encourage our fellow researchers to consider this topic on another level in the near future.

REFERENCES

[1] J. Lee and F. Zhao, "Global wind report 2019," Global Wind Energy Council, Brussels, Belgium, Tech. Rep., Mar. 2020.

[2] *The Systems and Networks Team. 9 August 2019 Power Outage Report. Technical Report*, Office of Gas and Electricity Markets, London, U.K., Jan. 2020.

[3] Y. Zhao, L. Ye, W. Wang, H. Sun, Y. Ju, and Y. Tang, "Data-driven correction approach to refine power curve of wind farm under wind curtailment," *IEEE Trans. Sustain. Energy*, vol. 9, no. 1, pp. 95–105, Jan. 2018.

[4] M. Lydia, S. S. Kumar, A. I. Selvakumar, and G. E. P. Kumar, "A comprehensive review on wind turbine power curve modeling techniques," *Renew. Sustain. Energy Rev.*, vol. 30, pp. 452–460, Feb. 2014.

[5] L. Ye, Y. Zhao, C. Zeng, and C. Zhang, "Short-term wind power prediction based on spatial model," *Renew. Energy*, vol. 101, pp. 1067–1074, Feb. 2017.

[6] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagation errors," *Nature*, vol. 323, no. 6088, pp. 533–536, 1986.

[7] G. P. Zhang, "Time series forecasting using a hybrid ARIMA and neural network model," *Neurocomputing*, vol. 50, pp. 159–175, Jan. 2003.

[8] A. Jain and A. M. Kumar, "Hybrid neural network models for hydrologic time series forecasting," *Appl. Soft Comput.*, vol. 7, no. 2, pp. 585–592, Mar. 2007.

[9] S. Dasgupta and T. Osogami, "Nonlinear dynamic Boltzmann machines for time-series prediction," in *Proc. AAAI*, 2017, pp. 1833–1839.

[10] J. Connor, E. L. Atlas, and R. D. Martin, "Recurrent networks and narma modeling," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 4, 1991, pp. 301–308.

[11] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[12] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," 2014, *arXiv:1412.3555*. [Online]. Available: http://arxiv.org/abs/1412.3555

[13] G. Lai, W.-C. Chang, Y. Yang, and H. Liu, "Modeling Long- and short-term temporal patterns with deep neural networks," in *Proc. 41st Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Jun. 2018, pp. 95–104.

[14] T. Luong, H. Pham, and C. D. Manning, "Effective approaches to attention-based neural machine translation," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2015, pp. 1412–1421.

[15] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *Proc. Int. Conf. Learn. Represent. ICLR*, 2015, pp. 1–15.

[16] A. Kusiak, H. Zheng, and Z. Song, "Short-term prediction of wind farm power: A data mining approach," *IEEE Trans. Energy Convers.*, vol. 24, no. 1, pp. 125–136, Mar. 2009.

[17] H. Demolli, A. S. Dokuz, A. Ecemis, and M. Gokcek, "Wind power forecasting based on daily wind speed data using machine learning algorithms," *Energy Convers. Manage.*, vol. 198, Oct. 2019, Art. no. 111823.

[18] A. J. Smola and B. Schölkopf, "A tutorial on support vector regression," *Statist. Comput.*, vol. 14, no. 3, pp. 199–222, Aug. 2004.

[19] P. Seidel, A. Seidel, and O. Herbarth, "Multilayer perceptron tumour diagnosis based on chromatography analysis of urinary nucleosides," *Neural Netw.*, vol. 20, no. 5, pp. 646–651, Jul. 2007.

[20] H. B. Azad, S. Mekhilef, and V. G. Ganapathy, "Long-term wind speed forecasting and general pattern recognition using neural networks," *IEEE Trans. Sustain. Energy*, vol. 5, no. 2, pp. 546–553, Apr. 2014.

[21] M. Hossain, S. Mekhilef, F. Afifi, M. L. Halabi, L. Olatomiwa, M. Seyedmahmoudian, B. Horan, and A. Stojcevski, "Application of the hybrid anfis models for long term wind power density prediction with extrapolation capability," *PLoS ONE*, vol. 13, no. 4, pp. 1–31, 2018.

[22] A. Lahouar and J. Ben Hadj Slama, "Hour-ahead wind power forecast based on random forests," *Renew. Energy*, vol. 109, pp. 529–541, Aug. 2017.

[23] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001.

[24] A. A. Abdoos, "A new intelligent method based on combination of VMD and ELM for short term wind power forecasting," *Neurocomputing*, vol. 203, pp. 111–120, Aug. 2016.

[25] Y. Hao and C. Tian, "A novel two-stage forecasting model based on error factor and ensemble method for multi-step wind power forecasting," *Appl. Energy*, vol. 238, pp. 368–383, Mar. 2019.

[26] Y. Zhao, L. Ye, Z. Li, X. Song, Y. Lang, and J. Su, "A novel bidirectional mechanism based on time series model for wind power forecasting," *Appl. Energy*, vol. 177, pp. 793–803, Sep. 2016.

[27] G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew, "Extreme learning machine: Theory and applications," *Neurocomputing*, vol. 70, nos. 1–3, pp. 489–501, Dec. 2006.

[28] J. Shi, Y. Liu, Y. Yang, and W.-J. Lee, "Short-term wind power prediction based on wavelet transform-support vector machine and statistic characteristics analysis," in *Proc. IEEE Ind. Commercial Power Syst. Tech. Conf.*, May 2011, pp. 1–7.

[29] D. Liu and H. Li, "Short-term wind speed and output power forecasting based on WT and LSSVM," in *Proc. Int. Conf. Inf. Eng. Comput. Sci.*, Dec. 2009, pp. 1–4.

[30] N. E. Huang, Z. Shen, S. R. Long, M. C. Wu, H. H. Shih, Q. Zheng, N.-C. Yen, C. C. Tung, and H. H. Liu, "The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis," *Proc. Roy. Soc. London. Ser. A, Math., Phys. Eng. Sci.*, vol. 454, no. 1971, pp. 903–995, Mar. 1998.

[31] X. An, D. Jiang, M. Zhao, and C. Liu, "Short-term prediction of wind power using EMD and chaotic theory," *Commun. Nonlinear Sci. Numer. Simul.*, vol. 17, no. 2, pp. 1036–1042, Feb. 2012.

[32] K. Dragomiretskiy and D. Zosso, "Variational mode decomposition," *IEEE Trans. Signal Process.*, vol. 62, no. 3, pp. 531–544, Feb. 2014.

[33] Z. Qian, Y. Pei, H. Zareipour, and N. Chen, "A review and discussion of decomposition-based hybrid models for wind energy forecasting applications," *Appl. Energy*, vol. 235, pp. 939–953, Feb. 2019.

[34] A. Meng, J. Ge, H. Yin, and S. Chen, "Wind speed forecasting based on wavelet packet decomposition and artificial neural networks trained by crisscross optimization algorithm," *Energy Convers. Manage.*, vol. 114, pp. 75–88, Apr. 2016.
[35] H. Liu, H. Wu, and Y. Li, "Smart wind speed forecasting using EWT decomposition, GWO evolutionary optimization, RELM learning and IEWT reconstruction," *Energy Convers. Manage.*, vol. 161, pp. 266–283, Apr. 2018.
[36] E. López, C. Valle, H. Allende, E. Gil, and H. Madsen, "Wind power forecasting based on echo state networks and long short-term memory," *Energies*, vol. 11, no. 3, pp. 1–22, 2018.
[37] H. Gouveia, R. de Aquino, and A. Ferreira, "Enhancing short-term wind power forecasting through multiresolution analysis and echo state networks," *Energies*, vol. 11, no. 4, p. 824, Apr. 2018.
[38] L. Han, H. Jing, R. Zhang, and Z. Gao, "Wind power forecast based on improved long short term memory network," *Energy*, vol. 189, Dec. 2019, Art. no. 116300.
[39] F. Shahid, A. Zameer, A. Mehmood, and M. A. Z. Raja, "A novel wavenets long short term memory paradigm for wind power prediction," *Appl. Energy*, vol. 269, Jul. 2020, Art. no. 115098.
[40] A. Li and L. Cheng, "Research on a forecasting model of wind power based on recurrent neural network with long short-term memory," in *Proc. 22nd Int. Conf. Electr. Mach. Syst. (ICEMS)*, Aug. 2019, pp. 1–4.
[41] X. Shi, X. Lei, Q. Huang, S. Huang, K. Ren, and Y. Hu, "Hourly day-ahead wind power prediction using the hybrid model of variational model decomposition and long short-term memory," *Energies*, vol. 11, no. 11, pp. 1–20, 2018.
[42] J. Duan, P. Wang, W. Ma, X. Tian, S. Fang, Y. Cheng, Y. Chang, and H. Liu, "Short-term wind power forecasting using the hybrid model of improved variational mode decomposition and correntropy long short -term memory neural network," *Energy*, vol. 214, Jan. 2021, Art. no. 118980.
[43] J. Zhang, J. Yan, D. Infield, Y. Liu, and F.-S. Lien, "Short-term forecasting and uncertainty analysis of wind turbine power based on long short-term memory network and Gaussian mixture model," *Appl. Energy*, vol. 241, pp. 229–244, May 2019.
[44] S.-Y. Shih, F.-K. Sun, and H.-Y. Lee, "Temporal pattern attention for multivariate time series forecasting," *Mach. Learn.*, vol. 108, nos. 8–9, pp. 1421–1441, Sep. 2019.
[45] Y. Qin, D. Song, H. Chen, W. Cheng, G. Jiang, and G. W. Cottrell, "A dual-stage attention-based recurrent neural network for time series prediction," in *Proc. 26th Int. Joint Conf. Artif. Intell.*, Aug. 2017, pp. 2627–2633.
[46] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Commun. ACM*, vol. 60, no. 6, pp. 84–90, May 2017.
[47] Y. LeCun and Y. Bengio, *Convolutional Networks for Images, speech, and Time Series*, 1998.
[48] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. U. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 5998–6008.
[49] A. Miskelly. *Aemo Energy Generation Data*. Accessed: Jan. 10, 2021. [Online]. Available: https://anero.id/energy/data
[50] Australian Government Bureau of Meteorology. *Climate Data Online*. Accessed: Jan. 10, 2021. [Online]. Available: http://www.bom.gov.au/climate/data
[51] P. D. Kingma and J. L. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Represent. ICLR*, 2015, pp. 1–15.
[52] Y. Zhang, B. Chen, G. Pan, and Y. Zhao, "A novel hybrid model based on VMD-WT and PCA-BP-RBF neural network for short-term wind speed forecasting," *Energy Convers. Manage.*, vol. 195, pp. 180–197, Sep. 2019.
[53] L. Zhang, Y. Dong, and J. Wang, "Wind speed forecasting using a two-stage forecasting system with an error correcting and nonlinear ensemble strategy," *IEEE Access*, vol. 7, pp. 176000–176023, 2019.

BIN HUANG received the M.S. degree from the University of Melbourne, Melbourne, Australia, in 2018. He is currently a Teacher with the Department of Electrical and Electronic Engineering, Nanchang Institute of Technology. His current research interests include renewable energy power systems, big data of power grids, and artificial neural networks.

YUYING LIANG received the Ph.D. degree in electronic engineering from Mechanical Engineering College, Shijiazhuang, China, in 2001. She is currently a Professor with the Department of Electrical and Electronic Engineering, Nanchang Institute of Technology. Her current research interests include renewable energy power systems, artificial intelligence, and big data for the power grid.

XIAOLIN QIU received the Ph.D. degree in material science and engineering from the Huazhong University of Science and Technology, Wuhan, China, in 2006. He is currently a Professor with the Department of Energy and Environmental Engineering, Nanchang Institute of Technology. His current research interests include renewable energy power systems, machine learning, and data prediction.