Article

Application of Long-Short-Term-Memory Recurrent Neural Networks to Forecast Wind Speed

Meftah Elsaraiti * and Adel Merabet

Division of Engineering, Saint Mary's University, Halifax, NS B3H 3C3, Canada; adel.merabet@smu.ca

* Correspondence: meftah.elsaraiti@smu.ca; Tel.: +1-902-420-5101

Abstract: Forecasting wind speed is one of the most important and challenging problems in the wind power prediction for electricity generation. Long short-term memory was used as a solution to short-term memory to address the problem of the disappearance or explosion of gradient information during the training process experienced by the recurrent neural network (RNN) when used to study time series. In this study, this problem is addressed by proposing a prediction model based on long short-term memory and a deep neural network developed to forecast the wind speed values of multiple time steps in the future. The weather database in Halifax, Canada was used as a source for two series of wind speeds per hour. Two different seasons spring (March 2015) and summer (July 2015) were used for training and testing the forecasting model. The results showed that the use of the proposed model can effectively improve the accuracy of wind speed prediction.

Keywords: forecasting; long-short-term memory; multiple time series; RNN; wind speed

Citation: Elsaraiti, M.; Merabet, A. Application of Long-Short-Term-Memory Recurrent Neural Networks to Forecast Wind Speed. *Appl. Sci.* **2021**, *11*, 2387. <https://doi.org/10.3390/app11052387>

Academic Editor: Sergio Montelpare

Received: 16 December 2020

Accepted: 5 March 2021

Published: 8 March 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Forecasting wind speed is a very difficult challenge compared to other variables of the atmosphere, and this is due to their chaotic and intermittent nature which causes difficulty in integrating wind power to the grid. As wind speed is one of the most developed and cheapest green energy sources, its accurate forecasting in the short term has become an important matter and has a decisive impact on the electricity grid. Both dynamic and statistical methods, and some hybrid methods that couple the two methods, were applied to forecast short term wind speed. Running high-resolution numerical weather prediction (NWP) models requires an understanding of many of the basic principles that support them, including data assimilation, knowledge on how to estimate the NWP model in space and time, and how to perform validation and verification of forecasts. This can be costly in terms of computational time. Reliable methods and techniques for forecasting wind speed are becoming increasingly important for characterizing and predicting wind resources [1]. The main purpose of any forecast is to build, identify, tune and validate time series models. Time series forecasting is one of the most important applied problems of machine learning and artificial intelligence in general, since the improvement of forecasting methods will make it possible to more accurately predict the behavior of various factors in different areas. Traditionally, such models are based on the methods of statistical analysis and mathematical modeling developed in the 1960s and 1970s [2]. The ARIMA model was used to forecast wind speed using common error ratio measures for model prediction accuracy [3]. Recently, deep learning in the machine learning community has gained significant popularity as it is considered a general framework that facilitates the training of deep neural networks with many hidden layers [4]. The availability of large datasets combined with the improvement in algorithms and the exponential growth in computing power led to an unparalleled surge of interest in the topic of machine learning. These methods use only historical data to learn the random dependencies between the past and the future. Among these methods, recurrent neural networks (RNNs) which are designed to learn a sequence of data by traversing a hidden state from one step of the sequence to

the next, combined with the input, and routing it back and forth between the inputs [5]. Long short-term memory based recurrent neural network (LSTM-RNN) has been used to forecasting 1 to 24 h ahead wind power [6]. A comparison was made between LSTM, extreme learning machine (ELM) and SVM. The results have shown that deep learning approaches are more effective than the traditional machine learning methods in improving prediction accuracy through the directional loop neural network structure and a special hidden unit [7]. Studies have suggested that coupling NWP models and artificial neural networks could be beneficial and provide better accuracy compared to conventional NWP model downscaling methods [8]. Numerical weather prediction method (NWP) is one of the most used methods, and it is suitable for long term rather than short term and medium term forecast due to the large amount of computation [9]. There has been analysis of the wind speed forecasting accuracy of the recurrent neural network models, and they have presented better results compared to the univariate and multivariate ARIMA models [10]. Linear and non-linear autoregressive models with and without external variables were developed to forecast short-term wind speed. Three performance metrics, MAE, RMSE and MAPE were used to measure the accuracy of the models [11]. LSTM method was used to forecast wind speed and the results were compared to traditional artificial neural network and autoregressive integrated moving average models, the proposed method proved better results [12]. Long Term Memory Model (LSTM) was done to short-term spatio-temporal wind speed forecast for five locations using two-year data for historical wind speed and auto-regression. The model aimed to improve forecasting accuracy on a shorter time horizon. For example, using LSTM for two or three hours can forecast horizons extending up to fifteen days using an NWP model that updates itself typically with a frequency of six hours [13]. The training speed of RNNs for predicting multivariate time series is usually relatively slow especially when used with large network depth. Recently a variety of RNN called long-term memory (LSTM) has been favored due to its superior performance during the training phase by better solving vanishing and exploding gradient problems of standard RNN architecture [14,15]. Long short-term memory (LSTM) and temporal convolutional networks (TCN) models were proposed for data-driven lightweight weather forecasting, and their performance was compared with classic machine learning approaches (standard regression (SR), support vector regression (SVR), random forest (RF)), statistical machine learning approaches (Autoregressive integrated moving average (ARIMA), vector auto regression (VAR), and vector error correction model (VECM)), and dynamic ensemble method (Arbitrage of forecasting expert (AFE)). The results of the proposed model demonstrate its ability to predict effective and accurate weather [16]. Despite the continuous development of research on the LSTM algorithm in long-term prediction of wind speed, the traditional RNN algorithm for prediction is still preferred in most research. In this paper, emphasis has been placed on the application of the LSTM algorithm in the field of forecasting wind speed and a comparison between prediction efficiency and accuracy of the algorithm under different wind speed time series were used for training and testing.

2. Methodology and Data Source

2.1. Data Sources

In this study, the proposed model was implemented only for short-term forecasting of wind speed in order to avoid the high computational time of making dynamical downscaling by using NWP models such as Weather Research and Forecasting Model (WRF). Wind speed data from the Halifax Dockyard station in Nova Scotia, which is located: Latitude 44.66° N, Longitude 63.58° W. Wind speed was measured at a height of 3.80 m, and it was used as a source for two different seasons, spring (March 2015) and summer (July 2015), as shown in Figure 1. For both seasons, the data recorded at each hour, (576 reads/24 days) as observations and (168 readings/7 days), respectively, as the training and testing groups. The proposed LSTM implementation fits well with the time series dataset, which can improve the convergence accuracy of the training process.

Figure 1. Time series of the wind speed in Halifax, Canada.

2.2. Recurrent Neural Networks

Recurrent neural networks (RNNs) are sequential data neural networks whose purpose are to predict the next step in a sequence of observations relative to previous steps in the same sequence. RNNs contain hidden layers distributed across time, which allows them to store information obtained in previous stages of reading serial data. Wind speed depends on the short and long term. The simple RNN model is unable to handle long-term time dependencies. One problem that arises from the unfolding of an RNN is that the gradient of some of the weights starts to become too small or too large if the network is unfolded for too many time steps. This phenomenon is called the vanishing gradients problem, and it can store the short-term memory only because it includes the functions of activating the hidden layer of the previous time step only and this causes the loss of information in the long term [17,18]. A type of network architecture that solves this problem is the LSTM. In a typical implementation, the hidden layer is replaced by a complex block of computing units composed by gates that trap the error in the block, forming a so-called “error carrousel” [5]. Figure 2 shows the RNN structure where the output of the previously hidden layer is input to the current hidden layer. The RNN model is expressed by

$$\begin{cases} h_t = \tanh(w_{hx}x_t + w_{hy}y_{t-1}) \\ y_t = \sigma(wh_t) \end{cases} \quad (1)$$

where x_t is the input, h_t is the state value of the hidden layer, y_t is value at the output layer at time t , w_{hx} is the weight from the input layer, w_{hy} is the weight for the delayed output at time $t - 1$, $\tanh$ is the hyperbolic tangent as the activation function at the hidden layer, and σ is the sigmoid function as the activation function at the output layer.

2.3. Long Short Term Memory

Long short-term memory networks are a type of recurrent neural network (RNN) designed to avoid the problem of long-term dependence, where each neuron contains a memory cell capable of storing the previous information used by the RNN or forgetting it if necessary [19]. Currently, it is widely used with success in time series prediction problems. LSTM-RNN was designed from a memory cell that stores long-term dependencies. In addition to the memory cell, the LSTM cell contains an input gate, output gate and forget gate. Each gate in the cell receives the current input x_t , the hidden state h_{t-1} at the previous moment and the state information C_{t-1} of the cell’s internal memory to perform various operations and determine whether to activate using a logic function. The state h_t of the unit,

the output at time t and the input hidden state at time t_1 are determined by non-linearly activating $\tanh()$ and the information of the output gate.

Figure 2. RNN structure.

The processing of a time point inside a LSTM cell could be described as below:

- (1) The unwanted information in the LSTM is identified and thrown away from the cell state through the sigmoid layer called forget gate layer defined by

$$f_t = \sigma[w_f(h_{t-1}, x_t) + b_f] \quad (2)$$

where, w_f is the weight, h_{t-1} is the output from the previous time stamp, x_t is the new input, b_f is the bias.

- (2) The new information that will be stored in the cell state is determined and updated by the sigmoid layer called the input gate Layer. Next, a $\tanh$ layer creates a vector of new candidate values that could be added to the state

$$i_t = \sigma[w_i(h_{t-1}, x_t) + b_i] \quad (3)$$

$$\hat{c}_t = \tanh[w_c(h_{t-1}, x_t) + b_c] \quad (4)$$

- (3) The old cell state, c_{t-1} , is updated into the new cell state c_t . The cell state is updated as follows

$$c_t = f_t * c_{t-1} + i_t * \hat{c}_t \quad (5)$$

- (4) A sigmoid layer will be run to decide what parts of the cell state are going to the output. It is expressed by

$$o_t = \sigma[w_o(h_{t-1}, x_t) + b_o] \quad (6)$$

$$h_t = o_t * \tanh(c_t) \quad (7)$$

The LSTM structure is illustrated in Figure 3.

2.4. Forecast Validation

In order to analyze the accuracy of the models, among the most commonly used parameters for estimating wind speed predictions is root mean square error (RMSE) [20]. It is expressed by

$$RMSE = \sqrt{\frac{\sum_{i=1}^N (y_{ai} - y_{fi})^2}{N}} \quad (8)$$

where, N the number of data, y_{ai} is the actual value and y_{fi} is the forecast value.

Figure 3. Description of LSTM cell configuration.

3. Results and Discussion

In this study, the MATLAB software (R2019b) was used for the training process of the LSTM, which is an advanced architecture for RNN to predict the values of future time steps of a sequence. The sequence regression network was trained to the LSTM sequence, where responses are training sequences with changing values in one time step. That is, for each time step of the input sequence, the LSTM network learns to predict the value of the next time step. In this work, to evaluate the effectiveness and applicability of the proposed model in a comprehensive and systematic manner, two series of data were selected for wind speed for two different seasons due to their different climatic characteristics, which are, respectively, spring (March 2015) and summer (July 2015). Each data series was divided into 1–576 (24 days) as observations and 577–744 (7 days), respectively, as the training and test groups. Training data have been standardized to have zero mean and unit variance at prediction time to prevent training from divergence. The best training parameter to obtain the lowest RMSE is found using an initial learning rate of 0.005. Figures 4 and 5 show the comparison of observed values with predicted values of hourly wind speed series collected in the spring (1–31 March, 2015) and Summer (1–31 July 2015), respectively, for evaluation of the LSTM, which was trained once (the value of the previous prediction) and reused to make a prediction for each time step between predictions which is represented by the Equations (2)–(4). This meaning that no updates are made once the model first fits the training data, and the model in this case is called the fixed model because there are no updates. LSTM network training options was selected for 200 hidden modules. The initial learning rate is 0.005, and the maximum number of iterations is fixed at 250. The gradient threshold is set to 1 to prevent the gradients from exploding. The learning rate is dropped after 125 epochs by multiplying by a factor 0.2.

In both Figures 6 and 7, the LSTM has been updated with new data for the time series forecasting by using the predicted values and the updated state values from the test set and made available to the model for the forecast on the next time step. Specially, the modified LSTM adopted C_{t-1} to the input, forget, and output gates. This is because every time the LSTM proceeds, C_{t-1} affects the input, forget, and output of the LSTM. All predictions are collected in the test data set and an error score is calculated to summarize the skill of the model. The root mean square error (RMSE) is used because it penalizes large errors

and results in a score in the same units as the forecasted data, which is wind speed per hour. Here, the predictions are more accurate when updating the network state with the observed values instead of the predicted values. From the results, it is observed that in both series Spring (March 2015) and Summer (July 2015) the value of RMSE dropped by 4.5845 and 4.9392, respectively, when using the LSTM update, and this is due to the different characteristics of each. In this work and based on various previous studies according to different models, it is noted that the accuracy of prediction models differs according to the different characteristics of the information, and therefore, until now, no model has been reached that works with the same accuracy with different information. Table 1 shows the errors for the two data series (July 2015) and (March 2015) in the proposed LSTM model using the RMSE error metrics.

Figure 4. Wind speed prediction of existing LSTM for Spring (March 2015) data: (a) Comparison of observed values and forecast values; (b) Error and RMSE value values.

Figure 5. Wind speed prediction of existing LSTM for Summer (July 2015) data: (a) Comparison of observed values and forecast values; (b) Error and RMSE value values.

Figure 6. Wind speed prediction of the proposed LSTM for Spring (March 2015) data: (a) Comparison of observed values and forecast values; (b) Error and RMSE value values.

Figure 7. Wind speed prediction of proposed LSTM for Summer (July 2015) data: (a) Comparison of observed values and forecast values; (b) Error and RMSE value values.

Table 1. Evaluation of two different time series data in the LSTM model.

	Time Series	
	Spring (1–31 March 2015)	Summer (1–31 July 2015).
RMSE	8.5128	4.7796

4. Conclusions

Accurate forecasting model of wind speed sources is necessary to provide essential information to empower grid operators and system designers in generating an optimal wind power plant, and to balance the supply and demand in the energy market. In this study, modified long short-term memory (LSTM) has been proposed to forecast wind speed. Since the actual value of the time steps between predictions can be accessed, the wind

speed is predicted by updating the network state in each prediction using the observed value instead of the gates. This is because every time the LSTM proceeds, the cell state affects the input, forget, and output of the LSTM. The results of the model demonstrated improved accuracy in predicting wind speed.

Author Contributions: Conceptualization, M.E.; methodology, M.E.; software, M.E.; validation, M.E.; formal analysis, M.E. and A.M.; writing—Original draft preparation, M.E.; writing—Review and editing, A.M.; supervision, A.M.; project administration, A.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded in part by Libyan Education Ministry, grant number 3772.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

- Monfared, M.; Rastegar, H.; Kojabadi, H.M. A new strategy for wind speed forecasting using artificial intelligent methods. *Renew. Energy* **2009**, *34*, 845–848. [\[CrossRef\]](#)
- Box, G.E.; Jenkins, G.M.; Reinsel, G.C.; Ljung, G.M. *Time Series Analysis: Forecasting and Control*, 5th ed.; John Wiley & Sons: Hoboken, NJ, USA, 2015; pp. 129–171.
- Elsaraiti, M.; Merabet, A.; Al-Durra, A. Time Series Analysis and Forecasting of Wind Speed Data. In Proceedings of the IEEE Industry Applications Society Annual Meeting, Baltimore, MD, USA, 29 September–3 October 2019; pp. 1–5.
- Yadav, A.P.; Kumar, A.; Behera, L. RNN based solar radiation forecasting using adaptive learning rate. In *The International Conference on Swarm, Evolutionary, and Memetic Computing*; Springer: Cham, Switzerland, 2013; pp. 442–452.
- Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural Comput.* **1997**, *9*, 1735–1780. [\[CrossRef\]](#) [\[PubMed\]](#)
- Kumar, D.; Mathur, H.D.; Bhanot, S.; Bansal, R.C. Forecasting of solar and wind power using LSTM RNN for load frequency control in isolated microgrid. *Int. J. Model. Simul.* **2020**. [\[CrossRef\]](#)
- Shi, X.; Lei, X.; Huang, Q.; Huang, S.; Ren, K.; Hu, Y. Hourly day-ahead wind power prediction using the hybrid model of variational model decomposition and long short-term memory. *Energies* **2018**, *11*, 3227. [\[CrossRef\]](#)
- Rodrigues, E.R.; Oliveira, I.; Cunha, R.; Netto, M. DeepDownscale: A Deep Learning Strategy for High-Resolution Weather Forecast. In Proceedings of the 2018 IEEE 14th International Conference on e-Science (e-Science), Amsterdam, The Netherlands, 29 October–1 November 2018; IEEE: New York, NY, USA, 2018; pp. 415–422.
- Chen, N.; Qian, Z.; Nabney, I.T.; Meng, X. Wind power forecasts using Gaussian processes and numerical weather prediction. *IEEE Trans. Power Syst.* **2013**, *29*, 656–665. [\[CrossRef\]](#)
- Cao, Q.; Ewing, B.T.; Thompson, M.A. Forecasting wind speed with recurrent neural networks. *Eur. J. Oper. Res.* **2012**, *221*, 148–154. [\[CrossRef\]](#)
- Lydia, M.; Kumar, S.S.; Selvakumar, A.I.; Kumar, G.E. Linear and non-linear autoregressive models for short-term wind speed forecasting. *Energy Convers. Manag.* **2016**, *112*, 115–124. [\[CrossRef\]](#)
- Ghaderi, A.; Sanandaji, B.M.; Ghaderi, F. *Deep Forecast: Deep Learning-Based Spatio-Temporal Forecasting*; Cornell University: Ithaca, NY, USA, 2017.
- Özen, C.; Kaplan, O.; Özcan, C.; Dinç, U. Short Term Wind Speed Forecast by Using Long Short Term Memory. In Proceedings of the 9th International Symposium on Atmospheric Sciences ATMOS 2019, Istanbul, Turkey, 23–26 October 2019.
- Kim, T.S.; Reiter, A. Interpretable 3D Human Action Analysis with Temporal Convolutional Networks. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Honolulu, HI, USA, 21–26 July 2017; IEEE: New York, NY, USA, 2017; pp. 1623–1631.
- Lea, C.; Flynn, M.D.; Vidal, R.; Reiter, A.; Hager, G.D. Temporal convolutional networks for action segmentation and detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 156–165.
- Sandhu, K.S.; Nair, A.R. A Comparative Study of ARIMA and RNN for Short Term Wind Speed Forecasting. In Proceedings of the 2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT), Kanpur, India, 6–8 July 2019; IEEE: New York, NY, USA, 2019; pp. 1–7.
- Squartini, S.; Hussain, A.; Piazza, F. Preprocessing Based Solution for the Vanishing Gradient Problem in Recurrent Neural Networks. In Proceedings of the 2003 International Symposium on Circuits and Systems (ISCAS), Bangkok, Thailand, 25–28 May 2003; IEEE: New York, NY, USA, 2003; Volume 5, p. 5.

18. Batbaatar, E.; Park, H.W.; Li, D.; Li, M.; Ryu, K.H. DeepEnergy: Prediction of appliances energy with long-short term memory recurrent neural network. In *Asian Conference on Intelligent Information and Database Systems*; Springer: Cham, Switzerland, 2018; pp. 224–234.
19. Sak, H.; Senior, A.; Beaufays, F. *Long Short-Term Memory Based Recurrent Neural Network Architectures for Large Vocabulary Speech Recognition*; Cornell University: Ithaca, NY, USA, 2014.
20. Ghofrani, M.; Suherli, A. Time series and renewable energy forecasting. *Time Ser. Anal. Appl.* **2017**, 2017, 77–92.