International Research Journal of Innovations in Engineering and Technology(IRJIET)

ISSN (online): 2581-3048
Volume 7, Issue 10, pp 376-392, October-2023
https://doi.org/10.47001/IRJIET/2023.710051

A Short-Term Hybrid Wind Power Forecasting Approach using BiLSTM_EMD and the Avrami Curve

[1]Joseph N. Mathenge, [2]John N. Nderu, [3]David K. Murage

[1]MSc Student, Department of Electrical and Electronic Engineering, Jomo Kenyatta University of Agriculture and Technology (JKUAT), Nairobi, Kenya
[2, 3]Professor, Department of Electrical and Electronic Engineering, Jomo Kenyatta University of Agriculture and Technology (JKUAT), Nairobi, Kenya

***Abstract* - In recent times, the world has inclined towards using renewable energy sources since they are emission-free, occur freely in nature, and unlike fossil fuels, cannot be depleted. Wind power is one such renewable energy source that has attracted a lot of research and interest in the power industry. With the growing quantities of wind power generation incorporated into power systems, grid reliability is at risk since wind power is highly intermittent. Wind power forecasts facilitate incorporation of wind in a grid's power mix more efficiently and reduce the quantity of power reserves allocated to cater to the intermittency of wind. This makes adopting more wind power resources into the grid more economical. In this paper, a novel approach to wind power forecasting is developed using Bidirectional Long Short-Term Memory Neural Networks (BiLSTM) hybridized with Empirical Mode Decomposition (EMD) then enhanced with a wind power curve layer derived from the Avrami Equation. The developed model was tested on an online-based dataset and compared with the traditional LSTM and other hybrid LSTM-data decomposition models. Using the developed BiLSTM + EMD enhanced with an Avrami Power Curve layer, wind power prediction improved by at least 50% compared to hybrid BiLSTM-data decomposition models. Modelling and coding were performed in MATLAB R2019a.**

Keywords: Wind Power Forecasting, Bi-directional Long Short Term Memory Network, Empirical Mode Decomposition, The Avrami Equation

I. INTRODUCTION

Conventionally, power grids have been dominated by a few centrally located high-output generators that produce power to meet the entire load in the grid. Hydro, geothermal, coal, and nuclear power generators have been the primary electrical power sources in the grid. In recent times, however, there has been a gradual paradigm shift from the exclusive use of these classical electrical power sources to adopting more renewables in the power mix to meet the growing demand. This is in line with ensuring that the power industries reduce their environmental carbon footprint [1]. In addition, many grids have continued leaning towards more reliable sources of energy that do not stand to get depleted soon. Fossil fuel deposits are running low, and with increasing populations, they stand to get exhausted at some point in the future [2]. Initiatives such as the Paris agreement and the Net Zero initiative have gone a long way in promoting the adoption of more renewables in a grid's energy mix and reducing global warming [3].

Solar, wind, and biofuels have attracted the most interest in this quest to go green in the 20th century [4]. Solar and wind occur freely in nature and can be harnessed anywhere in the world. This has made them an attractive area of research for scholars globally who aim to maximize the benefits the world can reap from these free resources. Governments have incentivized energy institutions, including giving tax waivers and funding to encourage the exploration and exploitation of solar and wind energy. There have been technological advancements in the hardware technology used in solar panels, battery storage systems, and wind turbines. These advancements aim to improve the efficiency of renewable systems to ensure maximum power conversion from solar irradiation and wind into electric power. Solar energy technologies have been steadily developing since solar is more reliable than wind energy. Solar energy is more predictable because it is naturally experienced during the day, and one can plan for an alternative for use at night. On the other hand, the wind is highly stochastic, making wind power highly unpredictable. In this regard, wind power is conventionally considered to be a dispatchable resource by power system operators.

Wind power's intermittent and unpredictable nature has spurred significant interest in developing accurate prediction algorithms and models to ensure the optimal integration of wind as a dispatchable resource in a grid's energy mix. Wind power uncertainty is a risk to grid stability in cases where wind power accounts for a significant amount of the grid's total installed capacity. Voltage stability at the point of

International Research Journal of Innovations in Engineering and Technology(IRJIET)

ISSN (online): 2581-3048

Volume 7, Issue 10, pp 376-392, October-2023

https://doi.org/10.47001/IRJIET/2023.710051

injection and transient stability become issues of concern regarding wind power integration into the grid. This challenge has called for the development of fast and accurate optimization tools capable of accurately predicting wind power, allowing for the best planning of grid operations and optimal usage of this free power as and when available.

Short-term forecasts extending from 1 hour to 72 hours are essential to plan the operations of a power system in terms of unit commitment and wind power dispatch. This helps ensure operations continuity and boosts the power system's reliability. In addition, accurate wind power prediction helps optimize the grid's operational costs by optimizing the grid's power mix at any given time and optimizing any balancing power required to take care of the uncertainty of wind power. The developed hybrid algorithm in this paper helps improve the accuracy of wind power forecasts, thus making integrating large-scale wind power more manageable and economical.

II. LITERATURE REVIEW

2.1 Harnessing the Power of Wind

Wind is the movement of air due to the sun's uneven heating of the earth's surface. In essence, there can never be uniform heating of the earth's surface due to the varying altitudes of topological features; hence, the wind always exists in nature. Wind is random and uncontrollable, and it is a renewable energy resource that occurs freely in nature virtually in all places around the globe. It is relatively simple to harness and has little to no negative environmental impact. There has been tremendous growth in incorporating renewables such as wind energy in present-day power systems driven by public policies that champion climate change and reduction of the carbon footprint of various sectors on the environment.

In some countries, such as the United States, which have well-developed power systems, fossil fuels, mainly coal and natural gas, are among the largest sources of electricity [5]. Deposits of these fossil fuels are limited in nature and hence stand to get depleted with the growing power demand. In addition, power stations running on fossil fuels are major emitters of CO2 as a byproduct [6]. CO2 is a greenhouse gas that has been the most significant contributor to global warming in the last century. Recently, the world has sought to reduce its CO2 emissions in the environment to reduce global warming. According to the Paris agreement, the world is looking to reduce global warming to below 2^{o}C [3]. Renewables offer this much-needed path towards achieving net-zero emissions in the atmosphere since they are clean energy sources with minimal adverse environmental impacts.

According to the global wind report, the global wind industry experienced its most remarkable growth in 2020, where the total new wind power installations surpassed the 90GW mark that year. This was a 53% increase in new installations compared to the year 2019. The cumulative wind power installations as of 2020 now stood at 743GW, with China reporting cumulative grid-connected wind power of over 272GW, making it the global leader in wind power installations. Of the 93GW of new wind power installations in 2020, 88.4GW (95.3%) of this power came from onshore installations, which shows the popularity of onshore wind power plants worldwide [7]. Figure 1 below shows the New global year-over-year wind power installations over the last seven years [7][8]:

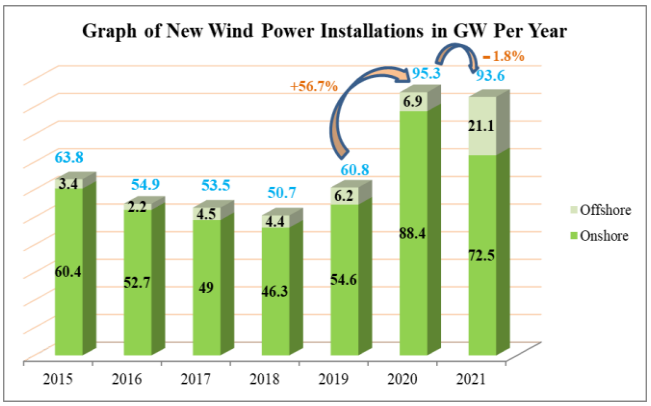

Figure 1: New Global Wind Power Installations in GW per Year

One of the biggest challenges in using wind turbines is the fast change in the power output of these turbines [9]. The wind turbine power output depends on the wind speed – an irregular and random variable, making wind power an intermittent variable. Due to this variability of wind power, for the efficient running of wind power units connected to the grid, accurate forecasts of up to 48 hours are crucial [10]. The knowledge and information regarding the future expected power generation from wind turbines is vital for the smooth operation of a power system.

2.2 The Wind Power Equation and Wind Turbines' Power Characteristics

Harnessing wind power entails taping the kinetic energy in the wind and converting it into electric power. Extracting all the kinetic energy in a wind stream is practically impossible. Mathematically, if all the kinetic energy from a wind stream were extracted, the stream's speed after the turbine would be zero [11]. This would imply a disruption in the flow of new wind streams across the wind turbine. German physicist Albert Betz researched on the maximum possible power that can be extracted from a mass of wind and called it the Betz limit. According to the Betz limit, a wind turbine cannot convert

International Research Journal of Innovations in Engineering and Technology(IRJIET)

ISSN (online): 2581-3048

Volume 7, Issue 10, pp 376-392, October-2023

https://doi.org/10.47001/IRJIET/2023.710051

more than 59.3% of the kinetic energy in the wind into mechanical energy to turn the generator rotor [12]. This is illustrated in *Figure 2* below:

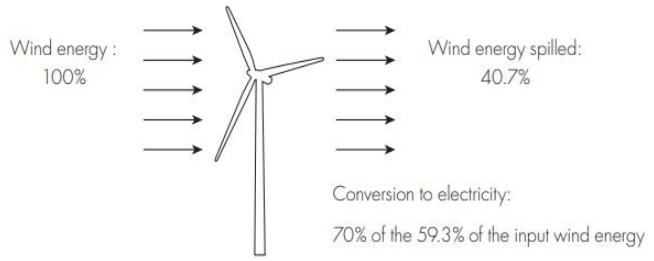

Figure 2: Wind Power Extraction Illustration According to the Betz Limit.

The equation for the average power generated by a wind turbine is given by:

$$P_{av} = \frac{1}{2} C_p \rho A v^3 \quad (1)$$

Equation 1, above shows the cubic relationship between wind speed and wind power, making the wind speed, v, the most significant variable in determining the power output of a wind turbine [13]. A graph of P_{av} against v is known as the wind power generation curve. There are three critical zones in a wind power generation curve:

i. Cut in speed
ii. Constant Cp region
iii. Constant Power output region.

These zones are illustrated in Figure 3 below [14]:

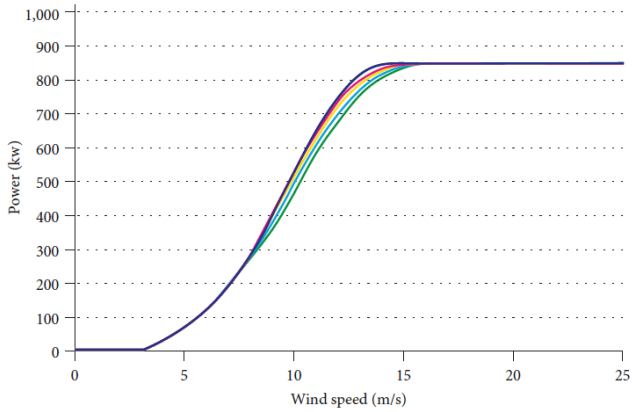

Figure 3: Wind Power Curve for a Vestas V52 - 850kW Wind Turbine

2.3 Wind Power Forecasting Approaches

Wind power forecasting approaches can be categorized into two based on the forecasting data used [15]:

a. **The indirect method** - Entails forecasting future wind speed values, after which appropriate transformations are applied to get the corresponding values of wind power.

b. **The direct method** - Here, wind power is forecast directly without predicting wind speed.

According to the approach used in wind power prediction, the various methods can be classified into:

i. **Persistence Methods** - Assumes that wind power at a given observation point in the future, t+1 is the same as the measured power at the current time, t. The accuracy of this method deteriorates fast with an increase in the prediction scope/timescale [16]. If the current wind power at time t is P(t), then wind power at a future time t+Δt is assumed to be P(t+Δt)= P(t). For this assumption to hold, the forecasting time frame needs to be very small; hence its accuracy is only viable for ultra-short-term time frames.

ii. **Physical Methods** - Utilize complex mathematical models to perform their prediction. Physical methods deal with numerical weather prediction data from meteorological services that study the behaviour and physics of the lower boundary of the atmosphere to predict future weather patterns. This method considers the topology of a wind farm's location. The main disadvantage of physical models is that they need very accurate online and offline data [17]. This method is also computationally tasking and requires considerable computing resources [16]. Physical methods present better performance for medium-term and long-term forecasts [18]. They, however, have poor performance in shorter-term forecasts.

iii. **Statistical Methods** - Are based on establishing the linear and non-linear relationships between weather parameters such as wind speed, direction, and temperature with the generated wind power. Statistical methods require historical data to train the models to determine these relationships. Statistical models are ideal for short-term wind prediction, and prediction accuracy decreases as the forecast time scale increases [16]. They are simple to model and require short computational durations. The models are occasionally tuned by comparing the predicted and measured power to ensure that the forecasts continuously improve as time progresses. Statistical models are divided into time series models and neural network models.

a. *Time series models* - These are mathematical models proposed by George Box and Gwilym Jenkins and use historical data to develop a mathematical model that forecasts future time series instances [16]. The Box - Jenkins models are based on the ARIMA models to find the best fit for a time series based on historical values [19]. The general form of the model is described as follows:

International Research Journal of Innovations in Engineering and Technology(IRJIET)

ISSN (online): 2581-3048

Volume 7, Issue 10, pp 376-392, October-2023

https://doi.org/10.47001/IRJIET/2023.710051

$$X_t = \sum_{i=1}^{p} \varphi_i X_{t-i} + \alpha_t - \sum_{j=1}^{q} \theta_j \alpha_{t-j} \qquad (2)$$

Where:

X_t - is the wind power forecast at the time t

φ_i - is the autoregressive parameter

θ_j - moving average parameter

α_t - white noise

p - order of the autoregressive model

q - order of the moving average model

If $p = 0$, the model becomes a Moving Average (MA) model, and if $q = 0$, the model becomes an Auto-Regressive (AR) model.

b. ANN Models

ANNs are some of the most popular methods used in wind power forecasting [16]. The strengths of ANNs come from their ability to establish non-linear relationships between input features and the predicted variable(s) without any need for mathematical formulations [19].

iv. Hybrid Methods

It involves combining physical and statistical models or two or more physical or statistical models. The aim is to have one model improve on the weakness of the other model they are hybridized with. In doing this, the overall prediction accuracy of the hybrid is improved. created file, highlight all of the contents and import your prepared text file. You are now ready to style your paper.

2.4 The Artificial Neural Network (ANN)

2.4.1 Basic Structure of the ANN

The ANN is a non-linear mapping architecture mimicking a human's central nervous system operation. The basic structure of an ANN is the neuron which mimics the biological neuron. The first artificial neural network model was created by McCulloch and Pitts back in 1943 [20]. The ideas behind that first model are still in use today. An ANN can also be defined as a massively paralleled processor consisting of simple processing units that can learn by experience and use that knowledge to make future decisions [21]. The ANN is a robust prediction tool for situations where the relationship between data is unknown and seeks to be established. ANNs learn from any correlated patterns observed between input data sets and target values in the training dataset. Once trained, an ANN can then predict subsequent future outcome(s) from the determined pattern/connections it establishes from the training dataset.

ANNs are well suited to deal with data that is considered vague, noisy, and at times changes erratically. Wind power exhibit such characteristics due to the fluctuating nature of wind speed. Wind power has a quasi-cubic relationship with wind speed. Thus, ANNs are ideal for a wind speed/ power data prediction tool since such data is complex and often non-linear [20]. Since a neural network consists of highly interconnected nodes, it can learn and generalize training patterns from the training data, just like the human brain. This vital learning capability of ANNs is the main advantage that makes it the best tool for forecasting applications.

The neuron is the basic information processing unit in a neural network, and its architecture is shown in *Figure 4*below [21]:

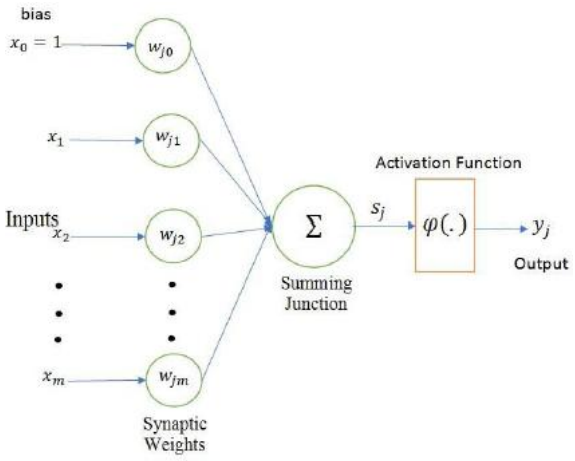

Figure 4: Basic Structure of an ANN

2.4.2 Types of ANNs

Artificial Neural Networks can be broadly categorized into the following categories:

Feedforward Neural Network (FFNN)

It is one of the simplest neural networks. In the FFNN, the data flows in one direction from the input node(s) to the output node(s). Neurons are interconnected by weights that form some weighted associations between the inputs and outputs. The network learns by comparing the processed output vs the actual/target output. If this value is less than the desired threshold, the network then adjusts the values of the interconnecting weights according to a set learning rate based on the error values. *Figure 5*below shows an illustration of a single-layer FFNN.

International Research Journal of Innovations in Engineering and Technology(IRJIET)

ISSN (online): 2581-3048

Volume 7, Issue 10, pp 376-392, October-2023

https://doi.org/10.47001/IRJIET/2023.710051

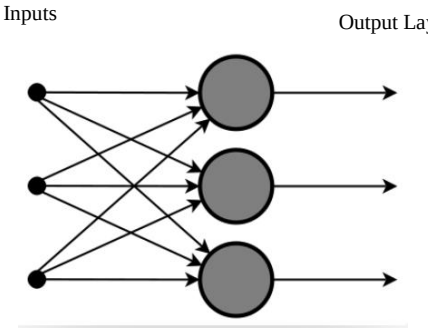

Figure 5: Illustration of a Single-Layer FFNN.

In the FFNN, the product of the inputs and the corresponding weights connecting them to a given node is calculated and then added together. This sum is then fed to the output, as shown in *Figure*5. If the FFNN has more than one layer, it is referred to as a Multilayer Perceptron (MLP).

The MLP has three or more layers and is used to classify data that cannot be linearly separated. Every single node in a given layer is connected to each node in the next layer; hence, the MLP is considered a fully connected ANN. An MLP uses a non-linear activation function, e.g., hyperbolic tangent or the logistic function. FFNNs mainly find applications in general regression and classification problems.*Figure 6* below shows the illustration of an MLP:

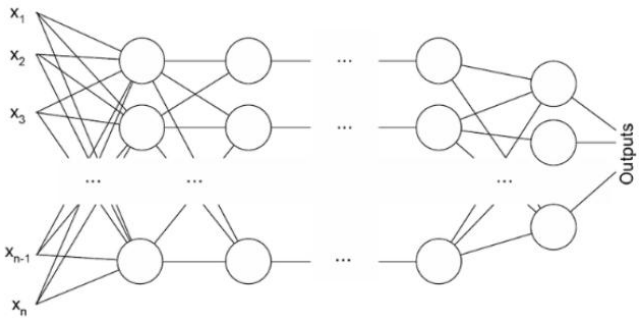

Figure 6: A Multilayer Perceptron

Radial Basis Function Neural Network (RBFNN)

RBFNNs are a class of FFNNs. The design of his network is such that it tries to establish the best curve of fit in a high-dimensional space. In getting the surface that provides the best line of fit for the training data, RBFNNs can use that to forecast future values of a given quantity. RBFNNs are applied in power restoration systems to ensure power restoration is done in the shortest time possible.

Convolutional Neural Network (CNN)

Convolutional networks are very effective in image and video recognition. They are mainly used for object detection and image classification problems. A CNN is a variant of the MLP in that it has several layers. The convolutional layers can either be wholly connected or pooled. *Figure 7* below shows a CNN.

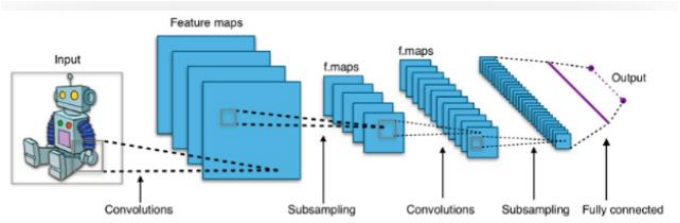

Figure 7: Illustration of a CNN

The Recurrent Neural Network (RNN)

The RNN is a type of neural network where the output of a given layer is saved and fed back to the network as an input. RNNs were created to solve the FFNN's problem of not handling sequential data well. The FFNN only considers the current inputs and cannot memorize previous inputs. Also, the input nodes of the FFNN are independent; hence, the neural network cannot learn any temporal relationship present in the input data [22].*Figure 8*below shows a simple illustration of an RNN.

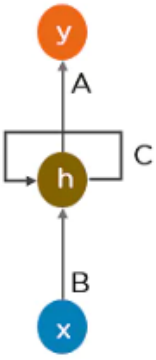

Figure 8: Illustration of an RNN (rolled)

Expanding/unrolling the illustration in *Figure 9*to show how the output of the hidden layer is fed back as an input in the next time step, we have:

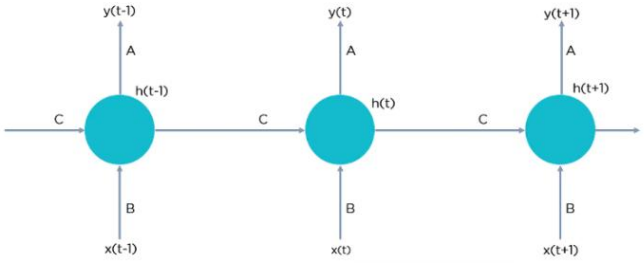

Figure 9: A Fully Connected RNN (unrolled)

International Research Journal of Innovations in Engineering and Technology(IRJIET)

ISSN (online): 2581-3048

Volume 7, Issue 10, pp 376-392, October-2023

https://doi.org/10.47001/IRJIET/2023.710051

Where:

$h(t)$is the new state of the network

$x(t)$is the input

$y(t)$the output

At the time (t), the state of the network is going to be determined by the input $x(t)$ as well as the previous/old state of the network $h(t-1)$, i.e.,

$$h(t) = f\left(x(t), h(t-1)\right) \qquad (3)$$

Therefore, an RNN forms a chain of interconnected modules of the simple neural network, as shown in*Figure 10*below:

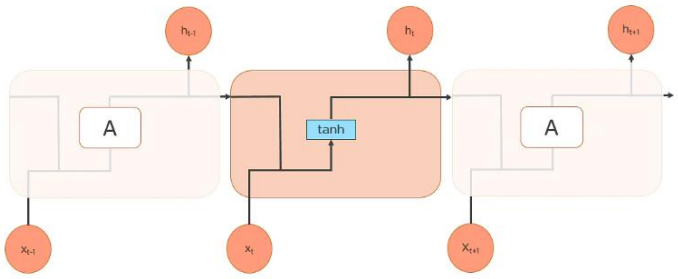

Figure 10: Chain of simple interconnected neural networks to form and RNN

The old state of the network is fetched back to improve the current output of the network. This makes RNNs well-adapted to deal with time series data, image captioning, and natural language processing. Such data usually correlates between previously observed data and the current data.

Standard RNNs have a few limitations:

i. They suffer from the vanishing gradient problem - RNNs work on time-dependent and sequential data problems. Gradients carry the information transmitted over time across the RNN to improve future updates. If the gradient becomes too small, the updates to the network become insignificant, limiting the network's ability to learn through long data sequences.

ii. They suffer from the exploding gradient problem - When training an RNN, if the slope of the data keeps growing exponentially and does not decay, this results in an exploding gradient. When large error gradients accumulate, huge updates are made to the networks for every step, resulting in longer training times and inferior performance.

To resolve the limitations of the RNNs, the Long Short-Term Memory (LSTM) neural network was developed. The LSTM has a long-term memory to learn any long-term dependencies in a dataset and store that information for future reference. For the LSTM, instead of having a chain of single neural networks, it has four layers that interact and communicate, as shown in*Figure 11*.

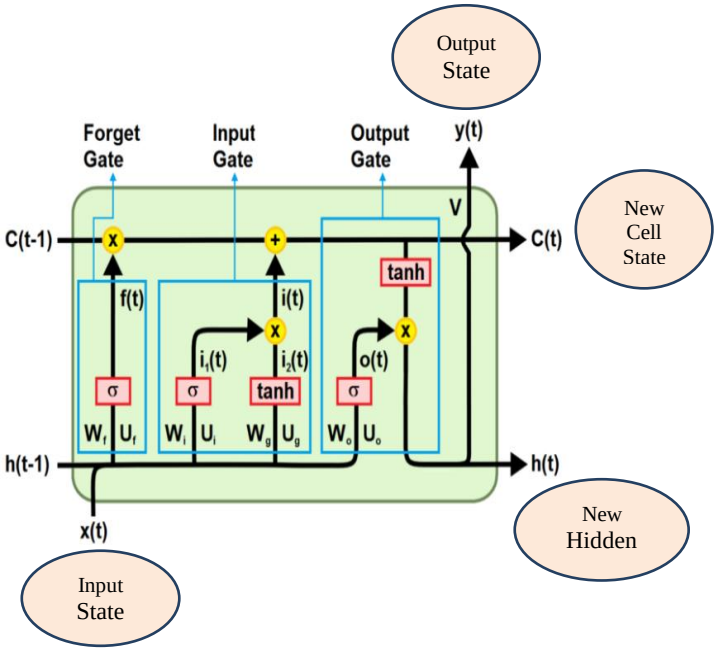

Figure 11: An LSTM Cell

The working of the LSTM is as explained below:

Stage 1: The Forget Gate

Here, the LSTM decides what to remember or forget from the information received from the previous timestep. The output of the forget gate is formulated as below:

$$f(t) = \sigma(U_f x(t) + W_f h(t-1)) \qquad (4)$$

The previous hidden state and the current input are passed through a sigmoid function to determine the relevance of the old information to the current input. The sigmoid function gives an output of a 0 or a 1. If the forget gate output $f(t) = 0$ or close to a zero, that information is irrelevant and forgotten.

If the forget gate output $f(t) = 1$ or close to one, then that information is remembered.

Stage 2: The Input Gate

The input gate updates the cell state with the current input. Initially, the previous hidden state and current input are given as inputs to a sigmoid function. The closer the output is to a 1, the more relevant the information is to the network. In this stage, to further improve the tuning of the network, the previous hidden state and current input are passed through a

International Research Journal of Innovations in Engineering and Technology(IRJIET)

ISSN (online): 2581-3048

Volume 7, Issue 10, pp 376-392, October-2023

https://doi.org/10.47001/IRJIET/2023.710051

tanh to squeeze the values between +1 and -1. This gives a weighting to the variables based on their relevance or level of importance. The two outputs are then multiplied element by element.

The sigmoid output then determines what information to keep from the tanh output.

$$i_1(t) = \sigma(U_i x(t) + W_i h(t-1)) \quad (5)$$

$$i_2(t) = \tanh(U_g x(t) + W_g h(t-1)) \quad (6)$$

$$i(t) = i_1(t) + i_2(t) \quad (7)$$

Stage 3: Calculation of the new cell state

After we get the output of our input gate, we calculate the new cell state as below:

$$c(t) = f(t)c(t-1) + i(t) \quad (8)$$

Stage 4: The Output Stage

First, the current cell state part that makes it to the output is determined. The cell state first passes through a tanh function to squash the values between $+1$and -1. The previous hidden state and the current input are summed and passed through a sigmoid function. After that, the two outputs are multiplied element by element, and this gives the matrix of the information to be contained in the next hidden state $h(t)$.

$$o(t) = \sigma(W_o h(t-1) + U_o x(t)) \quad (9)$$

$$h(t) = \tanh(c(t)) * o(t) \quad (10)$$

The Bidirectional LSTM is an improvement of the LSTM. It combines the LSTM network and the bidirectional RNNs allowing it to learn long-term dependencies in a dataset [23]. The BiLSTM allows better training of the network by traversing the training data twice, i.e. from left to the right (forward) and from right to left (backward) hence the term "bi".BiLSTM models have better accuracy than LSTM but reach equilibrium more slowly [24]. A structure of a BiLSTM network for three consecutive steps is as shown below [18]:

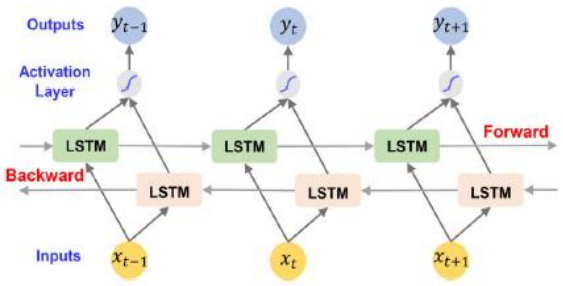

Figure 12: Structure of a BiLSTM Network

Data Decomposition Techniques.

Time series decomposition is a statistical task that involves breaking down data into several high and low-frequency components to extract information on seasonality or trend from the data [25]. Time series data is usually very noisy and complex, and decomposition helps break it down into simpler components. Decomposition gives a better understanding of the data by highlighting any repetitive seasons in the data or increasing or decreasing trends over time. Data decomposition reduces the non-stationary [22] and non-linear influence of a time-series dataset, making it easier to predict than the original data series.

Data decomposition models can be broken down into additive and multiplicative models. In the additive model, the variance of the data does not change much over time, and the trend line is linear [25]. To get the original time series, one only needs to add the lower frequency components the data is decomposed into. i.e.,

$$y(t) = s(t) + T(t) + R(t) \quad (11)$$

Where:

$y(t)$ - Original time series
$s(t)$- Seasonal component
$T(t)$ - Trend component
$R(t)$ - Residual component

In the multiplicative model, the seasonal and trend components also increase as the data increases over time. Hence, to reconstitute the original time series, one multiplies all the components the trend was broken down into as shown below [25]:

$$y(t) = s(t) \text{ x } T(t) \text{ x } R(t) \quad (12)$$

The working of data decomposition is illustrated in*Figure 13*:

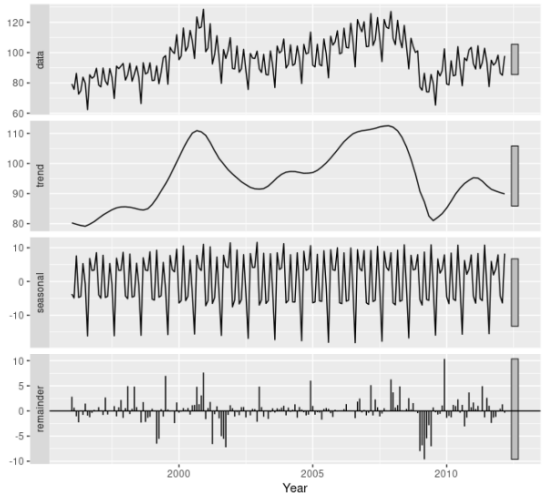

Figure 13: An Illustration of Data Decomposition.

ISSN (online): 2581-3048
Volume 7, Issue 10, pp 376-392, October-2023
https://doi.org/10.47001/IRJIET/2023.710051

There has been a lot of research on the significance of data decomposition of wind time series data on the accuracy of forecasts. Most researchers hybridize their algorithm of choice with a given data decomposition technique, and the results that have been posted show the improvements that are achieved with inclusion of data decomposition. The original time series/data is broken down into simpler constituent components which help improve the prediction of the data since the selected algorithm can now be optimized to forecast each of the components, after which the components are reconstituted to get the forecast series. The data decomposition techniques that have been previously used include the wavelet transform, empirical mode decomposition, variational mode decomposition, and their variants.

Previous Neural Network-Based Research on Wind Prediction.

In [20], the FFNN was used to predict wind power and gave MSE and MAE values of 13.5 and 9.35 for a 60-minute look ahead forecast. In [26], ANFIS was used to predict wind power and compared against the BPNN and GA-BPNN hybrid models. It was observed that ANFIS performed better than the BPNN but only slightly improved compared to the hybrid model. The BPNN model had 11 hidden layer neurons and was trained using the Levenberg-Marquardt algorithm with tangent sigmoid and pure linear activation functions at the input and output. This paper highlights the importance of hybridization since the GA_BPNN model had better performance than BP alone and its performance was almost equal to the ANFIS. In [27], ANFIS, ANN, and ARIMA were used to predict wind power. The ANN, with ten hidden layers and trained using Bayesian Regularization, was the best prediction algorithm for 1-hour ahead forecasts with RMSE and MAE values of 18.1 and 12.3. ANFIS came in at a close second with RMSE and MAE values of 18.4 and 12.2.

In [22], a neural network was used together with wavelet decomposition to forecast wind speed. The neural network had three layers with seven input neurons, 12 hidden layer neurons, and one output neuron for spring data. The network had nine-input neurons and 14 hidden layer neurons for the rest of the three seasons. The proposed network was enhanced using adaptive boosting and outperformed fuzzy logic and classical FFNNs. Test data was drawn from four seasons of the year; the least RMSE and MAE values posted were 1.1096 and 0.6855, respectively.

In [28], Wavelet Transform (WT) was combined with a two-hidden layer neural network to forecast wind speed. The data was decomposed into four components using wavelet transform. A neural network with two hidden layers was used to predict each of the obtained sub-series, after which the obtained forecasts were combined to get the final prediction. The proposed WT-TNN approach outperformed the NN, and the lowest RMSE and MAE values obtained for the four test datasets were 0.3317 and 0.2312. The TNN's corresponding RMSE and MAE values were 0.8019 and 0.4767, respectively. The superiority of the WT-TNN again showed the significance of including a data decomposition technique in a forecast.

In [29], the Discrete Wavelet Transform (DWT) was combined with ANNs to perform wind speed forecasting. A 5-level decomposition of the data was performed using DWT to give five detail coefficients and an approximation coefficient for each data value. Each detail and approximation coefficient had its neural network consisting of five inputs, two hidden layers, and one output. Five previous daily average wind speeds were used to forecast the wind speed for the sixth day, i.e., a sliding window of 5 days was used. After the prediction, all the outputs were reconstructed to give the final forecast. The dataset consisted of 4375 daily average wind speeds spanning twelve years from 2007 to 2018. The training data was from 2007 to 2017 (4015 data points), while the testing data used was for 2018 (360 points). The least RMSE and MAPE values posted for the test sets were 0.1265 and 0.0371.

In [30], empirical mode decomposition was used with RBFNN to forecast wind power with three input parameters - wind power, speed, and direction. EMD decomposed the data into eight intrinsic mode functions (IMFs) and one residual function. 900 samples were used to test the model, and 74 were used for testing. An RBFNN network was used for each of the obtained data sub-series, after which the values were combined to give the final prediction. The EMD_RBFNN model was compared to the RBFNN. EMD_RBFNN posted MAE and MSE values of 22.075 and 2.485, respectively, while RBFNN alone posted MAE and MSE values of 34.905 and 4.293, respectively. The results once more highlighted the advantage of data decomposition. Wind speed was forecast using EMD, ANNs, and ARIMA in [31], and the results were compared against ANN and EMD_ANN. EMD_ANN outperformed ANN alone, posting RMSE and MAE of 2.058 and 0.713 against RMSE and MAE of 2.949 and 1.456, respectively. The EMD_ANN_ARIMA was the best among the three, with RMSE and MAE of 0.890 and 0.527, respectively. In [32], a hybrid method combining EEMD and neural networks (Group Method of Data Handling networks) was used to forecast wind speed. The researcher compared the proposed method with GMDH neural networks alone, GMDH neural networks with wavelet packet decomposition, SVM, and ELM. The proposed method posted minimum RMSE and MAPE values of 0.0113 and 0.6337. These results were better than the algorithms used without any data decomposition techniques. EEMD performed better than WT in the same hybrid scenarios based on the obtained results.

International Research Journal of Innovations in Engineering and Technology(IRJIET)

ISSN (online): 2581-3048

Volume 7, Issue 10, pp 376-392, October-2023

https://doi.org/10.47001/IRJIET/2023.710051

In [33], a method based on variational mode decomposition (VMD) and the Bare Bones Fireworks Algorithm (BBFA) was proposed to forecast wind speed. Again, compared to the same approaches but with no data decomposition involved, the hybrid approaches using data decomposition came out as superior. VMD_BBFA had the least RMSE and MAE values of 0.20 and 0.17. In [34], LSTM and GA were combined to predict wind power for seven wind farms and showed an improvement of between 6% - 30% in wind power forecast accuracy. The LSTM network had three layers, a batch size of 30 and 10 epochs. The window size used is three timesteps. In [35], a hybrid of CNN-LSTM was used in wind power forecasting. The CNN was used to extract the temporal and spatial correlations between the parameters affecting wind power and the wind turbines' location on a wind farm. The CNN_LSTM improved the forecast accuracy by 14% compared to LSTM RMSE values.

III. METHODOLOGY

The developed hybrid approach utilizes the BiLSTM neural network and empirical mode decomposition to forecast wind power. The developed method follows the following stages, as illustrated in*Figure 14*and the detailed flowchart is shown in*Figure 15*.

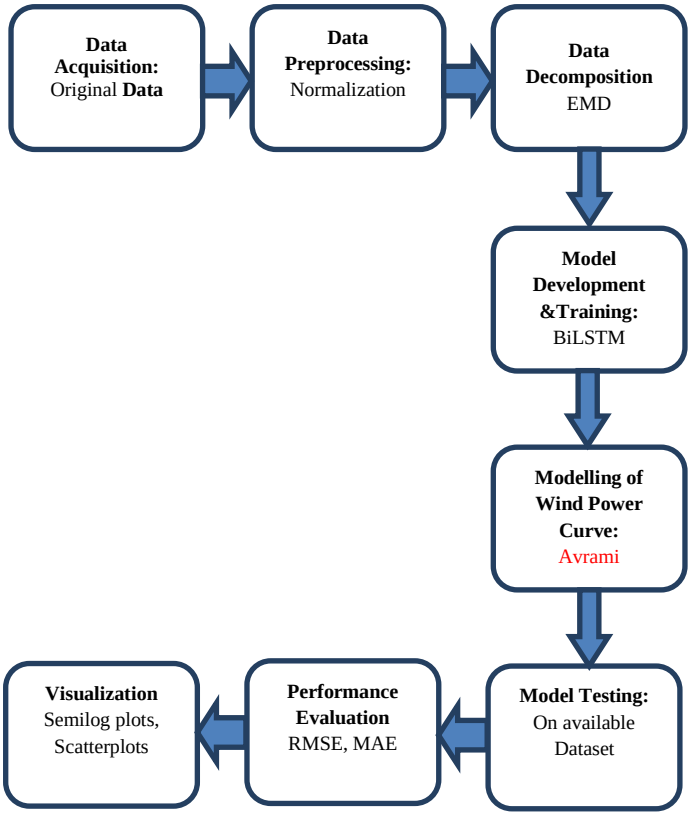

Figure 14: Illustration of the Various Stages of the developed BiLSTM+EMD+Power Curve Model.

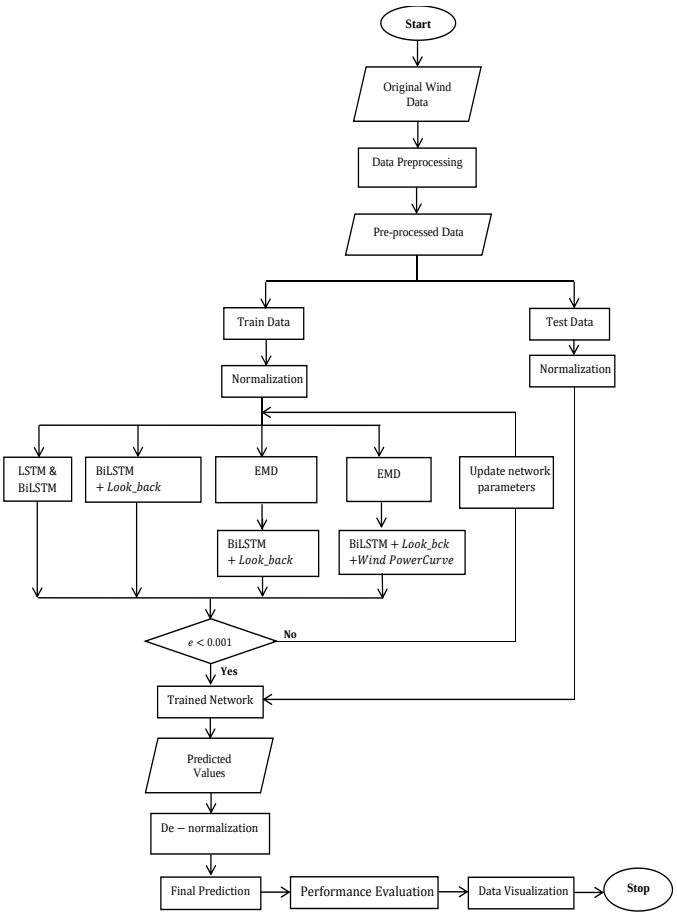

Figure 15: General Wind Power Forecasting Procedure

3.1 Dataset Description

The selected dataset is an online-based dataset used by various researchers looking into the wind power forecasting problem. The particulars of the dataset are summarized in*Table 1* below:

Table 1: Particulars of the Wind Dataset

Data Name	AL_WIND_07_12
Source	National Renewable Energy Laboratory (**NREL**)
Data Resolution	**Hourly** observations
Span	01/01/2007 00:00:00 to 31/12/2012 23:00:00 (**52560** observations equivalent to 6 years of data)

The AL_WIND_07_12 wind dataset consists of the following variables: a timestamp, air temperature (^{o}C), pressure (atm), wind speed (m/s), wind direction (deg) and wind power (kW).

International Research Journal of Innovations in Engineering and Technology(IRJIET)

ISSN (online): 2581-3048

Volume 7, Issue 10, pp 376-392, October-2023

https://doi.org/10.47001/IRJIET/2023.710051

3.2 Data Preparation

Data Preprocessing

Data preprocessing in time series data forecasting is one of the most important steps before any forecasting is done. It involves estimating missing values, identifying and filling existing outliers, and data feature scaling. The following are the steps that are used in the data preprocessing stage:

i. **Checking for missing data** - This is done by using the *rmmissing* function in MATLAB. The function has the option of removing rows or columns with missing data. One could specify the *Min Num Missing* threshold below which rows or columns with missing data are not deleted. The dataset used did not have any missing data.

ii. **Handing Outliers** - The *isoutlier* MATLAB function is used to identify the outliers in the wind dataset, after which any outliers are *clipped*. The AL_WIND_07_12 dataset did not have any outliers.

iii. **Feature scaling** - This is done to prevent the domination of one variable on the output. There are two types of standard feature scaling options: Standardization and normalization, and their formulas are represented in Equations 13 and 14 below:

$$x_{transformed} = \frac{x - mean\,(x)}{std.dev\,(x)} \quad (13)$$

$$x_{transformed} = \frac{x - min\,(x)}{\max(x) - \min(x)} \quad (14)$$

Where:

x is the data before scaling and $x_{transformed}$ is the data after scaling.

Standardization scales the data to the range of $-1 < x < 1$, while normalization squashes the data to the range $0 < x < 1$. For this work, **normalization** was used as the feature scaling method. This ensured the scaling of all the data values to between 0 and 1. Normalization ensures that no one factor or parameter is weighted higher than others based on their numerical values. For example, in the absence of feature scaling, the wind direction would be weighted higher than wind speed since wind direction values (angles) typically have larger values compared to wind speed values.

Below is a screenshot of part of the normalized data in the AL_WIND_07_12 dataset for illustration.

DateTime	AirTemperature	Pressure	WindSpeed	WindDirection	Power
1/1/2007 0:00	0.4727	0.4364	0.4531	0.6361	0.5501
1/1/2007 1:00	0.4537	0.4463	0.4742	0.6459	0.6084
1/1/2007 2:00	0.4281	0.4542	0.4371	0.6556	0.4980
1/1/2007 3:00	0.4150	0.4567	0.4259	0.6861	0.4640
1/1/2007 4:00	0.4033	0.4527	0.4209	0.7111	0.4469
1/1/2007 5:00	0.3903	0.4531	0.4144	0.7250	0.4214

Figure 16: A Section of Normalized AL_WIND_07_12 Wind Power Dataset

3.4 Modelling of the Wind Power Curve from Historical Data using the Avrami Equation

A wind power curve is a graphical illustration of the relationship between wind speed and wind power. The actual wind power curve for a wind turbine or a wind farm is usually not an exact match to the wind power curve from the manufacturer. The manufacturer wind power curve is derived from turbines operated in ideal conditions and with wind turbine blade angles positioned to extract as much energy as possible from the wind. For this research, the wind power curve was reconstructed from the historical training data (i.e., the data section used for training). First, the wind speed vs wind power scatter plot was drawn in excel. The wind turbine power curve presents itself in a graph mimicking the Avrami equation. The Avrami equation is an equation that defines the kinetics behind crystallization and is also applied to define other applications involving changes such as chemical reactions [36]. The best line of fit through the data was determined using Excel's solver, and then the Avrami equation coefficients were incorporated into the MATLAB prediction algorithm.

For each test, the dataset was split into 70% training and 30% testing data. Based on this split, the AL_WIND_07_12 wind dataset uses approximately four years of data for training and data from the final two years is used for testing. The partition of the data into train and test sets is illustrated in *Table 2* below:

Table 2: Partition of the Dataset into Train and Test Data

AL_WIND_07_12 Dataset	Train Data	$0.7 * 52560 = \mathbf{36792}$
	Test Data	$0.3 * 52560 = \mathbf{15768}$

The general form of the Avrami equation is defined as:

$$y(t) = 1 - e^{-kt^n} \quad (15)$$

Where $y(t)$ is the fraction of completed transformation at time t, k is a rate constant, t is time and n is the growth dimensionality.

International Research Journal of Innovations in Engineering and Technology(IRJIET)

ISSN (online): 2581-3048

Volume 7, Issue 10, pp 376-392, October-2023

https://doi.org/10.47001/IRJIET/2023.710051

The equation, as defined in Equation 15 above, is constrained between 0 and 1. Since our wind data exceeds these boundaries, we modify the Avrami equation to encompass the scope of wind power by introducing a fitting constant, $\boldsymbol{A}$. The modified Avrami equation to fit the wind power curve modelling context now becomes:

$$y(t) = \boldsymbol{A}(1 - e^{-kt^n}) \qquad (16)$$

The fitting constants are $\boldsymbol{A}, k, n$ where t *represents wind speed and* y *represents* P_{out}

The fitting constants are determined by minimizing the sum square residual where:

$$residual = \ actual\ power - avrami\ power\ estimate \quad (17)$$

The fitting of the avrami equation is done in excel using the ***Solver***tool**,** and the obtained Avrami equation is coded into the forecasting algorithm in MATLAB. For the AL_WIND_07_12 wind dataset, the Avrami equation for minimized Sum of Squared Residuals (SSR) was obtained as follows:

$$\boldsymbol{y = 59711.44\ \ (1 - e^{-k(8.06\,x10^{-05})\cdot\ t^{4.203855}})} \quad (18)$$

A scatter plot of the train data before curve fitting is shown in*Figure 17*.

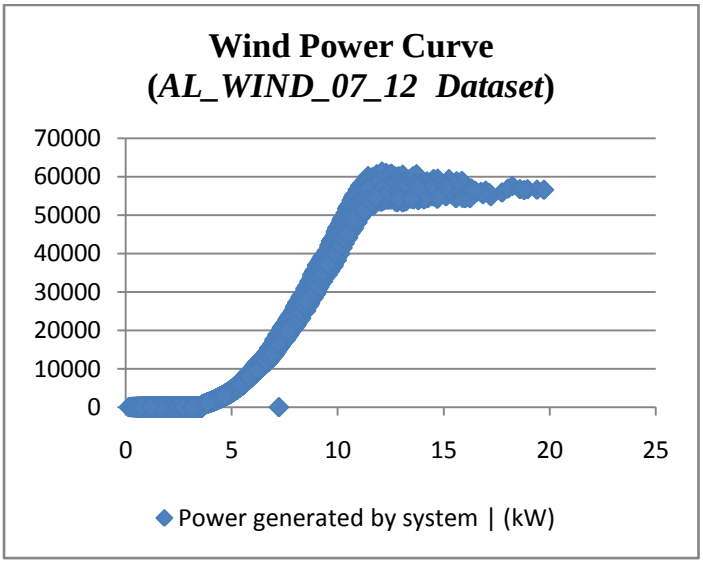

Figure 17: Scatter Plot of the Wind Power vs Wind Speed Data before curve fitting (AL_WIND_07_12 Dataset)

After the optimal Avrami equation parameters are defined, a curve of best fit is plotted as shown below:

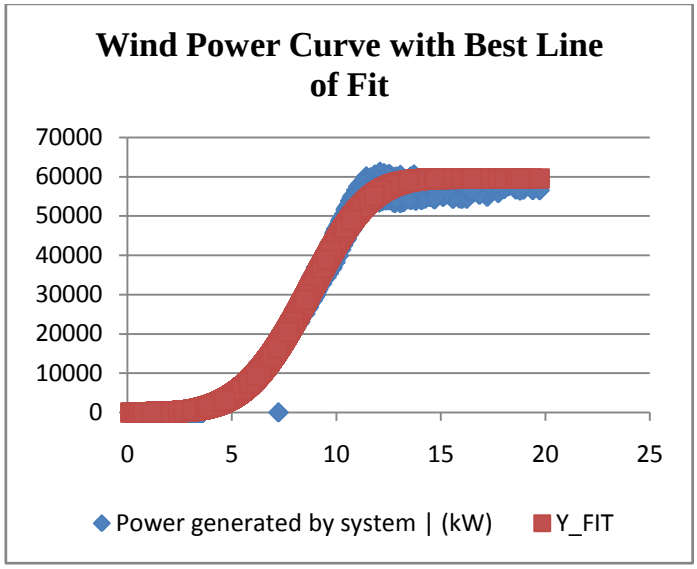

Figure 18: Scatter Plot of the Wind Power vs Wind Speed Data after curve fitting with Avrami (AL_WIND_07_12 Dataset)

The wind power curve, as determined by the Avrami equation, is used to provide a wind power estimate for wind power at time t. This new input provides a variable that improves the BiLSTM prediction accuracy for wind power at time t and with a look-back period of n time steps.

Models and Model Parameters

Based on repeated experiments, the hyper parameters for the LSTM and BiLSTM were set as follows:

Number of neurons in the hidden layer = 150
Iteration epochs = 10
Look back period/Sequence length = 6

The **Adam** optimizer was used since it helps accelerate the learning speed of LSTM while reducing the optimization resources required. It combines the benefits of two gradient descent algorithms: AdaGrad and RMSProp [22].
The below models were tested on the wind power dataset.

i. LSTM
ii. LSTM_EMD
iii. BiLSTM
iv. BiLSTM_EMD
v. BiLSTM_EMD_with Avrami Power Curve

IV. RESULTS AND DISCUSSION

Wind Power Forecasting using the Indirect Approach

In this paper, Wind Power forecasting is done using BiLSTM_EMDafter which the model is enhanced using the Avrami wind power curvemodelled from historical data.The BiLTSM + EMD+ Avrami Curve model was then compared

International Research Journal of Innovations in Engineering and Technology(IRJIET)

ISSN (online): 2581-3048

Volume 7, Issue 10, pp 376-392, October-2023

https://doi.org/10.47001/IRJIET/2023.710051

with the BiLSTM and BILSTM_EMD models for different forecast horizons. The evaluation metrics used were RMSE and MAE.

Test on Effectiveness of Data Decomposition and Illustration of the Deficiency of LSTMs

Simulations in this section were carried out on a section of the dataset. The forecasting algorithm used was the LSTM, and it was later hybridized using EMD and a wind power curve. The LSTM was used alone to predict 168 look-ahead instances (equivalent to 1 week for the AL_WIND_07_12 dataset with hourly observations). *Figure 4.1* below shows the plot of the predicted vs actual wind power forecast using LSTM alone and with wind power as the input parameter.

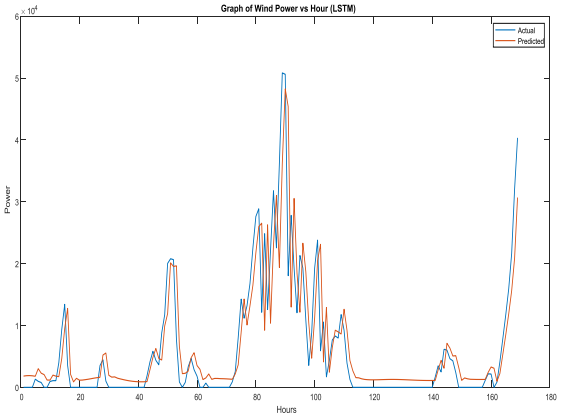

Figure 19: Graph of Wind Power vs Time for LSTM alone (168 look ahead instances)

Observation: The predicted Wind Power shows good accuracy and can. However, a time lag is noted in the LSTM prediction compared to the actual wind power data. This is caused by the high variability of the wind power data and can be resolved by breaking down the wind power time-series data into simpler forms using data decomposition techniques. *Figure 4.2* below shows the improvement in the forecast when data decomposition is used to break down the data before the prediction.

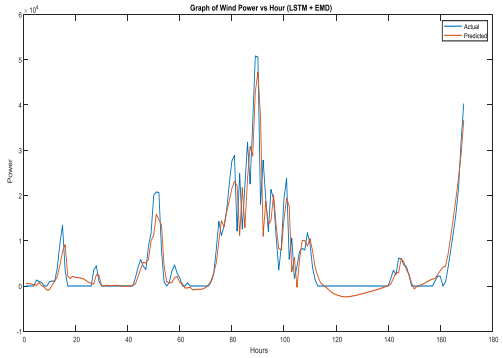

Figure 20: Graph of Wind Power vs Time for LSTM+EMD (168 look ahead instances)

Observation: The time lag phenomenon visible in the LSTM prediction graphs in *Fig 4.1* is now suppressed by EMD, and the prediction is now better. A deviation in the prediction below the zero mark is noted in some instances, e.g. at hour 70 and hour 130, and this results from the fact that in the decomposed data, some of the components are negative. To correct this, a simple curve limit definition was used to limit the prediction from swinging below the zero mark, and the final updated prediction is as shown in *Fig 4.3* below:

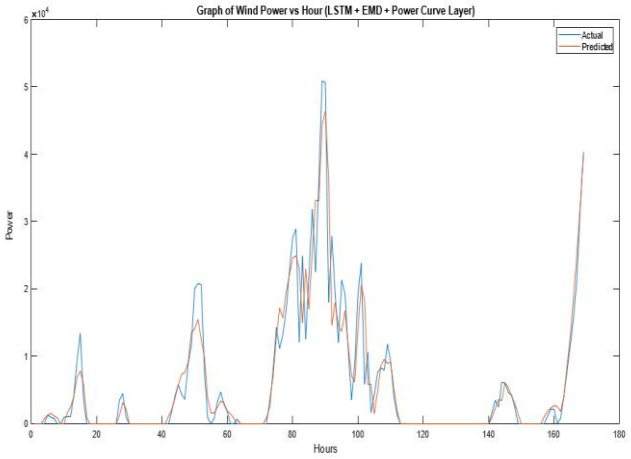

Figure 21: Graph of Wind Power vs Time for LSTM+EMD with Power Curve Layer (168 look ahead instances).

The three approaches were then compared, and a graph of a prediction of 100 future instances was plotted, as shown in *Fig 4.4* below.

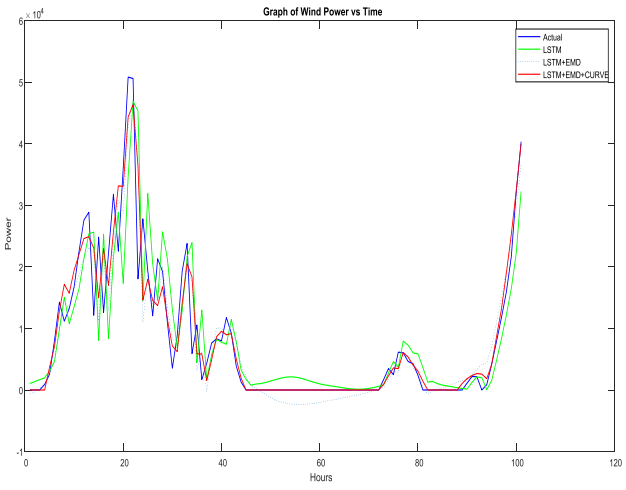

Figure 22: Graph of Wind Power vs Time for Models Comparison. (100 observations sampled for plotting from the 876 predictions made.)

Observation: The LSTM +EMD with Power Curve model gives the best prediction compared to the other two models since it follows the actual wind power better than the other two models. LSTM +EMD + Power curve gives a 22.858%

International Research Journal of Innovations in Engineering and Technology(IRJIET)

ISSN (online): 2581-3048

Volume 7, Issue 10, pp 376-392, October-2023

https://doi.org/10.47001/IRJIET/2023.710051

improved forecast accuracy compared to LSTM alone and a 13.895% better forecast compared to LSTM +EMD. *Table 4.1* below summarizes the RMSE and MAE values for the forecast.

Table 4.1: RMSE Values Comparison for the Models.

S/No	Model	RMSE (kW)
1.	LSTM	6184
2.	LSTM+EMD	5540
3.	LSTM+EMD+Curve Correction	4770

The importance of this section was to show the limitation of LSTMs and how that limitation was overcome. In the next section, the BiLSTM was used as the forecasting network, EMD as the data decomposition technique and the Avrami approach as the method of obtaining the wind power curve from the historical training wind data. The BiLSTM model offers better prediction accuracy than LSTM since it traverses the data backwards and forward during training [24], giving it a better understanding of the relationship between various variables to determine more precise patterns and connections in the dataset. Therefore, this research now focuses on the BiLSTM networks from this point onwards.

BiLSTM Alone

Dataset: AL_Wind_07_12 Dataset
Predicted Variable: Wind Power (t)
Input Variables: Air Temperature, Pressure, Wind Speed & Wind Direction
Hidden layer neurons = 150 (Determined through trial and error)
Look_back/Sequence length = 6
Average simulation time: = 390 seconds
The results obtained from this forecast are as shown below:

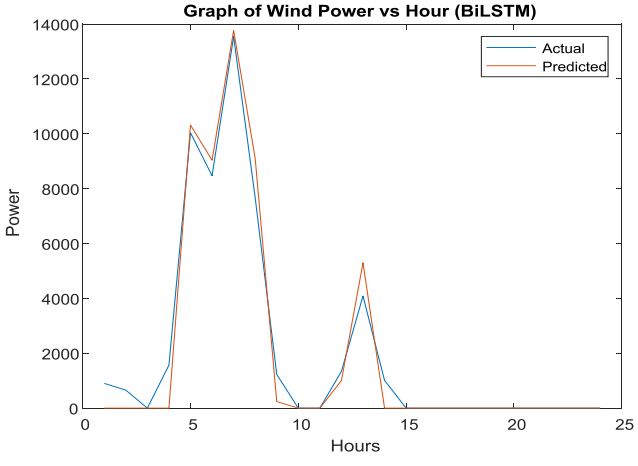

Figure 23: 24 - Hours Ahead Wind Power Forecast

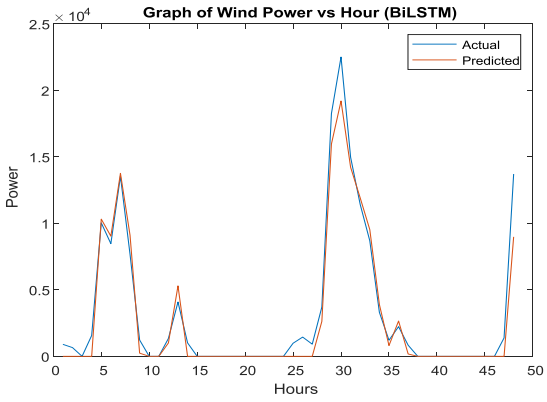

Figure 24: 48 Hours Ahead Wind Power Forecast

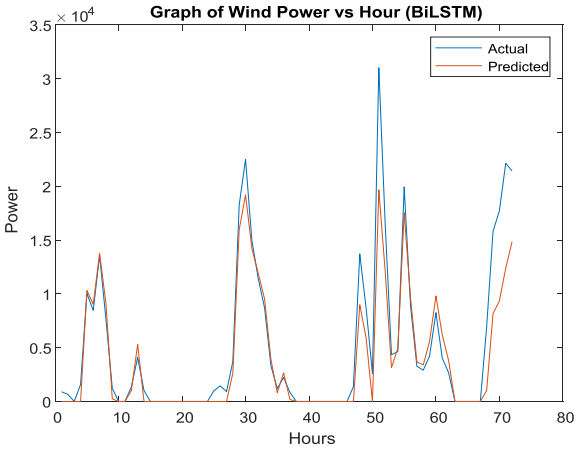

Figure 25: 72 Hours Ahead Wind Power Forecast

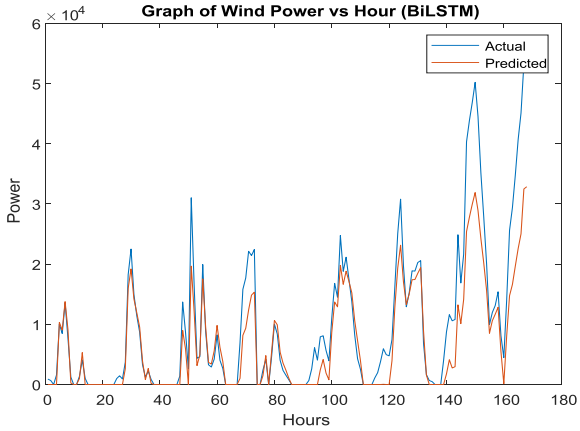

Figure 26: 168 Hours Ahead Wind Power Forecast

Table 3: Prediction Results of BiLSTM Alone

Look Ahead (hrs)	RMSE (kW)	MAE (kW)	R^2
24	639.0	382.6	0.9898
48	1098.5	613.9	0.9827
72	1514.8	1398.4	0.9478
168	5580.7	3174.4	0.9568

International Research Journal of Innovations in Engineering and Technology(IRJIET)

ISSN (online): 2581-3048

Volume 7, Issue 10, pp 376-392, October-2023

https://doi.org/10.47001/IRJIET/2023.710051

Observation: The BiLSTM network has its best prediction accuracy during the 24-hour look ahead test. The forecast accuracy deteriorates as the forecast period increases.

BiLSTM_EMD

Dataset: AL_Wind_07_12 Dataset
Predicted Variable: Wind Power (t)
Input Variables: Wind Power (t-1, t-2, ….)
Hidden layer neurons = 150 (Determined through trial and error)
Average simulation time: = 10 minutes
The results obtained from this forecast are as shown below:

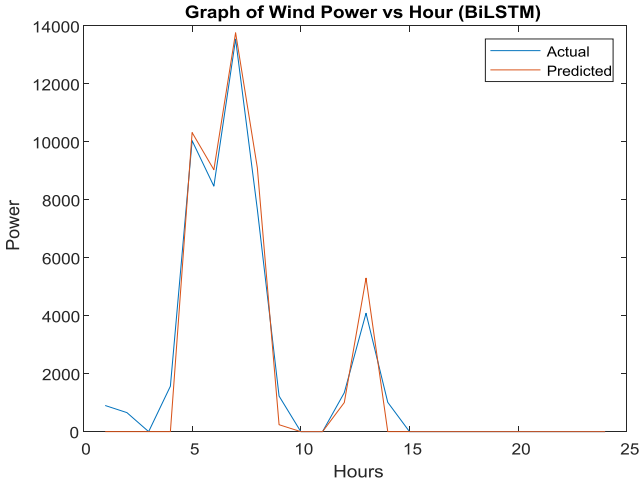

Figure 27: 24 - Hours Ahead Wind Power Forecast

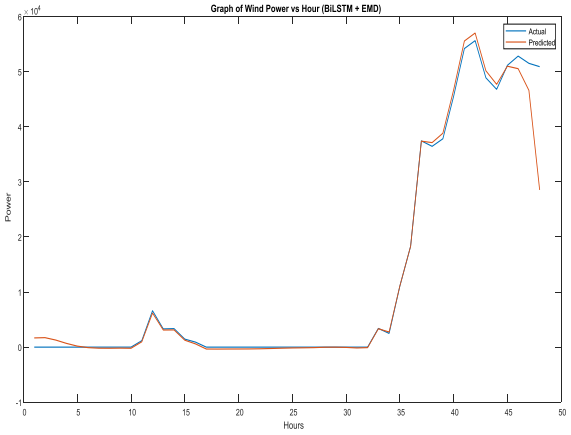

Figure 28: 48 Hours Ahead Wind Power Forecast

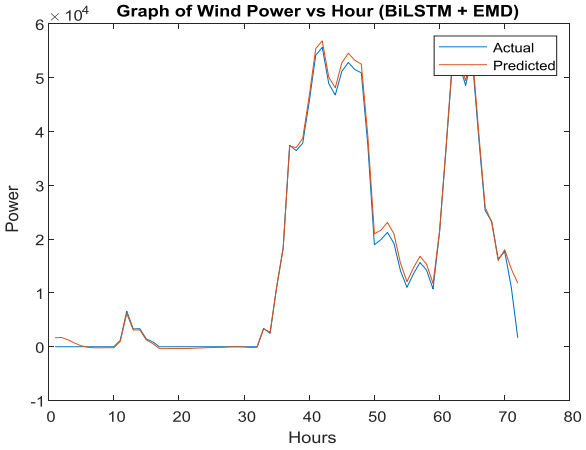

Figure 29: 72 Hours Ahead Wind Power Forecast

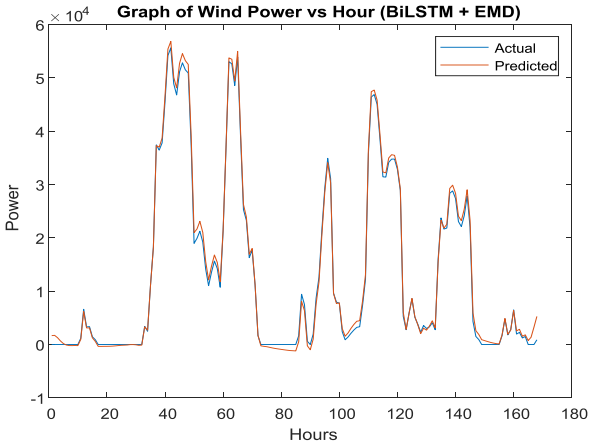

Figure 30: 168 Hours Ahead Wind Power Forecast

Table 4.2: Prediction Results of BiLSTM_EMD

Look Ahead (hrs)	RMSE (kW)	MAE (kW)	R^2
24	164.1	101.6	0.9989
48	908.1	686	0.9977
72	1242.2	789	0.9866
168	3381	927	0.9868

Observation: The BiLSTM_EMD model has its best prediction accuracy during the 24-hour look ahead test. The forecast accuracy deteriorates as the forecast period increases. EMD is seen to improve the accuracy of the BiLSTM model working alone.

BiLSTM_EMDwith Avrami Curve

Dataset: AL_Wind_07_12 Dataset
Predicted Variable: Wind Power (t)
Input Variable: Wind Power (t-1, t-2, ….)
Hidden layers = 150 (Determined through trial and error)
Look Back/ Sequence Length = 6
The results obtained from this forecast are as shown below:

International Research Journal of Innovations in Engineering and Technology(IRJIET)

ISSN (online): 2581-3048

Volume 7, Issue 10, pp 376-392, October-2023

https://doi.org/10.47001/IRJIET/2023.710051

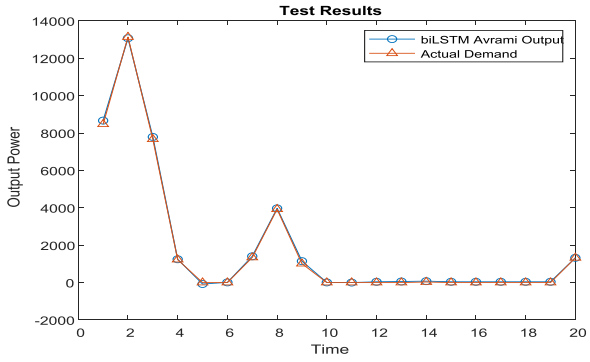

Figure 31: 24 - Hours Ahead Wind Power Forecast

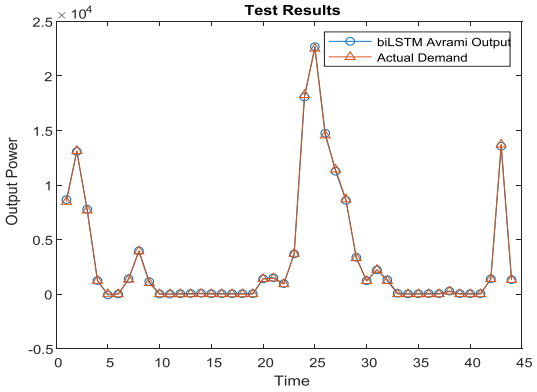

Figure 32: 48 Hours Ahead Wind Power Forecast

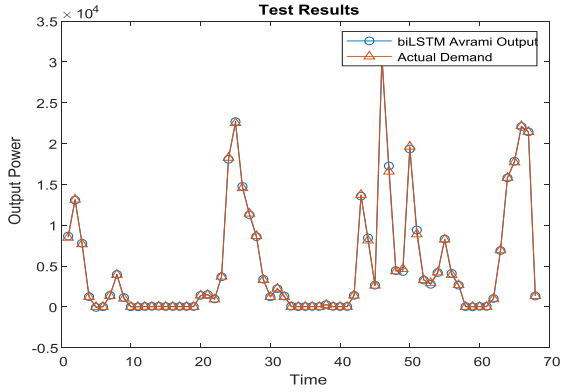

Figure 33: 72 Hours Ahead Wind Power Forecast

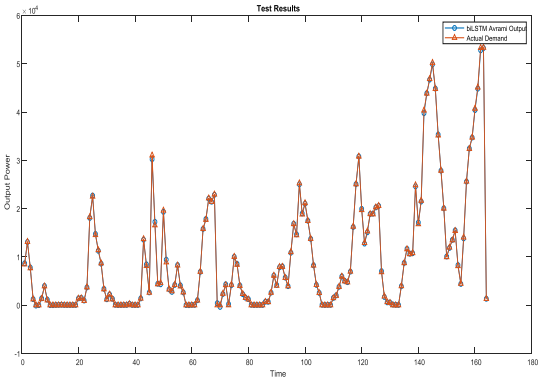

Figure 34: 168 Hours Ahead Wind Power Forecast

Table 4.3: Prediction Results of BiLSTM with Avrami Curve

Look Ahead (hrs)	RMSE (kW)	MAE (kW)	R^2
24	**72.1**	**55.2**	**0.9999**
48	84.1	64.6	0.9999
72	174.9	99.5	0.9997
168	161.7	102.6	0.9999

Observation: The BiLSTM_EMD+Avrami model has its best prediction accuracy during the 24-hour look ahead test. The forecast accuracy deteriorates as the forecast period increases. The inclusion of the concept of the Avrami Equation is seen to improve the accuracy of the BiLSTM model in predicting future wind power.

Summary

The RMSE value of the BiLSTM_EMD+Avrami Power curve is 72.1kW making it the lowest among the hybrid approaches. The improvements that data decomposition brings to wind power forecasting are seen when the predictions from LSTM vs LSTM+EMD and BiLSTM vs BiLSTM+EMD approaches, respectively. The latter two models with EMD have lower RMSE values than the base models. Neural networks perform better when the time series data is broken down/decomposed into its constituent frequency components. The data becomes easier to understand and to extract meaningful information from a time series. Therefore, the developed approach to wind power forecasting based on BiLSTM_EMD +Avrami Power Curve is an effective approach to wind power prediction.

V. CONCLUSION

This work successfully developed an approach to wind power forecasting based on BiLSTM and EMD. Data preprocessing was done to ensure no missing data values, after which feature scaling was done on the data using normalization. The preprocessed data was then used in various models based on BiLSTM alone, BiLSTM and with EMD and BiLSTM_EMD with the Avrami Power Curve. The output of all these models was the predicted wind power. In the adopted forecasting model, pas wind power forecasts $(t-1, t-2, ..)$ were used to predict wind power at time t. The Avrami power curve was used to approximate wind power at time t given wind speed to ensure that the BiLSTM neural network could converge faster and offer more accurate results. The wind power curve was modelled from the historical train data in the dataset to ensure as much factual information was captured as compared to using the manufacturer's wind power curve. The

ISSN (online): 2581-3048
Volume 7, Issue 10, pp 376-392, October-2023
https://doi.org/10.47001/IRJIET/2023.710051

modelled wind power curve considers dynamics of the wind farm operation that could have affected the wind power output, such as the prevailing weather conditions at that time and other operational constraints.

The best RMSE value obtained was when the prediction was made using the BiLSTM+Avrami model, which was an RMSE value of **72.1kW** for a 24-hour ahead forecast. This was a more accurate forecast than the BiLSTM+EMD model with a 164.1kW RMSE for the 24-hour forecast and the BiLSTM alonewith a 639.0 kW RMSE for the 24-hour forecast. The use of EMD does, however, come at a small cost. When the data is decomposed into its various mode components, each input mode component data and its corresponding output data is forecast by its own BiLSTM network, increasing the computation times. However, a longer execution time is worth it since the forecast model's accuracy is improved.

With improved forecasts, integrating large-scale wind power in present-day grids becomes easier and more economical since improved accuracy of wind forecasts means less balancing power requirements.

REFERENCES

[1] J. Jarke and G. Perino, "Do Renewable Energy Policies Reduce Carbon Emissions? On Caps and Inter-Industry Leakage," J. Environ. Econ. Manage., 2017, doi: 10.1016/j.jeem.2017.01.004.

[2] K. N. KalairAnam, Abas Naeem, ShoaibSaleem Muhammad, Raza Kalair Ali, "Role of Energy Storage Systems in Energy Transition from Fossil Fuels to Renewables," Energy Storage, vol. 3, no. 1, 2020, doi: 10.1002/est2.135.

[3] Oxford Net Zero, "WHAT IS NET ZERO?" https://netzeroclimate.org/what-is-net-zero/.

[4] Y. Lu, Z. Khan, M. Alvarado, Y. Zhang, and Z. Huang, "A Critical Review of Sustainable Energy Policies for the Promotion of Renewable Energy Sources," MDPI J. Sustain., vol. 12, no. 12, 2020.

[5] U.S Energy Information Administration, "Electricity explained Electricity in the United States," 2021. https://www.eia.gov/energyexplained/electricity/electricity-in-the-us.php.

[6] EPA, "The Sources and Solutions: Fossil Fuels," 2019. https://www.epa.gov/nutrientpollution/sources-and-solutions-fossil-fuels.

[7] J. Lee and F. Zhao, "Global Wind Report 2021," 2021.

[8] Global Wind Energy Council, "GWEC | GLOBAL WIND REPORT 2022," 2022.

[9] S. D. Ahmed and F. S. M. Al-ismail, "Grid Integration Challenges of Wind Energy : A Review," vol. 8, no. type 1, pp. 10857–10878, 2020.

[10] H. Lotfi, T. Lotfi, and S. Hesari, "Short and Mid-Term Wind Power Forecasting with ANN-PSO," no. September, 2014.

[11] D. C. Aliprantis and W. Lafayette, "Fundamentals of Wind Energy Conversion for Electrical Engineers," 2014.

[12] A. M. Ragheb and Ragheb Magdi, "Wind Turbines Theory - The Betz Equation and Optimal Rotor Tip Speed Ration," in *Fundamental and Advanced Topics in Wind Power*, 2011, pp. 19–29.

[13] S. M. Rafaat and R. Hussein, "Power Maximization and Control of Variable-Speed Wind Turbine System Using Extremum Seeking," pp. 51–69, 2018, doi: 10.4236/jpee.2018.61005.

[14] Vestas Wind Systems, "V52-850 kW The turbine that goes anywhere."

[15] "A Novel and Alternative Approach for Direct and Indirect Wind-Power Prediction Methods _ Enhanced Reader.pdf." .

[16] S. Hanifi, X. Liu, Z. Lin, and S. Lotfian, "A Critical Review of Wind Power Forecasting Methods—Past, Present and Future," *Energies*, vol. 13, no. 15, p. 3764, Jul. 2020, doi: 10.3390/en13153764.

[17] X. Zhao, S. Wang, and T. Li, "Review of Evaluation Criteria and Main Methods of Wind Power Forecasting," *Energy Procedia*, vol. 12, pp. 761–769, 2011, doi: 10.1016/j.egypro.2011.10.102.

[18] Y. Wang, R. Zou, F. Liu, L. Zhang, and Q. Liu, "A review of wind speed and wind power forecasting with deep neural networks," *Appl. Energy*, vol. 304, no. April, 2021, doi: 10.1016/j.apenergy.2021.117766.

[19] S. M. Lawan, W. A. W. Z. Abidin, W. Y. Chai, A. Baharun, and T. Masri, "Different Models of Wind Speed Prediction; A Comprehensive Review," vol. 5, no. 1, 2014.

[20] V. Singh, "Application of Artificial Neural Networks for Predicting Generated Wind Power," vol. 7, no. 3, pp. 250–253, 2016.

[21] A. Sharkawy, "Principle of Neural Network and Its Main Types : Review Principle of Neural Network and Its Main Types : Review," no. August, pp. 7–19, 2020, doi: 10.15377/2409-5761.2020.07.2.

[22] Y. Liu, L. Guan, C. Hou, H. Han, Z. Liu, and Y. Sun, "Wind Power Short-Term Prediction Based on LSTM and Discrete Wavelet Transform," MDPI J. Appl. Sci., vol. 9, 2019, doi: 10.3390/app9061108.

[23] A. Dolatabadi, H. Abdeltawab, and Y. Abdel-Rady, "Hybrid Deep Learning-Based Model for Wind

ISSN (online): 2581-3048
Volume 7, Issue 10, pp 376-392, October-2023
https://doi.org/10.47001/IRJIET/2023.710051

Speed Forecasting Based on DWPT and Bidirectional LSTM Network," pp. 229219–229232, 2020, doi: 10.1109/ACCESS.2020.3047077.

[24] S. S. Namini, N. Tavakoli, and Ak. Namin, "The Performance of LSTM and BiLSTM in Forecasting Time Series," doi: 10.1109/BigData47090.2019.9005997.

[25] A. Chourasia, "Decomposition in Time Series Data," 2020. https://medium.com/analytics-vidhya/decomposition-in-time-series-data-b20764946d63.

[26] Y. Kassa, J. H. Zhang, D. H. Zheng, and Dan Wei, "Short term wind power prediction using ANFIS," in 2016 IEEE International Conference on Power and Renewable Energy (ICPRE), Oct. 2016, pp. 388–393, doi: 10.1109/ICPRE.2016.7871238.

[27] Q. Chen, C. Town, C. Town, and C. Town, "Wind Power Forecasting Wind Power Forecasting," IFAC-PapersOnLine, vol. 51, no. 28, pp. 414–419, 2018, doi: 10.1016/j.ifacol.2018.11.738.

[28] J. Wang, "A Hybrid Wavelet Transform Based Short-Term Wind Speed," vol. 2014, pp. 12–17, 2014.

[29] F. Berrezzek, K. Khelil, and T. Bouadjila, "Efficient Wind Speed Forecasting Using Discrete Wavelet Transform and Artificial Neural Networks," vol. 33, no. 6, pp. 447–452, 2019.

[30] Z. Zheng, Y. Chen, X. Zhou, and M. Huo, "Short-Term Wind Power Forecasting Using Empirical Mode Decomposition and RBFNN," 2012.

[31] M. K. Bantupalli, "decomposition with ANN and ARIMA models," 2017.

[32] H. Liu, Z. Duan, H. Wu, Y. Li, and S. Dong, "Wind speed forecasting models based on data decomposition , feature selection and group method of data handling network," Measurement, vol. 148, p. 106971, 2019, doi:10.1016/j.measurement.2019.106971.

[33] J. Quan and L. Shang, "An Ensemble Model of Wind Speed Forecasting Based on Variational Mode Decomposition and Bare-Bones," vol. 2021, 2021.

[34] F. Shahid, A. Zameer, and M. Muneeb, "A novel genetic LSTM model for wind power forecast," Energy, vol. 223, p. 120069, 2021, doi: 10.1016/j.energy.2021.120069.

[35] Y. Wu, Qianyu Guan1, FeiLv, Chen Huang, "IET Renewable Power Gen - 2021 - Wu - Ultra- short- term multi- step wind power forecasting based on CNN- LSTM.pdf."

[36] R. G. Finke and S. Ozkar, "Silver Nanoparticles Synthesized by Microwave Heating: A Kinetic and Mechanistic Re-Analysis and Re-Interpretation," J. Phys. Chem., no. 111, p. 121, 2017, doi: 10.1021/acs.jpcc.7b06323.

Citation of this Article:

Joseph N. Mathenge, John N. Nderu, David K. Murage, "A Short-Term Hybrid Wind Power Forecasting Approach using BiLSTM_EMD and the Avrami Curve" Published in *International Research Journal of Innovations in Engineering and Technology - IRJIET*, Volume 7, Issue 10, pp 376-392, October 2023. Article DOI https://doi.org/10.47001/IRJIET/2023.710051

****

