Volume 13, Issue 4, Page 107-119, 2024 https://doi.org/10.46810/tdfd.1525648 Research Article

BİNGÖL UNIVERSITY

Türk Doğa ve Fen Dergisi
Turkish Journal of Nature and Science
www.dergipark.gov.tr/tdfd

Comparative Analysis of LSTM Architectures for Wind Speed Prediction: A Case Study in Muş, Türkiye

İhsan TUĞAL[1*]

[1] Muş Alparslan University, Engineering and Architecture Faculty, Software Department, Muş, Türkiye
İhsan TUĞAL ORCID No: 0000-0003-1898-9438

**Corresponding author: i.tugal@alparslan.edu.tr*

(Received: 31.07.2024, Accepted: 27.11.2024, Online Publication: 30.12.2024)

Keywords
Energy,
LSTM,
Wind Speed,
Prediction,
Time Series

Abstract: Accurate wind speed prediction is critical for energy planning and sustainable development, particularly in regions like Muş, Turkey, where renewable energy potential remains underexplored. While many studies focus on wind speed forecasting using conventional methods, there is a research gap in evaluating the comparative effectiveness of advanced Long Short-Term Memory (LSTM) architectures for this purpose. This study aims to assess the predictive performance of five LSTM models (Vanilla LSTM, Stacked LSTM, Bidirectional LSTM, Attention LSTM, and Residual LSTM) on daily wind speed data from Muş. The dataset, obtained from the Muş Meteorological Office, consists of 20,088 daily wind speed measurements from 1969 to 2023. The results demonstrated that the Vanilla LSTM achieved the lowest MSE and MAE, indicating its superior overall accuracy, while the Attention LSTM achieved the lowest MAPE, showcasing better percentage-based accuracy. These findings suggest that Vanilla LSTM and Attention LSTM are the most effective models for wind speed forecasting in Muş. The choice between these models depends on prioritizing either absolute error minimization or percentage error accuracy, providing a strategic framework for model selection in similar renewable energy forecasting applications.

Rüzgar Hızı Tahmini için LSTM Mimarilerinin Karşılaştırmalı Analizi: Türkiye, Muş'ta Bir Vaka Çalışması

Anahtar Kelimeler
Enerji,
LSTM,
Rüzgar hızı,
Tahmin,
Zaman serileri

Öz: Rüzgar hızı tahmini, enerji planlaması ve sürdürülebilir kalkınma için kritik bir öneme sahiptir. Yenilenebilir enerji potansiyelinin yeterince araştırılmadığı Türkiye'nin Muş gibi bölgelerinde özellikle daha önemlidir. Pek çok çalışma rüzgar hızı tahmini için geleneksel yöntemlere odaklanmış olsa da, gelişmiş Uzun Kısa Süreli Bellek (LSTM) mimarilerinin karşılaştırmalı etkinliğini değerlendirme konusunda bir araştırma eksikliği bulunmaktadır. Bu çalışma, Muş bölgesindeki günlük rüzgar hızı verilerinde beş farklı LSTM modelinin (Vanilla LSTM, Stacked LSTM, Bidirectional LSTM, Attention LSTM ve Residual LSTM) tahmin performansını değerlendirmeyi amaçlamaktadır. Muş Meteoroloji Müdürlüğü'nden elde edilen veri seti, 1969-2023 yıllarını kapsayan 20.088 günlük rüzgar hızı ölçümünden oluşmaktadır. Sonuçlar, Vanilla LSTM'in en düşük MSE ve MAE değerleri ile genel doğrulukta üstün performans gösterdiğini, Attention LSTM'in ise en düşük MAPE ile yüzdesel doğruluk açısından daha başarılı olduğunu göstermektedir. Bu bulgular, Vanilla LSTM ve Attention LSTM modellerinin Muş veri setinde rüzgar hızı tahmini için en etkili modeller olduğunu ortaya koymaktadır. Bu modeller arasındaki seçim, toplam hata veya yüzdesel hata önceliklendirilmesine bağlı olarak stratejik bir çerçeve sunmaktadır.

1. INTRODUCTION

In today's world, climate change and environmental impacts have made the use of sustainable and renewable resources in energy production imperative. Wind energy is recognized as one of the most crucial renewable energy sources. Precise wind speed prediction is essential for maximizing the efficiency of wind energy utilization. These predictions are used in various fields, including energy production forecasts, turbine placement, and energy grid management.

Wind energy is essentially a byproduct of the sun. The uneven heating of the atmosphere by the sun, the irregular surfaces of the Earth (such as mountains and valleys), and the planet's rotation around the sun all combine to create wind. Because wind is abundant and will persist as long as the sun heats the Earth, it is a sustainable resource. Wind energy is a clean and renewable power source. Wind turbines use the power of the wind to drive a generator and produce electricity. Wind generates electricity without burning fuel or creating air pollution.

Land-based, large-scale wind turbines are among the most cost-effective energy sources available today. Moreover, the cost efficiency of wind energy continues to improve with advancements in wind energy science and technology. Wind energy can be seamlessly incorporated into rural or isolated areas, including farms, mountainous regions, or coastal and island areas, where strong wind resources are frequently present. These prime wind locations are usually found in regions with sparse populations.

Wind farms impact the environment differently than traditional power plants, but there are still concerns about the noise from turbine blades and the visual impact on the scenery. There are also impacts on wildlife.

Turkey plans to significantly increase its renewable energy capacity by 2035, with substantial growth in solar, wind, and hydroelectric energy. Given the growing preference for renewable sources in new installations, it is anticipated that renewable energy will account for 64.7% of the total installed capacity. Turkey aims to achieve net zero emissions by 2053. By 2035, the installed capacity of wind energy is projected to reach 29.6 GW (24.6 GW onshore, 5 GW offshore). Wind and solar are considered intermittent renewable energy sources. (Turkey National Energy Plan (2020-2035), 2022)

Energy production and consumption require extensive planning skills. The greatest support for this planning comes from advancing technology. The increase in data and the development of artificial intelligence technologies have significantly aided this planning. Long-term recorded meteorological data serve as a guide in predicting energy production and consumption. Some of this data consists of a series of data points or observations recorded at different or regular intervals. Defined as time series data, this information can be analyzed using artificial intelligence techniques, and it can be used for future predictions.

Time series forecasting is a method that predicts future events based on historical data. Recent advancements in this field have been substantial, particularly with the rise of deep learning techniques. Long Short-Term Memory (LSTM) networks have gained significant attention for their effectiveness in time series forecasting. LSTM networks have demonstrated remarkable success in predicting meteorological time series data due to their capacity to capture long-term dependencies, distinguishing them from other forecasting models [1].

This study was conducted to analyze regional wind energy and predict wind energy using machine learning methods, utilizing long-term daily average wind speed data from Muş Province. For this study, 54 years of daily average wind speed time series data, spanning from 1969 to 2023, were obtained from the Muş Central Meteorology Measurement Station (38°45'03.3"N 41°30'08.1"E). The dataset comprises a total of 20,089 daily records. Each record in our dataset includes the day, month, year, and average wind speed. Wind measurements were taken at a height of 10 meters above ground level. The central region of Muş does not experience strong winds. However, certain areas of Muş hold potential for wind energy production. Consequently, various projects have been initiated to harness wind energy in Muş, and it is anticipated that the number of such projects will increase over time[2].

The primary objective of this study is to predict future wind speeds using wind speed data from Muş Province spanning from 1969 to 2023. The aim is to provide analyses to support decision-makers in showcasing the regional wind energy potential. Additionally, the study seeks to evaluate the performance of the Long Short-Term Memory (LSTM) model and examine the impact of different LSTM model selections on performance. For this purpose, forecasts will be made using various LSTM models. Efforts have been undertaken to boost the use of wind energy, a clean and renewable source, for generating electricity, in support of the goal to achieve a zero-carbon footprint.

2. LITERATURE REVIEW

Studies on wind speed forecasting typically focus on physical models, statistical methods, machine learning and deep learning techniques, as well as combined and hybrid models [3].

Physical methods rely on data such as terrain, topography, obstacles, atmospheric pressure, and ambient temperature for predictions. These models, however, demand significant computational resources and require detailed information about various weather variables, which may not always be available [4].

Statistical methods include techniques such as Kalman filtering, ARIMA, and wavelet transform [5]. Autoregressive Integrated Moving Average (ARIMA) models are commonly used [6]. However, the accuracy of these models depends on the characteristics of the dataset and model parameters, and they often fall short in complex and variable datasets. Statistical methods are particularly less favored when data is nonlinear.

Machine learning techniques offer more flexible and powerful forecasting capabilities compared to statistical methods. Techniques such as Support Vector Machines (SVM) [7] and Decision Trees [8] are commonly used for wind speed prediction. However, the performance of these techniques often varies depending on the size and complexity of the dataset.

Deep learning techniques, particularly in large and complex datasets, exhibit strong performance. Recurrent Neural Networks (RNNs) are commonly used for forecasting temporal dependencies [9]. However, RNNs face challenges in learning long data sequences. Long Short-Term Memory (LSTM) networks address these issues. LSTM networks play a significant role, especially in predicting time series data. Studies have demonstrated that LSTM networks perform exceptionally well in wind speed forecasting [10].

Köse and Güneşer [11] evaluated the annual wind speed distribution and wind power density at seven stations in the Western Black Sea Region of Turkey for the period 2010–2014. The results indicate that, with the exception of Sinop, the region does not have sufficient wind energy potential for investment in wind energy.

Wadi et al. [12] carried out a technical evaluation of Turkey's wind energy potential. They conducted a feasibility study using hourly wind speed data recorded at a height of 30 meters in the Çatalca district from 2008 to 2010, aiming to assess the potential use of wind energy in Turkey. The Weibull two-parameter probability function was employed to estimate monthly and annual wind potential and power density, utilizing three different calculation methods. The simulation outcomes indicated that the studied area is appropriate for establishing large-scale wind farms.

Onat and Ersoz [13] examined the wind climate characteristics and energy potentials of three regions in Turkey. They used a five-layer Sugeno-type ANFIS model to identify the relationship between wind speed and other climatic variables to determine the wind characteristics in these regions. In the second phase, they employed WASP software to analyze the wind energy potential using wind speed data. Finally, the study calculated the technical electricity output and capacity utilization rates of installed turbines if wind farms were to be established in the selected regions.

Arslan et al. [14] analyzed the wind speed variability across Turkey and its influence on electricity production from 1980 to 2013. The study utilized reliable data from 77 stations. Hourly average wind speeds of 3.80 m/s or higher were recorded at the Gökçeada, Çanakkale, and Mardin stations, located in the Aegean, Southeastern, and Marmara regions of Turkey, respectively. Wind energy potential was assessed using the Weibull distribution. The findings reveal that Çatalca boasts the highest wind energy potential in Turkey, not only due to its high wind speeds but also because of its vast rural areas suitable for wind farm development.

Sırdaş [15] used harmonic analysis to model daily wind speed data collected from ten stations in Turkey's Marmara region between 1993 and 1997, accounting for various meteorological conditions. Notable differences in wind patterns were found between the western and eastern parts of the Marmara region. The study involved calculating the contribution of each harmonic component to the total variance, which led to the creation of regional variance maps.

Shao et al. [16] introduced a wind speed forecasting model utilizing LSTM neural networks, optimized with the Firework Algorithm (FWA) for tuning hyperparameters. The performance of this optimized model was assessed against other deep learning and regression-based wind speed prediction methods. The findings revealed that the LSTM model enhanced with FWA achieved lower prediction errors compared to alternative wind speed forecasting models.

Pradhan et al. [17] created a hybrid model for forecasting wind speed, consisting of two phases. The first phase involves breaking down wind speed sample data using wavelet techniques, while the second phase uses this decomposed data for predictions with a Recurrent Wavelet Neural Network (RWNN). To evaluate the model's performance, it was compared with traditional Recurrent Neural Network (RNN) forecasting approaches. The results from real-world data highlighted the model's effectiveness in terms of average absolute error and convergence rate.

Lu et al. [18] performed an extensive review of metaheuristic optimization techniques for forecasting wind energy. They created a detailed classification system for these algorithms to enhance the optimization of wind energy forecasting model parameters. The algorithms are designed to discover optimal solutions within constraints, which are essential for fine-tuning the primary parameters of forecasting models. They also proposed a thorough and scientific multi-error evaluation framework for analyzing wind energy forecasting errors. This review covers various error evaluation methods, including deterministic, uncertainty, and testing approaches. Additionally, it offers a quantitative analysis of the strengths, weaknesses, accuracy, and computational costs associated with these methods.

Hu et al. [19] aimed to enhance wind speed forecasting accuracy with their LSTMDE-HELM approach. This method combines LSTM networks, Hysteretic Extreme Learning Machine (HELM), Differential Evolution (DE) algorithm, and nonlinear hybrid mechanisms. To improve the Extreme Learning Machine (ELM) performance, they integrated a hysteretic biological neural feature into the ELM's neuron activation function. Furthermore, because the ideal number of hidden layers and neurons per layer in the LSTM was not initially clear, the DE algorithm was used to optimize these parameters. This approach aims to strike a balance between learning accuracy and model complexity. The hybrid model was evaluated using data from a wind farm in Inner Mongolia, China, with two forecasting intervals: ten minutes (short-term) and one hour (medium-term). The findings reveal that this hybrid method outperforms other models across four performance metrics and in statistical assessments.

Chen et al. [20] developed a new two-layer nonlinear combination method called EEL-ELM for short-term wind speed forecasting problems, such as ten minutes and one hour ahead predictions. To demonstrate the effectiveness of the proposed EEL-ELM method, two real-world case studies from a wind farm in Inner Mongolia, China, were applied. Simulation results reveal that EEL-ELM achieved better forecasting performance compared to eight other wind speed forecasting methods, based on three evaluation metrics and three statistical tests.

Alhussan et al. [21] proposed an improved model for enhancing wind speed forecasting accuracy. They utilized a novel optimization algorithm known as Generalized Adaptive Differential Evolution (GADTO), which integrates Dipper-Throated Optimization (DTO) with Genetic Algorithm (GA). This optimization technique was applied to fine-tune the parameters of a Bidirectional LSTM (BiLSTM) forecasting model. To assess the statistical significance of their approach compared to existing methods, they employed variance analysis (ANOVA) and Wilcoxon signed-rank tests. The findings confirmed the statistical significance and reliability of their method, achieving a mean root mean square error (RMSE) of 0.00046, which outperforms the accuracy of other new forecasting methods.

Subramani et al.'s [27] study reviews advancements in renewable energy, focusing on solid oxide fuel cells and electrolysers for green hydrogen production. Highlighting the significance of wind energy, it emphasizes accurate forecasting for efficient energy management. Machine learning methods like Support Vector Regression (SVR) and Random Forest have improved prediction accuracy. The paper also explores challenges like uncertainty in renewable energy production, data availability, and model interpretability, aiming to enhance grid integration and support a sustainable future.

A key challenge in wind energy utilization is its variability across seasonal and interannual timescales due to atmospheric changes. Yang et al.'s study [28] highlights a model's ability to provide skillful seasonal wind energy predictions in the U.S. Great Plains, particularly during peak energy seasons (winter and spring). The model leverages year-to-year variations in the El Niño-Southern Oscillation, which influence large-scale wind and storm patterns. In the Southern Great Plains, it predicts significant wind energy changes months in advance with high accuracy. This capability supports optimizing wind energy use during peak production periods.

In wind forecasting, different time intervals are used in the literature (long term: >3 days, medium term: a few hours - 3 days, short term: a few minutes - a few hours). However, these time periods are not fixed and vary according to researchers and needs [22, 23]. Some studies have divided the time intervals into four categories as "very short", "short", "medium" and "long", each of which has been aimed at different areas of use [24]. In addition, some other studies have determined intervals of a few hours for the short term, a few hours to 3 days for the medium term and >6 days for the long term [25, 26]. This study, which makes daily forecasts, can be accepted as medium term when evaluated according to different studies.

3. MATERIAL AND METHOD

3.1. Data and Wind in Muş

Data obtained from the Muş Provincial Directorate of Meteorology were preprocessed and prepared for application to the LSTM model. Due to technical reasons, dates with missing measurements were identified, and the missing values for these dates were replaced with the averages of corresponding days and months from 1969 to 2023. The dataset includes 20,088 daily wind speed measurements. Measurement values for the specified time interval are shown in Figure 1. 80% of the data was used for training, while the remaining data was used for testing. Additionally, 20% of the training data was reserved for validation. The data was normalized to a scale between 0 and 1.

In Muş, summers are characterized by hot, dry, and clear weather, while winters are known for being freezing, snowy, and partly cloudy. Annual temperatures generally vary between -20°C and +30°C [29]. Wind patterns in any area are strongly affected by local topography and other environmental factors, resulting in more pronounced variations in wind speed and direction at any given moment compared to average conditions.

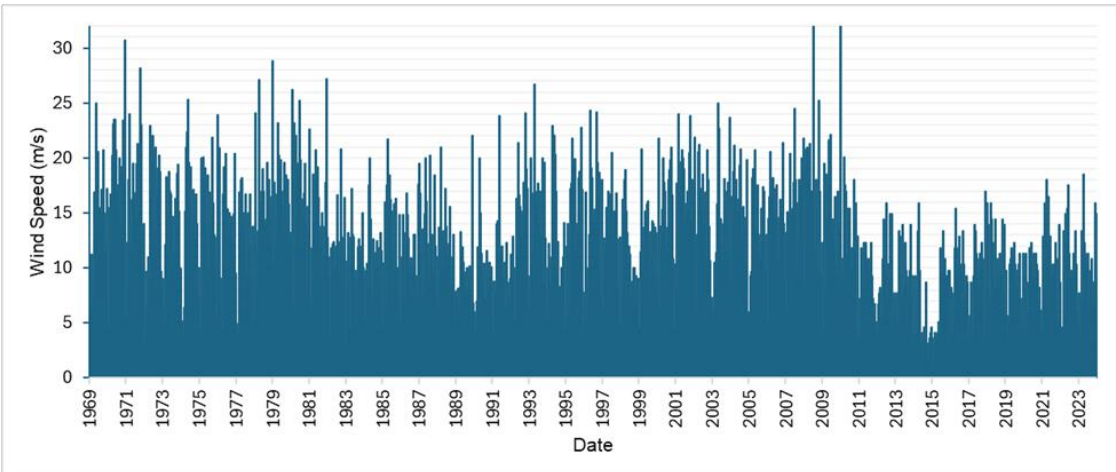

Figure 1. Average Daily Wind Speed of Muş

Figure 2 provides information on the wind speed distribution in Muş Province. It is observed that the most common daily average wind speed range is between 10 and 20 m/s. The average wind speed in the dataset is approximately 6.31 m/s, with a standard deviation of around 3.88 m/s. This standard deviation of 3.88 m/s indicates a wide dispersion of wind speed values around the mean in the dataset.

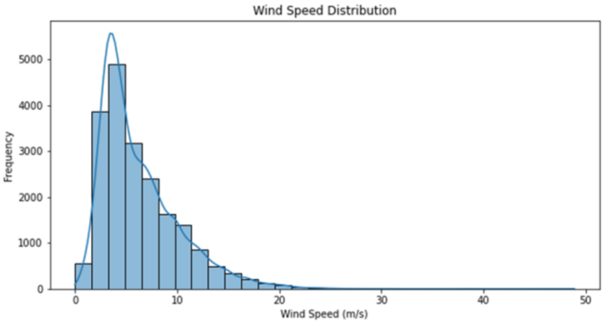

Figure 2. Wind Speed Distribution in Muş

Examining the boxplot in Figure 3, the minimum wind speed is 0 m/s, indicating that there were some days with no wind. This could potentially be a measurement error. It is observed that 25% of the wind speed values in the dataset are below 3.5 m/s, which is the first quartile. The median wind speed, which is the 50th percentile, is below 5.1 m/s. While the annual average wind speed is 6.31 m/s, the median is 5.1 m/s. The fact that the mean is slightly higher than the median suggests that a few high wind speed values (outliers) are raising the average. Additionally, 75% of the wind speed values are below 8.2 m/s, indicating that a large portion of the dataset has moderate wind speeds. The wide distribution of the data is an important factor to consider in the modeling process. The highest recorded wind speed is 48.9 m/s, which suggests the presence of very strong winds.

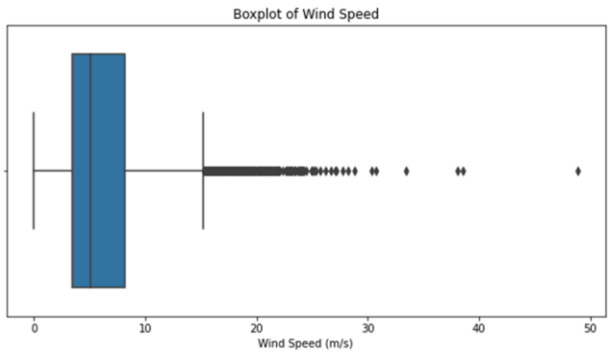

Figure 3. Boxplot of Wind Speed in Muş

For Muş, considering the hourly average wind vector (speed and direction), the average daily wind speed shows slight seasonal variations throughout the year. Figure 4 presents the monthly average wind speeds for the years 1969-2023. The windiest period of the year, characterized by an average wind speed exceeding 7 m/s, lasts from April to September. May is the windiest month in Muş, with an hourly average wind speed of around 8.57 m/s. Conversely, January is the calmest month, with an hourly average wind speed of approximately 4 m/s.

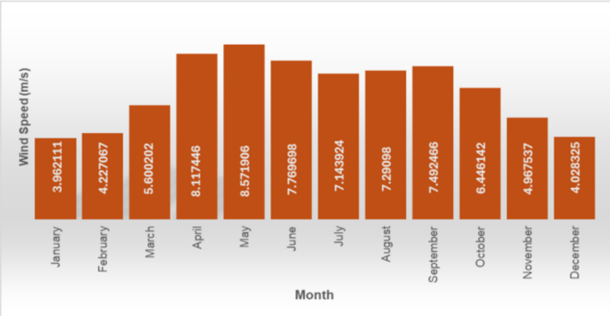

Figure 4. Monthly average wind speed

Examining the monthly and annual average wind speeds in Figure 5, some years show noticeable increases or decreases. The year with the highest annual average wind speed was 1979, with an average speed of 9.33 m/s. In contrast, 2014 had the lowest average wind speed at 2.27 m/s.

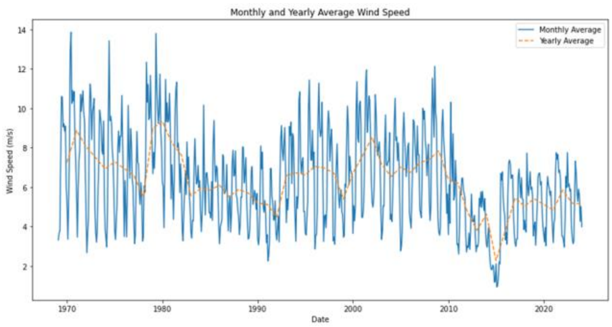

Figure 5. Monthly and Yearly Average Wind Speed in Muş

In Figure 6, the “Observed” section represents our original time series data, encompassing all changes in wind speed over time. This data includes seasonal variations, long-term trends, and random fluctuations. The “Trend” component illustrates long-term trends over time, showing whether wind speed generally increases or decreases. In Muş, we can observe both long-term increases and decreases in wind speed. The “Seasonal” component represents regular, repeating changes occurring at specific times of the year. The “Residual” component refers to random fluctuations that the model does not explain. It represents what remains after removing the trend and seasonal components from the observed data. This component indicates irregular and unpredictable changes in the dataset. Significant fluctuations in the residual component may suggest that the model does not fully capture all the dynamics of the data. Variations can be observed in certain time intervals.

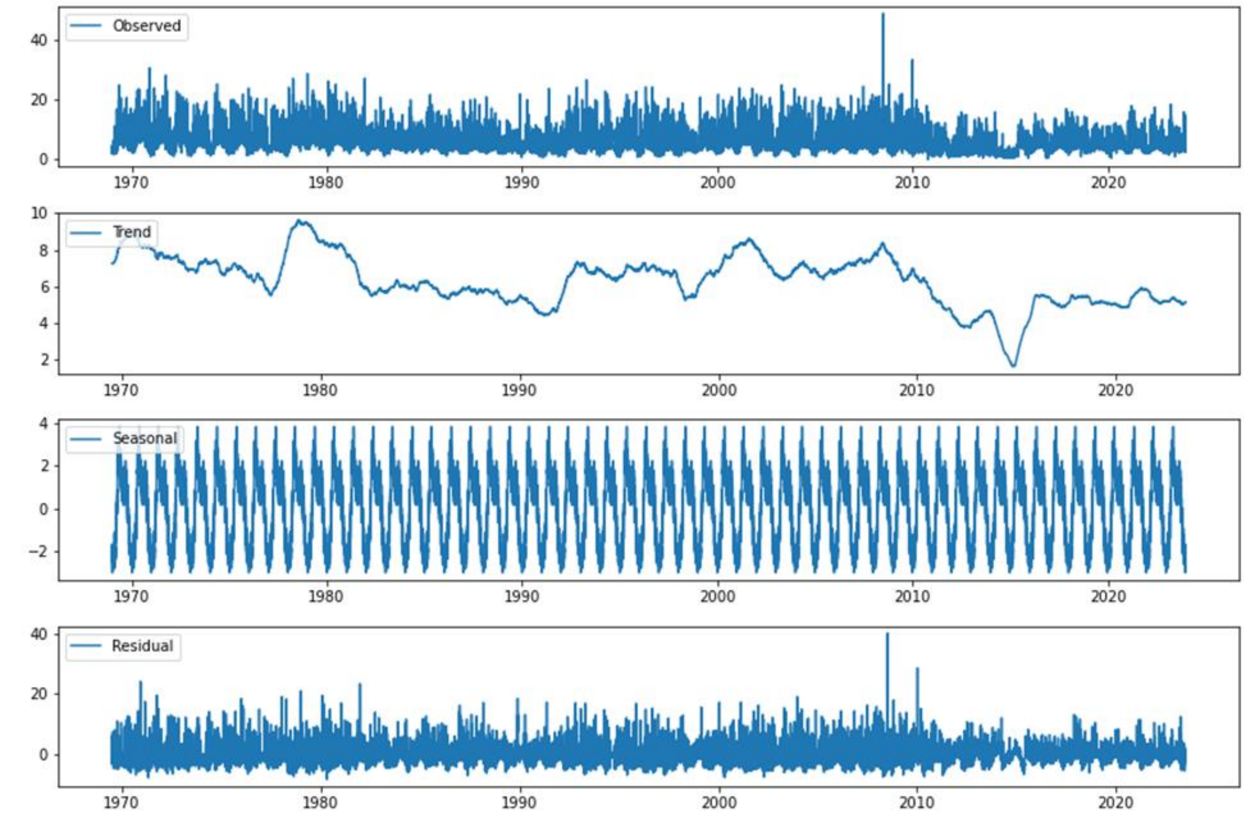

Figure 6. Observed, Trend, Seasonal, Residual behavior of wind speed in Muş

3.2. Long-Short Term Memory

LSTM networks are a type of artificial neural network commonly used for modeling time series data. They are designed to address the long-term dependency problems encountered by traditional Recurrent Neural Networks (RNNs). LSTMs are highly effective for sequence prediction tasks because they can retain information over long periods. The architecture of an LSTM can be visualized as a series of repeating "blocks" or "cells". An LSTM network consists of five fundamental components: hidden state (h_t), cell state (c_t), forget gate (f_t), input gate (i_t) and output gate (o_t). These components work together to control how information is stored, updated, and retrieved, enabling LSTMs to handle complex time series and sequence data effectively.

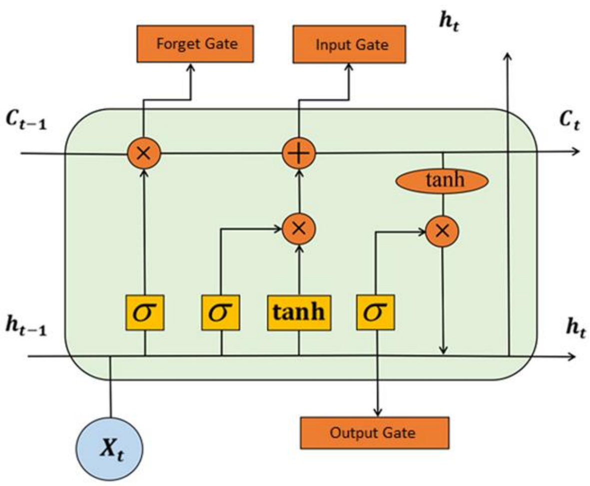

Figure 7. The architecture of an LSTM

Forget Gate

The forget gate determines which information from the cell state should be discarded. The output of the forget gate is computed using the sigmoid activation function.

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \tag{1}$$

σ is the sigmoid function. W_f is the weight matrix for the forget gate. b_f is the bias vector for the forget gate. h_{t-1}, is the hidden state from the previous time step. x_t is the input vector at the current time step. The forget gate's output, f_t, is a vector with values between 0 and 1, indicating how much of each component of the cell state should be retained or forgotten [30].

Input Gate

The input gate controls how new information is added to the cell state. This process occurs in two stages: first, it determines how much of the new information should be updated; second, it generates candidate values for the cell state.
The input gate's output and the candidate values for the cell state are determined through the following calculations:

$$i_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \tag{2}$$

$$\tilde{c}_t = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c) \tag{3}$$

In summary, the input gate determines which portions of the new information will be incorporated into the cell state, while the candidate cell state offers the potential new information for addition [31].

Updating the Cell State

The cell state is updated using the outputs from the forget gate and the input gate:

$$c_t = f_t \cdot c_{t-1} + i_t \cdot \tilde{c}_t \quad (4)$$

c_t is the updated cell state. c_{t-1} is the cell state from the previous time step. $f_t \cdot c_{t-1}$ represents the amount of information retained from the previous cell state. $i_t \cdot \tilde{c}_t$ represents the amount of new information added to the cell state. This mechanism allows the LSTM to maintain long-term dependencies by effectively managing and updating the cell state over time.

Output Gate

The output gate determines which information from the cell state will be outputted. The output gate's output is calculated as follows:

$$i_o = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \quad (5)$$

The hidden state (h_t) is then calculated using the output gate's output and the updated cell state:

$$h_t = o_t \cdot \tanh(c_t) \quad (6)$$

Through these equations and gates, the LSTM model effectively manages long-term dependencies and overcomes the long-term dependency problems faced by traditional RNNs.

3.3. Performance Evaluation

Five different LSTM architectures were employed to make predictions, and their performance was compared. Each model's predictions were assessed against the actual values using various error metrics. The training process was conducted over 100 epochs, and minibatches consisting of 32 samples were utilized. The prediction performance was evaluated using the following three loss function metrics:

Mean Squared Error (MSE): It calculates the mean of the squared differences between predicted and observed values. MSE assigns greater weight to larger errors compared to smaller ones, which makes it effective for evaluating the extent of prediction errors. Smaller MSE values suggest that the model's predictions are more accurate.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \quad (7)$$

Mean Absolute Error (MAE): It assesses the mean of the absolute differences between predicted and actual values. MAE gives equal weight to all errors, without emphasizing larger errors over smaller ones. Lower MAE values indicate a higher overall accuracy of the model's predictions.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \quad (8)$$

Mean Absolute Percentage Error (MAPE): It calculates the average of the absolute differences between predicted and actual values, expressed as a percentage of the actual values. MAPE offers a percentage-based evaluation of prediction errors, with lower MAPE values reflecting a smaller percentage of error in the predictions.

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{(y_i - \hat{y}_i)}{y_i}\right| x100 \quad (9)$$

In all models, the LSTM layer contained a default of 50 neurons. The dropout rate was set at the default value of 0.2, the learning rate was maintained at 0.001, and the Adam optimization algorithm was used. These parameters are significant factors influencing each model's training and prediction performance. The evaluation determined which model demonstrated superior performance.

3.4. LSTM Models

Vanilla LSTM

The Vanilla LSTM is a single-layer, straightforward LSTM network. Initially proposed by Hochreiter and Schmidhuber (1997), this model is commonly used for processing time series and sequential data [1]. LSTM cells are capable of storing and updating information over time, making them effective for learning long-term dependencies. However, the disadvantage of the Vanilla LSTM is that its performance can be limited, especially on very deep and complex data. The model may tend to forget the information it has learned over time, making it less effective at learning more complex relationships. Its advantages include its ability to learn long-term dependencies and its generally easy implementation [32]. A Vanilla LSTM model typically consists of an input layer, an LSTM layer, and an output layer. Input Layer receives a sequence containing time steps and features. LSTM Layer processes the data by updating the cell state and hidden state. The output layer is usually a dense layer and produces the final predictions. Vanilla LSTM networks are utilized in various domains, including time series forecasting, language modeling, machine translation, and speech recognition. In this study, the model comprises a single LSTM layer, a dropout layer, and an output layer as seen in Figure 8.

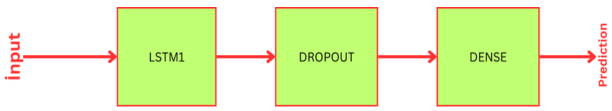

Figure 8. Vanilla LSTM

Stacked LSTM

The Stacked LSTM network consists of multiple LSTM layers stacked on top of each other. The advantage of this structure is that it increases the capacity to learn more complex and abstract features. Deeper structures generally provide stronger model performance. However, the disadvantage of Stacked LSTM is that it carries the risk of overfitting, as the model requires larger parameter sets and more computational power [33].

A Stacked LSTM model typically includes an input layer, multiple LSTM layers, and an output layer. Stacked LSTM networks are used for handling complex sequential data tasks, including natural language processing (NLP) and sophisticated time series forecasting [33]. In this study, the model architecture stacks two LSTM layers as seen in Figure 9. The first LSTM layer generates outputs for all time steps, which are then fed into the second LSTM layer. A dropout layer follows each LSTM layer.

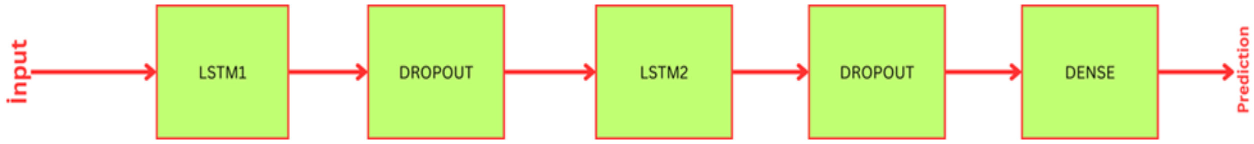

Figure 9. Stacked LSTM

Bidirectional LSTM

The Bidirectional LSTM network analyzes input data in both forward and backward directions, enabling the model to capture contextual information from both past and future time steps at each stage. However, the disadvantage of this model is the increased computational cost due to bidirectional learning [34].
Typically, a Bidirectional LSTM model includes an input layer, a bidirectional LSTM layer, and an output layer. The Bidirectional LSTM layer comprises two distinct LSTM layers: one processes data in the forward direction and the other in the backward direction. This architecture is particularly useful for tasks that benefit from understanding context from both directions, such as natural language processing (NLP), bioinformatics, speech recognition, and sentiment analysis. [10, 35]. In this study, the model includes two bidirectional LSTM layers, each followed by a dropout layer as seen in Figure 10. Each LSTM layer is configured to operate in both forward and backward directions.

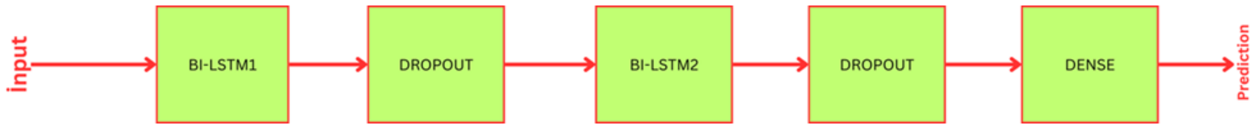

Figure 10. Bidirectional LSTM

Attention LSTM

The attention mechanism enables the model to prioritize different input time steps differently, particularly in noisy and fluctuating data. When integrated into LSTM networks, known as Attention LSTM, this mechanism allows the model to concentrate on key time steps, improving its overall performance. The attention mechanism allows the model to focus on important information, so it can go beyond the limited memory capacity of LSTM. The advantage of this model is that it can make more accurate predictions, especially in data with long-term dependencies. The disadvantage is that it requires additional computational load and more time to train the model [36].

An Attention LSTM model typically includes an input layer, an LSTM layer, an attention layer, and an output layer. Attention LSTM networks are particularly effective for tasks that require emphasis on specific segments of the input, such as machine translation, image captioning, speech recognition, and text summarization [37, 38]. In this study, the model includes the attention mechanism in addition to LSTM layers as seen in Figure 11. The first LSTM layer generates outputs for all time steps, which are then processed by the attention layer

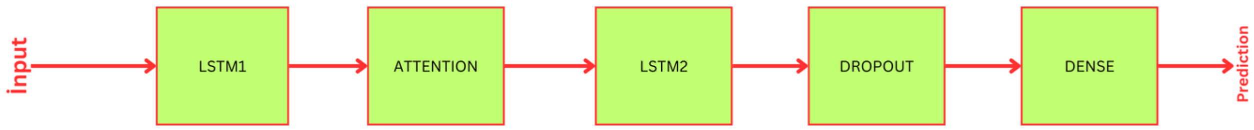

Figure 11. Attention LSTM

Residual LSTM

Residual LSTM networks use residual connections to overcome the vanishing gradient problem commonly found in deep networks. These connections help preserve the flow of information between LSTM layers, making it possible to train deeper models more effectively. Its advantages are that it provides a more efficient learning process and is resistant to the vanishing gradient problem. However, the disadvantage of Residual LSTM is that the model becomes more complex with additional layers and parameters, and therefore carries the risk of overfitting [39].

A Residual LSTM model generally includes input layer, multiple LSTM layers, residual connections and output layer. Residual connections facilitate information flow between LSTM layers by bypassing some layers, thus mitigating the vanishing gradient problem. Residual LSTMs are used in tasks requiring very deep networks, such as advanced time series analysis and complex sequence modeling [40, 41]. In this study, the residual model comprises three LSTM layers connected with residual connection as seen in Figure 12. The first two LSTM layers generate outputs for all time steps. The third LSTM layer integrates with the last time step of the first LSTM layer via an Add layer to form a residual connection. The model is completed with a dropout layer and an output layer.

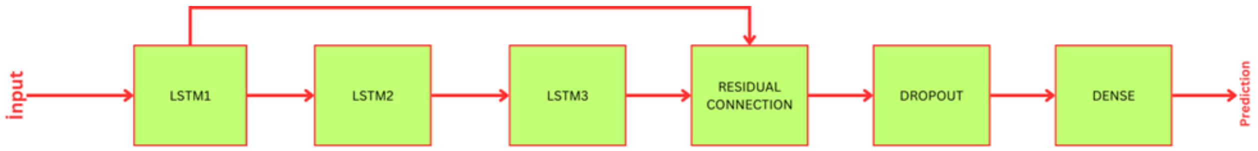

Figure 12. Residual LSTM

4. RESULTS

When evaluating LSTM models, it is crucial to consider both their performance and training time to determine which model is most appropriate for a given application scenario. In this study, various LSTM models have been compared specifically for wind speed forecasting. This comparison helps identify the model that best balances prediction accuracy and computational efficiency for the task at hand. This comparison does not mean that these models always give similar ranking results. Comparing models with the same hyperparameters (learning rate, dropout rate, number of LSTM units) allows you to assume that the performance differences are due solely to the model architectures. This approach provides a good starting point to see the relative differences between the models. It makes the results more methodologically meaningful. However, different structures of LSTM architectures may give different responses to the same hyperparameter set. Since Stacked and Residual LSTM contain more parameters, they may need different learning rate or dropout rate values to reach optimum performance. Attention and Bidirectional LSTM have more information processing capacity and therefore may work better with a different number of units (number of LSTM cells). Rather than comparing the models, it would be healthier to perform hyperparameter optimization to obtain better results and fully evaluate the potential of each model. This study did not focus on suitable hyperparameters.

Table 1. Performance Evaluation of LSTM models

Model	MSE	Time	MAE	Time	MAPE	Time
Vanilla LSTM	**4.218**	2s 9ms/step	**1.410**	1s 8ms/step	0.299	1s 6ms/step
Stacked LSTM	4.258	3s 15ms/step	1.441	2s 13ms/step	0.305	2s 11ms/step
Bidirectional LSTM	4.384	4s 18ms/step	1.470	4s 21ms/step	0.299	3s 15ms/step
Attention LSTM	4.256	3s 14ms/step	1.505	3s 15ms/step	**0.288**	1s 8ms/step
Residual LSTM	4.373	2s 10ms/step	1.425	3s 18ms/step	0.302	2s 12ms/step

Table 1 offers a comparison of performance metrics to determine the most effective LSTM model for wind speed forecasting for Muş dataset. Mean Squared Error (MSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE) are metrics. These metrics allow for a comprehensive assessment of the models' prediction accuracy from different perspectives. Additionally, training times help evaluate the model's complexity and computational efficiency.

The Vanilla LSTM model shows the lowest MSE and MAE values, demonstrating its superior overall performance. It can be seen in Figure 13 and Figure 14. Its MAPE value is also moderate compared to other models, suggesting a reasonable level of percentage error in predictions. Given its simpler architecture, the Vanilla LSTM has the shortest training time, making it a favorable option for scenarios that require rapid model training.

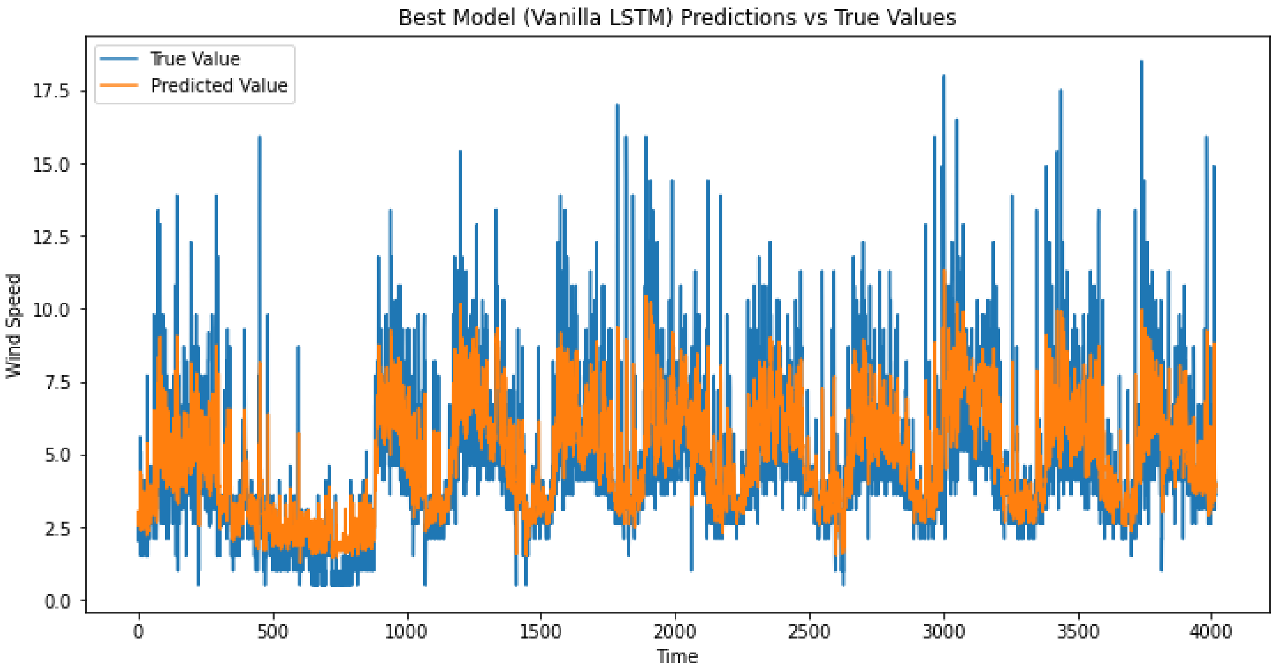

Figure 13. Vanilla LSTM predictions with MSE loss function

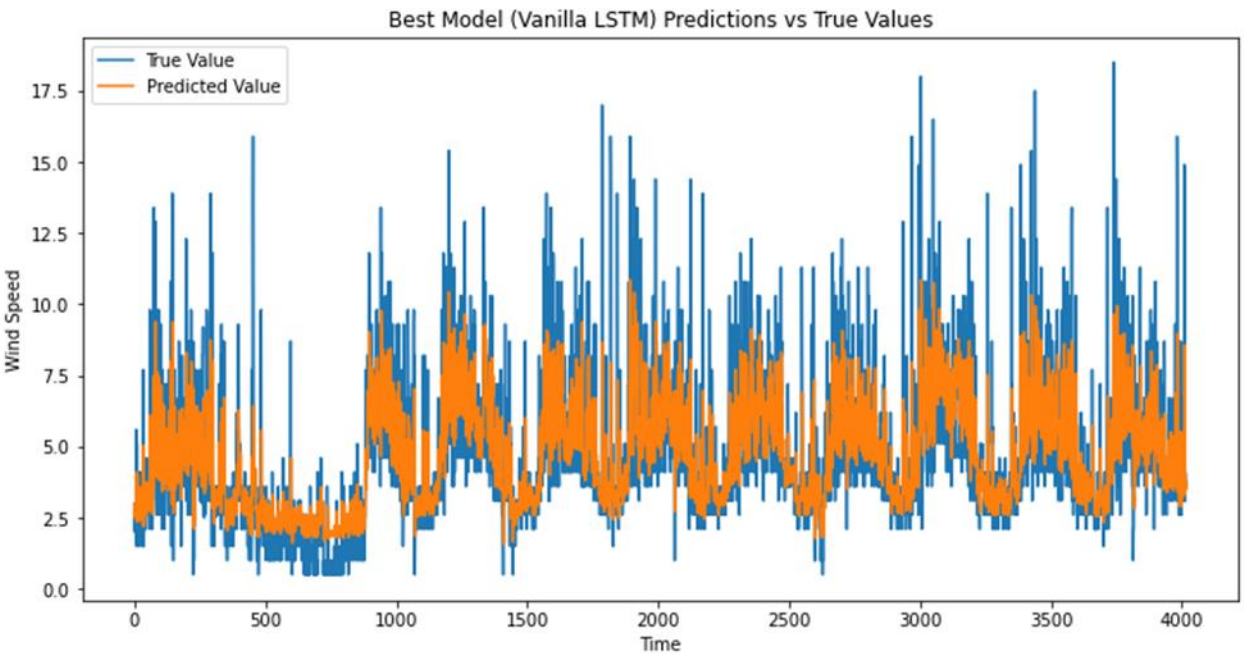

Figure 14. Vanilla LSTM predictions with MAE loss function

The Attention LSTM model has a moderate MSE value but achieves the lowest MAPE, indicating that it makes the least percentage error in predictions. Despite having a somewhat higher MAE value, the model's ability to focus on significant time steps has reduced the overall percentage error rate. The inclusion of the attention mechanism has not significantly increased the training time, suggesting that the Attention LSTM can provide rapid training even with the addition of this mechanism. Due to the nature of time series such as wind, Attention models are better able to capture trends and sudden changes.

The Residual LSTM model shows higher error rates than other models for both MSE and MAE. Its MAPE value is average. The residual connections have not significantly improved the model's performance on this dataset. In terms of training time, the model shows an average performance.

Overall, the Attention LSTM model demonstrates the best performance in terms of MAPE as seen in Figure 15. This indicates that the model is more consistent and accurate in percentage terms. However, the Vanilla LSTM model has the lowest MSE and MAE values, showing the best performance in terms of total error amount.

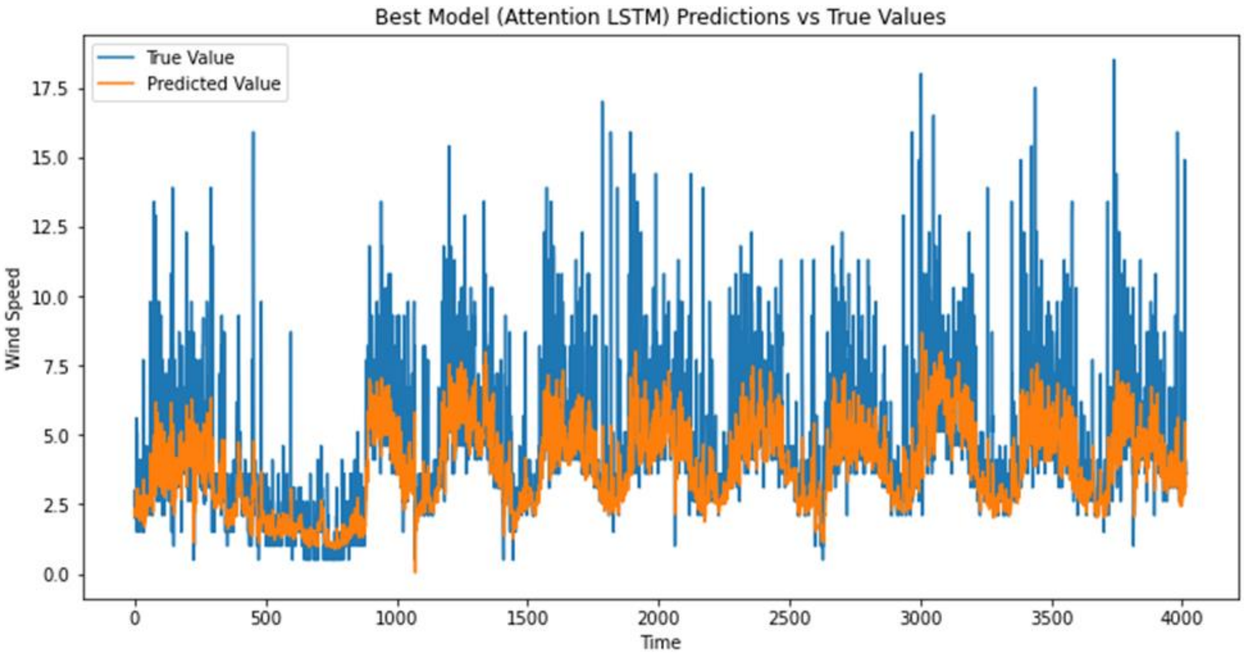

Figure 15. Attention LSTM predictions with MAPE loss function

Based on the results, it would be beneficial to focus on optimizing and fine-tuning the better-performing Vanilla and Attention LSTM models. If the percentage error rate is higher, the Attention LSTM model should be preferred. However, if minimizing the total error amount is the goal, the Vanilla LSTM model is a more suitable option.

Table 2. Total parameters of Models

Model	Total Parameter Numbers
Vanilla	10,451
Stacked	30,651
Attention	30,651
Residual	50,851
Bidirectional	81,301

Table 2 shows the total numbers of parameters for each model. Total parameter numbers reflect the complexity and computational requirements of each model. Vanilla LSTM (10,451 parameters) is suitable for small data sets due to its simplicity and fast training. Stacked and Attention LSTM models (30,651 parameters) can learn more complex dependencies; Attention provides a more focused context between time steps, while Stacked provides a broader perspective. Residual LSTM (50,851 parameters) is advantageous in cases that require deep structures and aim to reduce gradient loss. Bidirectional LSTM (81,301 parameters) can provide the strongest results by learning the past and future simultaneously, but it carries the highest computational cost and overfitting risk. If there is no significant difference between the results, models with fewer parameters may be preferred.

The difference between the models is higher in comparison in terms of time. It is seen that there is no extreme difference in the results in terms of error metrics. Since there is no significant difference between error metrics, simpler or faster models may be preferred. If your application requires real-time prediction, models that require less computation time (e.g. Vanilla LSTM) should be preferred. Attention and Bidirectional LSTM models generally learn more information, but in time series such as wind this can create an unnecessary time cost if this extra information is not useful. Therefore, the model to be chosen also depends on the dataset.

5. CONCLUSION

This study compared the performance of five distinct LSTM architectures for wind speed prediction in Muş, Turkey. The findings indicate that both Vanilla and Attention LSTM models are effective for wind speed prediction in Muş, with the Vanilla LSTM model excelling in total error minimization and the Attention LSTM model excelling in percentage error accuracy.

Overall, the study's findings underscore the efficacy of LSTM techniques in predicting wind speed and emphasize their potential for real-world applications. The success of LSTM models in predicting wind speed is a significant step for wind energy production and management. Accurate predictions can improve energy production planning and grid management processes.

In this study, an initial comparison was made with the same hyperparameters for all models, in a subsequent study, individual hyperparameter settings for each model should be optimized using Grid Search or Bayesian Optimization methods to get the best results. This makes it better for us to reveal the true potential of the models. Future work could involve further optimization and fine-tuning of these models to enhance their predictive capabilities. To further enhance the performance of the Vanilla and Attention LSTM models, hyperparameter optimization and additional data preprocessing techniques can be applied. The effects of different learning rates and dropout rates can also be explored.

Acknowledgement

I would like to thank MUŞ Provincial Directorate of Meteorology for their information sharing and support.

REFERENCES

[1] Hochreiter S, Schmidhuber J. Long Short-Term Memory. Neural Comput 1997; 9: 1735–1780.
[2] Governorship of Muş, www.mus.gov.tr.
[3] Shang Z, He Z, Chen Y, et al. Short-term wind speed forecasting system based on multivariate time series and multi-objective optimization. Energy 2022; 238: 122024.
[4] Wu C, Wang J, Chen X, et al. A novel hybrid system based on multi-objective optimization for wind speed forecasting. Renew Energy 2020; 146: 149–165.
[5] Aly HHH. An intelligent hybrid model of neuro Wavelet, time series and Recurrent Kalman Filter for wind speed forecasting. Sustain Energy Technol Assessments 2020; 41: 100802.
[6] Liu X, Lin Z, Feng Z. Short-term offshore wind speed forecast by seasonal ARIMA - A comparison against GRU and LSTM. Energy 2021; 227: 120492.
[7] He J, Xu J. Ultra-short-term wind speed forecasting based on support vector machine with combined kernel function and similar data. EURASIP J Wirel Commun Netw 2019; 2019: 248.
[8] Xu W, Ning L, Luo Y. Wind Speed Forecast Based on Post-Processing of Numerical Weather Predictions Using a Gradient Boosting Decision Tree Algorithm. Atmosphere (Basel) 2020; 11: 738.
[9] Wang Y, Zou R, Liu F, et al. A review of wind speed and wind power forecasting with deep neural networks. Appl Energy 2021; 304: 117766.
[10] Joseph LP, Deo RC, Prasad R, et al. Near real-time wind speed forecast model with bidirectional LSTM networks. Renew Energy 2023; 204: 39–58.
[11] Köse B, Güneser MT. Assessment of Wind Characteristics and Wind Energy Potential in West Black Sea Region Of Turkey. Eskişehir Tech Univ J Sci Technol A - Appl Sci Eng 2019; 20: 227–237.
[12] Wadi M, Kekezoglu B, Baysal M, et al. Feasibility Study of Wind Energy Potential in Turkey: Case Study of Catalca District in Istanbul. In: 2019 2nd International Conference on Smart Grid and Renewable Energy (SGRE). IEEE, pp. 1–6.
[13] Onat N, Ersoz S. Analysis of wind climate and wind energy potential of regions in Turkey. Energy 2011; 36: 148–156.
[14] Arslan H, Baltaci H, Akkoyunlu BO, et al. Wind speed variability and wind power potential over Turkey: Case studies for Çanakkale and İstanbul. Renew Energy 2020; 145: 1020–1032.
[15] Sirdaş S. Daily wind speed harmonic analysis for Marmara region in Turkey. Energy Convers Manag 2005; 46: 1267–1277.
[16] Shao B, Song D, Bian G, et al. Wind Speed Forecast Based on the LSTM Neural Network Optimized by the Firework Algorithm. Adv Mater Sci Eng 2021; 2021: 1–13.
[17] Pradhan PP, Subudhi B. Wind speed forecasting based on wavelet transformation and recurrent neural network. Int J Numer Model Electron Networks, Devices Fields; 33. Epub ahead of print 12 January 2020. DOI: 10.1002/jnm.2670.
[18] Lu P, Ye L, Zhao Y, et al. Review of meta-heuristic algorithms for wind power prediction: Methodologies, applications and challenges. Appl Energy 2021; 301: 117446.
[19] Hu Y-L, Chen L. A nonlinear hybrid wind speed forecasting model using LSTM network, hysteretic ELM and Differential Evolution algorithm. Energy Convers Manag 2018; 173: 123–142.
[20] Chen M-R, Zeng G-Q, Lu K-D, et al. A Two-Layer Nonlinear Combination Method for Short-Term Wind Speed Prediction Based on ELM, ENN, and LSTM. IEEE Internet Things J 2019; 6: 6997–7010.
[21] Alhussan AA, M. El-Kenawy E-S, Abdelhamid AA, et al. Wind speed forecasting using optimized bidirectional LSTM based on dipper throated and genetic optimization algorithms. Front Energy Res; 11. Epub ahead of print 1 June 2023. DOI: 10.3389/fenrg.2023.1172176.
[22] Chandra DR, Kumari MS, Sydulu M. A detailed literature review on wind forecasting. In: 2013 International Conference on Power, Energy and Control (ICPEC). 2013, pp. 630–634.
[23] Keren B., Sabitha K. Probabilistic forecasting of wind power generation using extreme learning machineinvolving bootstrap method. ITECH 2016; 04: 729–736.
[24] Azad HB, Mekhilef S, Ganapathy VG. Long-Term Wind Speed Forecasting and General Pattern Recognition Using Neural Networks. IEEE Trans Sustain Energy 2014; 5: 546–553.
[25] Wang X, Guo P, Huang X. A Review of Wind Power Forecasting Models. Energy Procedia 2011; 12: 770–778.
[26] Sharma N, Deo R. Wind speed forecasting in Nepal using self-organizing map-based online sequential extreme learning machine. In: Predictive Modelling for Energy Management and Power Systems Engineering. Elsevier, pp. 437–484.
[27] Subramani K, J SS, Habelalmateen MI, et al. Predicting Wind Energy: Machine Learning from Daily Wind Data. E3S Web Conf 2024; 540: 03009.
[28] Yang X, Delworth TL, Jia L, et al. Skillful seasonal prediction of wind energy resources in the contiguous United States. Commun Earth Environ 2024; 5: 313.
[29] Tugal I, Sevgin F. Analysis and forecasting of temperature using time series forecasting methods a case study of Mus. Therm Sci 2023; 27: 3081–3088.
[30] Lu J, Wang Y, Zhu Y, et al. DACLnet: A Dual-Attention-Mechanism CNN-LSTM Network for the Accurate Prediction of Nonlinear InSAR Deformation. Remote Sens 2024; 16: 2474.
[31] Lattari F, Rucci A, Matteucci M. A Deep Learning Approach for Change Points Detection in InSAR Time Series. IEEE Trans Geosci Remote Sens 2022; 60: 1–16.

[32] Wu Y, Yuan M, Dong S, et al. Remaining useful life estimation of engineered systems using vanilla LSTM neural networks. Neurocomputing 2018; 275: 167–179.

[33] Ma M, Liu C, Wei R, et al. Predicting machine's performance record using the stacked long short-term memory (LSTM) neural networks. J Appl Clin Med Phys; 23. Epub ahead of print 16 March 2022. DOI: 10.1002/acm2.13558.

[34] da Silva DG, Meneses AA de M. Comparing Long Short-Term Memory (LSTM) and bidirectional LSTM deep neural networks for power consumption prediction. Energy Reports 2023; 10: 3315–3334.

[35] Ghosh S, Ekbal A, Bhattacharyya P. Natural language processing and sentiment analysis: perspectives from computational intelligence. In: Computational Intelligence Applications for Text and Sentiment Data Analysis. Elsevier, pp. 17–47.

[36] Muhammad K, Mustaqeem, Ullah A, et al. Human action recognition using attention based LSTM network with dilated CNN features. Futur Gener Comput Syst 2021; 125: 820–830.

[37] Kang Q, Chen EJ, Li Z-C, et al. Attention-based LSTM predictive model for the attitude and position of shield machine in tunneling. Undergr Sp 2023; 13: 335–350.

[38] Wen X, Li W. Time Series Prediction Based on LSTM-Attention-LSTM Model. IEEE Access 2023; 11: 48322–48331.

[39] Yue B, Fu J, Liang J. Residual Recurrent Neural Networks for Learning Sequential Representations. Information 2018; 9: 56.

[40] Vatsa A, Hati AS, Kumar P, et al. Residual LSTM-based short duration forecasting of polarization current for effective assessment of transformers insulation. Sci Rep 2024; 14: 1369.

[41] Fu S, Zhang Y, Lin L, et al. Deep residual LSTM with domain-invariance for remaining useful life prediction across domains. Reliab Eng Syst Saf 2021; 216: 108012.