Rainfall Time-Series Prediction Using Neural Network Approach

Nor Shamira Sabri, Azleena Mohd Kassim, Noor Farizah Ibrahim, Mohd Heikal Husin, Mohd Shareduwan Mohd Kasihmuddin, and Ismail Ahmad Abir

***Abstract*—Rainfall prediction is essential in various domains involving agriculture, water resource management, irrigation planning, and disaster preparedness measures. The accuracy of rainfall forecast is vital, especially for tropical countries due to the complexity of weather patterns near the equator. In this study, we explore the application of neural networks techniques to predict precipitation distribution in Penang Island, Malaysia. The temporal intricacies and unique segmentation within time-series data make machine-learning approaches such as Long Short-Term Memory (LSTM) stand out among other methods. The research attempts to evaluate the performance of various neural networks techniques focusing on traditional non-LSTM and variation of LSTM methods including Vanilla LSTM, Bidirectional LSTM, Stacked LSTM, CNN-LSTM, RNN-LSTM, Attention-LSTM and Transformer-LSTM in predicting univariate rainfall data aggregated by month. The results reveal that traditional methods such as RNN and GRU performed better across all metrics compared to more complex architecture of hybridized LSTM. Vanilla LSTM showed a stable results proving its capability as a strong baseline model. The research emphasizes the importance of model generalization and model selection suitable that align with data characteristics, particularly in improving decision-making links to agricultural planning and disaster mitigation.**

***Index Terms*—Rainfall, Prediction, Neural Network, Time-Series, LSTM.**

I. Introduction

THE study on climate has been around for years. Climate change, such as the shift in temperature and weather patterns, continues to increase day by day. For countries near the equatorial region, phenomena such as El Niño intensify prolonged hot and dry weather. These extreme weather conditions are associated with higher temperatures, less precipitation, an increase in the risk of fires, haze, and drought [1]. This condition has caused the water level for domestic and agricultural use to reach a critical level, dropping drastically from its normal level and leading to a severe water crisis. The severe impact is visible in the reduction in crop production due to the significant loss of water supply for farm irrigation [2]. The Malaysia Meteorological Department (MetMalaysia) reported that rainfall in Peninsular Malaysia will be below average even during the monsoon season in 2024 due to the El Niño phenomenon [3].

Precipitation is a vital element of the hydrological cycle as its profound impacts numerous spheres of human life. Precise rainfall forecasting is indispensable for systematic allocation of water resources in domestic, agricultural and industrial settings. The distribution of rainfall may vary depending on the season, annual phenomena and the location around the globe. Thus, developing predictive models plays an important role in interpreting the complex elements in climate circulation. Precipitation forecasting involves various techniques ranging from the traditional statistical approach to advanced machine learning and deep learning methods. Numerical approaches such as regression algorithms combined with meteorological data are commonly utilized in simulating and predicting weather patterns. However, the non-linear nature and irregular patterns of time-series data limit the ability of the statistical model to achieve prediction with higher precision [4][5].

On the other hand, time series analysis offers a compatible approach by leveraging historical rainfall data to identify patterns and trends. Time series prediction analyses historical data in chronological order to estimate the underlying temporal data processes and predict future data characteristics. Theoretically, this analysis method can be defined as the sequence of vectors, $x(t),\ t = 0, 1, \ldots, n$ where the values x deviates continuously within timesteps (denoted as t), for any given system [6]. Among the various time series techniques, neural network models have demonstrated remarkable performance in predicting the values of up to the present time highlighting its capabilities in handling sequential data and capturing long-term dependencies [6].

Other key traits of time series analysis compared to the other existing approaches are spatiotemporal dependencies, probabilistic model and discrete output distribution [7]. The heavy reliance on spatiotemporal features such as the location conditions in rainfall forecasting is very challenging due to its high degree of uncertainty. Probabilistic models such as neural networks can capture the learning latent features and their dependencies which lead to more accurate predictions while producing a discrete output [7]. Discrete distribution refers predefined of categories in the prediction model instead of providing continuous numerical values.

The term neural network is coined by scientists and engineers who aim to synthesis artificial brain power by

Manuscript received December 31, 2024; revised August 25, 2025.

This work was supported by the Ministry of Higher Education Malaysia for the Transdisciplinary Research Grant Scheme (TRGS) with Project Code: TRGS/1/2022/USM/02/3/3.

Nor Shamira Sabri is a PhD candidate from School of Computer Sciences, Universiti Sains Malaysia (USM), 11800, Pulau Pinang, Malaysia. (Email: snshamira@student.usm.my).

Azleena Mohd Kassim is a Senior Lecturer from School of Computer Sciences, Universiti Sains Malaysia (USM), 11800, Pulau Pinang, Malaysia. (corresponding author – Email:azleena.mk@usm.my).

Noor Farizah Ibrahim is a Senior Lecturer from School of Computer Sciences, Universiti Sains Malaysia (USM), 11800, Pulau Pinang, Malaysia. (E-mail: nfarizah@usm.my).

Mohd Heikal Husin is a Senior Lecturer from School of Computer Sciences, Universiti Sains Malaysia (USM), 11800, Pulau Pinang, Malaysia. (E-mail: heikal@usm.my).

Mohd Shareduwan Mohd Kasihmuddin is an Associate Professor from School of Mathematical Science, Universiti Sains Malaysia (USM), 11800, Pulau Pinang, Malaysia. (e-mail: shareduwan@usm.my).

Ismail Ahmad Abir is a Senior Lecturer from School of Physics, Universiti Sains Malaysia (USM), 11800, Pulau Pinang, Malaysia. (E-mail: iahmad-abir@usm.my).

mimicking the biological brain [8]. The pattern recognition, data analysis, and control provided by this technique are very notable in solving practical problems [9]. The neural network approach serves as the foundation for the LSTM network and its variations, enabling the processing of sequential data and time series forecasting. The traditional approach of neural networks often struggle with vanishing problems when processing data with sequential dependencies. LSTMs address this shortcoming by introducing memory cells and gating mechanisms that moderate the flow of data by allowing the retention of temporal dependencies over a prolonged sequence [10]. Variation of LSTMs, such as stacked LSTM, Bidirectional LSTM (BiLSTM) and Attention-based LSTMs, further enhanced the performance of handling complex sequential data by achieving higher accuracy in applications such as rainfall prediction, speech recognition, and financial analysis.

II. Related Work

In predicting rainfall, extensive data sets and expert knowledge are needed, especially in calibrating and validating models. The advancement of prediction models utilizing LSTM methods shows promising results. A study based in Bangladesh uses LSTM approach to predict the rainfall while accounting the butterfly effect -also known as chaos- together with weighted features [11]. The studies show prominent results when compared to traditional machine learning methods like K-Nearest Neighbor (KNN), Logistic Regression, Support Vector Machine (SVM), Random Forest, Naïve-bayes and traditional neural network. The designed two layers of the LSTM with Principal Component Analysis (PCA) model able to achieve 97.14% in accuracy while neural network and K-Nearest Neighbor (KNN) achieved 76.9% and 76.8% in accuracy respectively. Endalie at al. [12] proposed LSTM-based rainfall prediction model in Ethiopia. The study shows that the proposed method outperformed other approaches used such as KNN, SVM, Decision Tree (DT) and Multi-Layer Perceptron (MLP) in estimating rainfall in 60 days. The LSTM-based networks achieved 99.72% in accuracy and outperformed other machine learning methods in RMSE with 0.010 while KNN, SVM, DT, and MLP scores reduced by 4.5%, 3.6%, 7.4% and 2% respectively.

Ebtehaj and Bonakdari [13] compare the performance of CNN and LSTM in predicting hourly rainfall intensity based in Canada. The prediction measures the level of precipitation in from one to six hours and classifies the intensities as slight, moderate, heavy and very heavy. The outcome of the study shows that LSTM performed better than CNN for minor and high-intensity events, whereas CNN showed superior performance in major precipitation events with shorter lead time. The study highlights the effectiveness of LSTM performance in long-term forecast while CNN excels in short-term predictions.

A study using the hybrid algorithm of CNN-LSTM to predict rainfall for crop harvesting also highlights the practicality of neural networks variation [14]. The study compared the performance of CNN-LSTM and other existing machine learning approaches such as Linear Regression, Logistics Regression and KKN. The outcomes illustrate that the proposed method of CNN-LSTM with First order optimization algorithms outperformed the traditional methods across different metrics. Aderyani et al. [15] performed a prediction on short-term precipitation to test the efficiency of three different methods namely particle swamp optimization-support vector machine (PSO-SVM), LSTM and CNN. The experiment ensuing that PSO-SVM and LSTM are more suitable in predicting rainfall compared to CNN, especially in 15 and 5 minutes ahead rainfall depth forecast respectively. The research suggests that the result may change depending on the forecast's model lead time.

A comparative analysis was also conducted using machine learning algorithms on time-series rainfall data [16]. Five different methods were used in the analysis which are LSTM, Stacked LSTM, Bidirectional-LSTM, XGBoost, and the ensemble of Gradient boosting Regressor, Linear Support Regression and Extra-trees Regressor to predict hourly rainfall volume. The result shows that Bidirectional-LSTM and Stacked-LSTM performed neck and neck when predicting the data. The study notes that LSTM networks with fewer hidden layers performed better compared to other methods tested in the similar experimental settings. Tao et al. [17] proposed to improve the accuracy of rainfall prediction by experimenting with multiscale LSTM. The research compared the performance of multiscale LSTM and multiscale LSTM with attention mechanism to forecast precipitation in monthly duration. The study shows that multiscale LSTM and multiscale LSTM with attention mechanism improve the accuracy of prediction significantly compared to using Vanilla LSTM. The addition of attention mechanism to multiscale LSTM algorithms provides the best prediction performance among all the other models.

A study tested the performance of data-driven machine learning approaches in predicting rainfall by feeding 41 years of monthly precipitation data into the prediction model [18]. The research compares the performance of Transformer-based Deep Learning algorithm, RNN, LSTM and Gated Recurrent Unit (GRU) in forecasting the long sequence of data. Result shows that Transformer-based architecture performed the best with minor prediction errors when tested by using Diebold-Mariano test. This study highlights the superiority of transformer models in processing data parallelly while demonstrating the faster and efficient large datasets handling. A study on the impact of temperature variation on rainfall patterns utilizing five machine- learning highlights the capability of LSTM architecture [6]. This study considers multivariate input data such as rainfall, temperature, humidity, surface pressure, wind direction and wind speed. Among other algorithms, Recurrent Neural Network (RNN) with LSTM demonstrates great capabilities in prediction studies. The previous study shows that LSTM algorithms have the potential to produce great accuracy models, especially in predicting rainfall. This study aims to compare the LSTM-based architecture to uncover which approaches best suit the nature of time-series prediction of rainfall.

III. Methodology

The most common neural network model used for prediction is the Recurrent Neural Network (RNN) model. RNN is widely used in research of time-series data as the cells of RNN can capture the dependencies that exist in the data element which are stored in succession by integrating hidden states while maintaining the relationship between

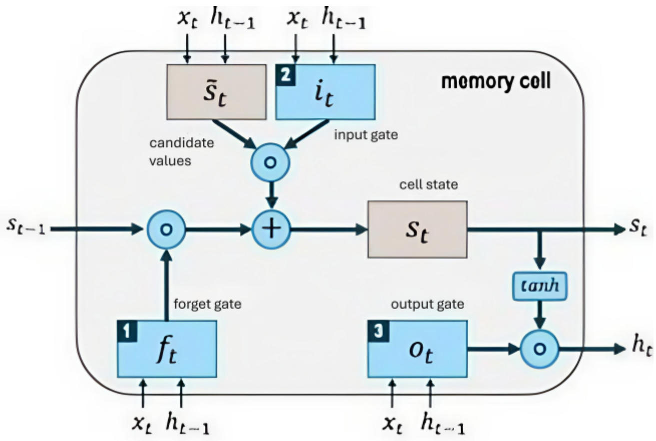

Figure 1: A Simple LSTM Architecture

previous and current data [18]. However, RNN architecture has its limitations, often known as the gradient vanishing and exploding problem, which makes this method unable to learn and make long-term predictions [19]. Improved versions of RNN models, the Long Short-Term Memory (LSTM) was introduced to mainly overcome these issues [20].

A. Long Short-Term Memory (LSTM)

LSTM is a modified version of RNN that specializes in sequential learning tasks [21] such as speech recognition and weather prediction. LSTM can learn long-term dependencies and overcome the gradient vanishing and exploding problem due to the long series of training processes [22]. LSTM is made up of three layers: the input layer, the hidden layer(s) and the output layer. In the hidden layer or memory cell, four different neural network layers can be found which include the cell state, input gate, forget gate and output gate. The forget gate functionality is to remove information from the cell state while the input gate is specified for adding information to the cell state. The information in the cell state is then used as the output of this approach [23]. The changes in the cell state vector circulated iteratively to capture the long-term dependencies which RNN lacks [10].

The forget gate of LSTM is designed to be selective and only forget the information from the previous timestep, t-1 that is considered unimportant. The input gate stores new information in the cell state at the current time (denoted as t) while the output gate filters the information in the cell state and produces the model's output. Formulation of the forget gate, input gate and output gate are defined in Eq. (1), where is the weight and is the bias matrix.

$$\begin{aligned} \text{Forget gate,} \quad & f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] + b_f\right) \\ \text{Input gate,} \quad & i_t = \sigma\left(W_i \cdot [h_{t-1}, x_t] + b_i\right) \\ \text{Output gate,} \quad & o_t = \sigma\left(W_o \cdot [h_{t-1}, x_t] + b_o\right) \end{aligned} \tag{1}$$

B. Bidirectional Long Short-Term Memory (BiLSTM)

LSTM architecture processes data in a single forward direction, normally from the past to the future. In anticipating an upcoming element in the sequence, it can only access the data based on the previous context. Bidirectional LSTM (Bi-LSTM) is an improved version of the LSTM network that processes data bidirectionally, in forward and reverse routes. Bi-LSTM is programmed to precisely identify and extract more time dependencies compared to the unidirectional LSTM [24]. The structure of this architecture allows the information to flow forward to discover pattern variation in the system and flow backwards for smooth prediction [10]. This variation of LSTM can manage spatial, temporal and incomplete data due to its flexible connection in the cell state vector [24].

C. Stacked Long Short-Term Memory (Stacked LSTM)

The stacked LSTM approach is constructed using two or more LSTM structures built together as hidden layers. These stacked layers can provide better performance and represent an elevated level of time-series analysis compared to LSTM networks in some applications [27]. The structure of stacked LSTM is especially helpful for long-term sequence forecasting and feature extraction due to its complex structure allowing adequate representation of temporal dependencies and historical patterns in data [28]. The multiple stacked layers can be applied in financial forecasting, rainfall prediction and speech recognition [28].

D. Convolutional Neural Network Long Short-Term Memory (CNN-LSTM)

CNN-LSTM combines Convolutional Neural Network (CNN) and LSTM architecture. A traditional CNN algorithm is designed to process data in a structural grid like images [12]. A simple CNN structure comprises a convolutional layer, a pooling layer and fully connected layers [10]. The CNN model constructed multiple strainers capable of extracting hidden features by pooling input data and performing layer-by-layer convolution [26][4]. The merger of CNN and LSTM layers allows for accurate spatial and temporal prediction where the CNN part of the model extracts spatial features from datasets [28] and LSTM handles temporal dependencies.

E. Recurrent Neural Network Long Short-Term Memory (RNN-LSTM)

The RNN-LSTM approach is a type of neural network designed to process sequential data while capturing both short-term dependencies from RNN [17][29] and long-term dependencies of LSTM [22]. Conventional RNNs are often facing challenges of vanishing gradient which limit their capacity to retain extended sequence. The hybridization of this technique addresses the issue by ensuring precise control of information flow through the gated mechanism in LSTM (refer to Fig. 1).

F. Attention Long Short-Term Memory (Attention-LSTM)

The Attention-LSTM network is an integration of LSTM and attention mechanism which boosts the model's capacity to focus on the most relevant part of sequential data. LSTMs are excellence in capturing long-term dependences; however, they often struggle to retain the important information as the sequence gets longer [16]. The attention mechanism addresses the limitation by assigning varying importance

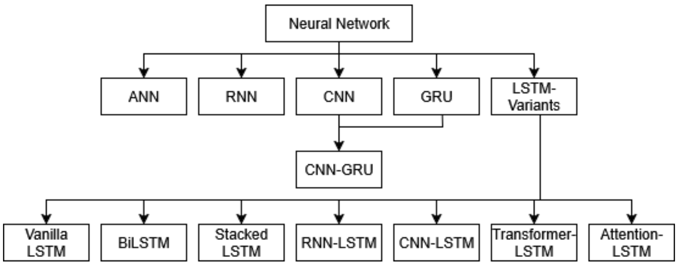

Figure 2: Neural network methods used in this study

weights to different timesteps, allowing the model to prioritize the most relevant inputs and reduce the influence of the less significant one [16][25]. This hybridization let LSTM tackle temporal dependencies while attention layer computes the weightage of the output. This architecture is applicable in stock market forecasting, rainfall prediction, and machine translation [16].

G. Transformer Long Short-Term Memory (Transformer-LSTM)

The Transformer-LSTM is a hybrid architecture that merges Transformer model with LSTM to enhance the processing of sequential data. While LSTM are excelled at capturing long-term dependencies, handling a very long sequence can be a challenge due to limitation in memory retention [17]. On the contrary, Transformer model utilizes self-attention mechanism allowing them to acquire efficient global dependencies without relying on the sequential computation [17]. The integration of both methods benefited from the self-attention in learning long-range dependencies by Transformer layers and processing sequential patterns offered by LSTM networks. The combination of approaches is especially practical in forecasting trends that emphasize both local and global dependencies.

IV. Dataset Description and Experimental Setting

The dataset used in this study is the precipitation-corrected daily data (mm/day) from January 1st, 1994, until December 31st, 2023, with 10,957 readings. The dataset downloaded was based on Modern-Era Retrospective analysis for Research and Applications, Version 2 (MERRA-2) dataset pinpointed on Fig 3 which is Penang Island, Malaysia (Latitude 5.353, Longitude 100.2702) at an average elevation point of 9.27 meters. This dataset can be accessed from NASA Prediction of Worldwide Energy Resources or the NASA POWER project (https://power.larc.nasa.gov/). The precipitation dataset is loaded into the Python environment where the data frame is converted into a time series object with a cycle set aggregated into monthly value, plotted in Fig 4.

Based on the dataset, the average reading of the rainfall is 6.459865 with a standard deviation of 8.377298. The minimum rainfall with the value of 0.00 indicates no rain while the maximum reading is at 132.35 dated 27th November 2023. The dataset revealed that the highest rainfall recorded is past the middle of the year, around June to August scattered throughout the 20 years. The record also uncovers that the dry season for this region starts from November up

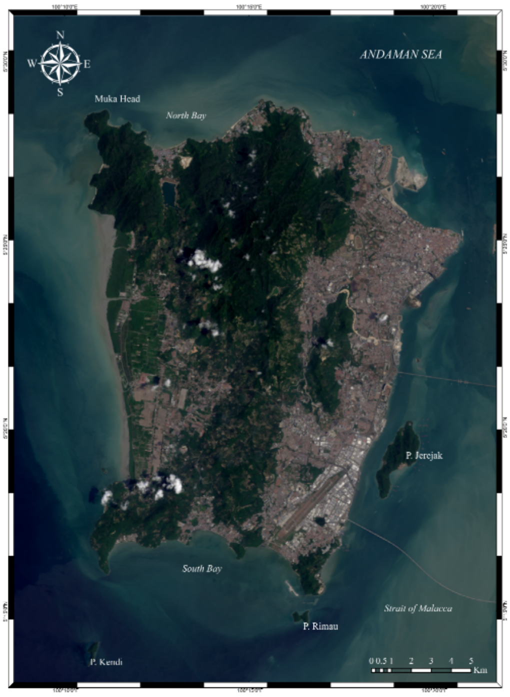

Figure 3: Penang Island, Malaysia

to March with occasional rainfall in between. The difference between the highest and the lowest recorded rainfall is very big due to seasonal occurrences such as the Southwest and Northeast monsoon. Even though the data shows fluctuation in the rainfall distribution, the overall reading shows that there is an increase in precipitation records throughout two decades.

Modelling the rainfall time series dataset is done in three stages: data preprocessing, model construction and model evaluation. In the stage before prediction, the data is split into two parts following the timeframe: 80% of the data is for the training set while the remaining 20% is for the testing dataset. The starting date is 1994-01-01, split on 2017-12-31 and ended on 2023-12-31 which makes the training set have 8766 readings while the testing set has 2192 readings. The training dataset was used to train and construct the model. The input data undergoes normalization process using MinMaxScalar before training in the models. In the second stage, all methods models implemented. LSTM-based deep learning method has a high reputation for achieving significant results in time series prediction compared to other approaches. Therefore, the Vanilla LSTM is chosen as the baseline performance comparison with the models.

The experimental setting for all the prediction method were performed with 3.30 GHz AMD Ryzen 5 5600H with Radeon Graphics processor, 8 GB DDR RAM, 500 GB of SSD and Windows 11. The prediction was implemented using Python 3.9.7, Tensorflow 2.16.1, Keras 3.0, and Scikit Learn library version 0.22.1.

The model parameters should be adjusted to fit the dataset used and accomplish accurate prediction. For simplicity purposes, the LSTM and Bi-LSTM model has only two hidden layers: the first layer with 100 nodes and the second layer with 50 nodes. The Stacked LSTM was assigned with

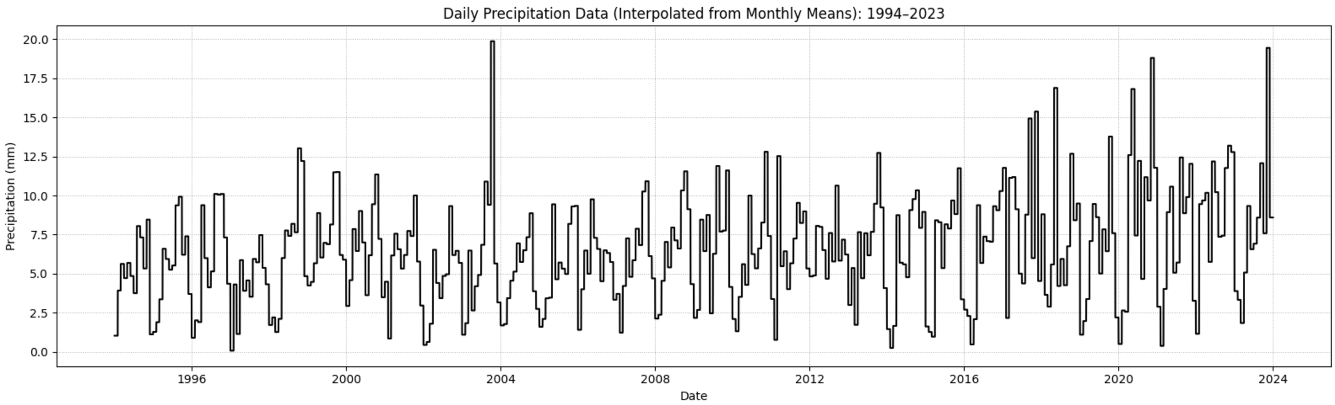

Figure 4: Monthly precipitation data from 1994-2023

an extra layer of LSTM with 25 nodes. The dropout was set to 0.2 for the first layer to prevent overfitting. CNN-LSTM consists of a convolutional layer with 64 filters and two kernel sizes, followed by max pooling layer size two. Then, the subsequent layers in this model combination are three LSTM layers like the base model. RNN-LSTM has four hidden layers with the first layer being on the recurrent part of the model. The SimpleRNN layer was tuned with 64 nodes while the second, third and fourth layers are the LSTM layers following similar settings to the base model. The Attention-LSTM and Transformer-LSTM have the same hidden units as the base model with the addition of three attention mechanisms and four heads respectively.

All models were trained with 100 epoch and 5 iterations with a default batch size of 32 using the adaptive movement estimation (Adam) optimizer. The models adopt MAE as the loss function while setting the sliding window size to 30 with a 0.001 learning rate. In ensuring smooth and non-linear activation, we utilized the hyperbolic tangent (tanh) function for all models. Finally, in the evaluation of the performance of the prediction models the following metrics: root mean square error (RMSE), mean square error (MSE), mean absolute error (MAE), the coefficient of determination (R^2) and mean absolute percentage error (MAPE).

MSE is a common loss function used in regression models including LSTM where it measures the difference between predictions and the actual values [31]. The loss error was also measured by MSE in this experiment. RMSE is a variation of MSE which is associated with the difference between model outputs and the observation [32]. MAE measures the average difference between the results and actual data while MAPE showcases the precision of prediction for any model type [32]. The coefficient of determination, measures the degrees of prediction models correspond to the observed values where the values are bounded by 1, indicating 1 is the perfect fit model. The values of MSE, MAE, RMSE and MAPE indicate that the smallest and closer the values to zero represent higher model accuracy. The equations are defined as follows:

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

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{3}$$

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{4}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{5}$$

$$\text{MAPE} = \frac{100\%}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \tag{6}$$

$$\text{Accuracy} = 100 - MAPE \tag{7}$$

where are the actual values, are the predicted values, indicates the mean of actual values and is number of observations. The experiment also measures the accuracy of prediction using Eq.(7). In this study, the baseline model chosen is the Vanilla LSTM due to its outstanding effectiveness in sequential data modeling. Several non-LSTM techniques will be incorporated to better understand the performance differences across different models types.

V. Result and Discussion

The experiment compared the results between all the forecasting approaches, namely Vanilla LSTM, BiLSTM, Stacked LSTM, CNN-LSTM, RNN-LSTM, Attention-LSTM, Transformer-LSTM with several non-LSTM architectures such as CNN-GRU, GRU, ANN, CNN and RNN to showcase their performance in forecasting rainfall. In this section, we will include the discussion of training loss analysis over epochs, training time comparison and the overall performance metrics. Fig. 3 illustrates the behavior of prediction models over 100 epochs where all approaches revealed a rapid decline in training loss in the initial stage of epochs. This patterns indicate the effective learning in the time series rainfall data. While most models show low and stable loss values over the time, CNN-LSTM demonstrate a poorer fit to the training data indicated by the higher training loss compared to the other method. Transformer-LSTM illustrate the highest initial loss but quickly match the other model. Despite all the different techniques used, majority of the models reach plateau before epoch 30 which

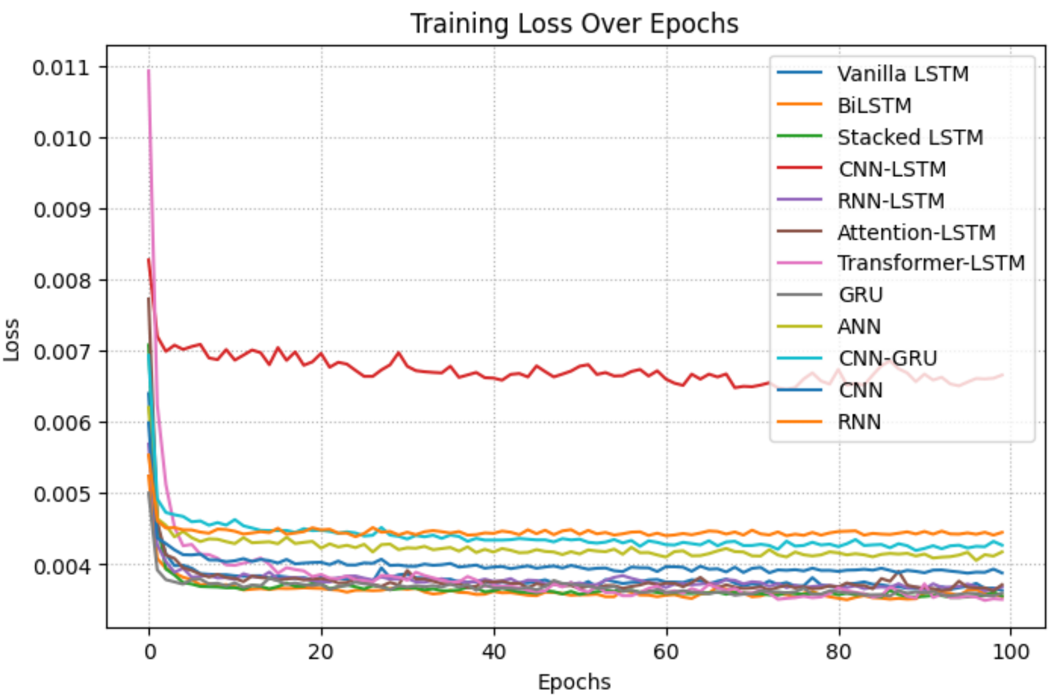

Figure 5: Training Loss

Table I: Training Time and Accuracy of Prediction Model

Neural Network	Accuracy	Training Time (sec)
Vanilla LSTM	94.02	447.24
BiLSTM	93.32	461.99
Stacked LSTM	93.55	519.59
CNN-LSTM	89.69	590.54
RNN-LSTM	93.33	539.25
Attention-LSTM	93.18	477.43
Transformer-LSTM	91.35	584.67
CNN-GRU	91.55	190.03
GRU	94.00	542.08
ANN	92.40	29.97
CNN	92.57	69.85
RNN	**94.43**	74.69

indicate most learning completed in the earlier stage of training.

The training time comparison shows the computational efficiency of the models used and it reflects the ability of the model to predict rainfall in real-time. The findings illustrate that Vanilla LSTM took 447.24 seconds and BiLSTM with 461.99 seconds, the most efficient among the other LSTM variations while maintaining strong accuracy with 94.02% and 93.32% respectively. The fastest training time among all the methods is by ANN with 29.97 seconds and 92.40% of accuracy. Despite ANN is not design specifically for sequential data, allowing the feedforward network to work on the recent historical values by transforming the the nature of data into a fixed-size input vector allow it to forecast the temporal dependencies [33].

GRU take the longest to train with 542.67 seconds compered to the other non-LSTM approach while achieving the third highest accuracy of 94% after RNN and Vanilla LSTM as the baseline model. CNN-LSTM performed poorly despite having the longest training time with accuracy of 89.69% implying that the convolutional layers were ineffective in capturing the precipitation patterns in this dataset. The combination of results from training time and accuracy suggest that RNN and Vanilla LSTM are the best models for deployment as both approaches provide highest prediction accuracy with only 0.38% difference. While the accuracy of RNN and Vanilla LSTM difference is not much, the time taken to train both models differs by almost 150% with RNN take a lot less time training the dataset. Ironically, the hybridization of both model, RNN-LSTM take longer to train but achieve lower accuracy in predicting rainfall. Based on Fig. 6 - Fig. 9, we can assume that the complex architecture does not guarantee better prediction accuracy.

The evaluation metrics were used to evaluate the prediction performance of all methods which include RMSE, MAE, R^2 and MAPE. MAPE is a percentage error, however in this study we plot it in decimal point to ease the comparison process as plotted in Fig. 9 and tabulated in Table II. Based on Table II, RNN outperformed other architecture with the lowest RMSE (0.9563) and MAE (0.2406), while achieving the highest R^2 score with 0.9504. ANN score the same R^2 value as RNN with less than 0.0001 difference in RMSE and MAE values 0.9564 and 0.9148 respectively. GRU also performed better by achieving higher R^2 value of 0.9426 and lower RMSE (1.0282) compared to Vanilla LSTM, however, Vanilla LSTM show better MAPE score of 59% while GRU score 60%.

Transformer-LSTM achieved the lowest performance in prediction process across all the metrics with the highest RMSE (1.3786), MAE (0.5374) and the lowest R^2 score of 0.8970. CNN-LSTM followed next as the second lowest results with RMSE of 1.3324, MAE of 0.5368, and R^2 0.9037 being the second lowest. However, in term of percentage error, MAPE, CNN-LSTM scores the highest with 0.1031 followed by Transformer-LSTM with 0.0845 and CNN-GRU with 0.0845. Interestingly, the complex hybrid models such as Transformer-LSTM and CNN-LSTM did

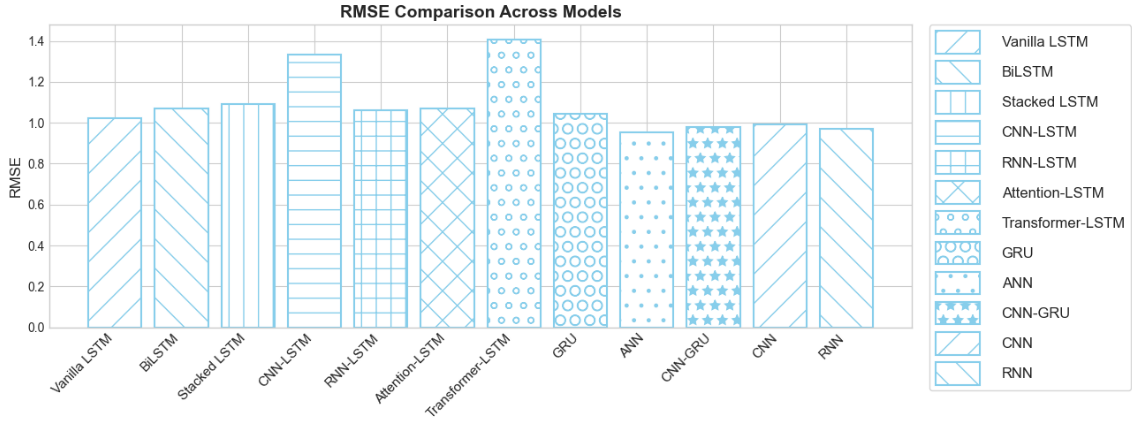

Figure 6: Comparison of RMSE across models

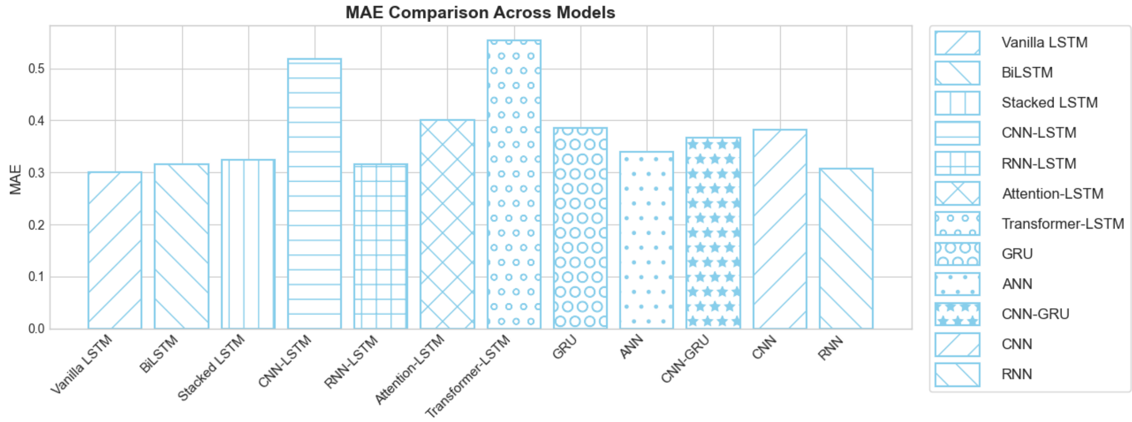

Figure 7: Comparison of MAE across models

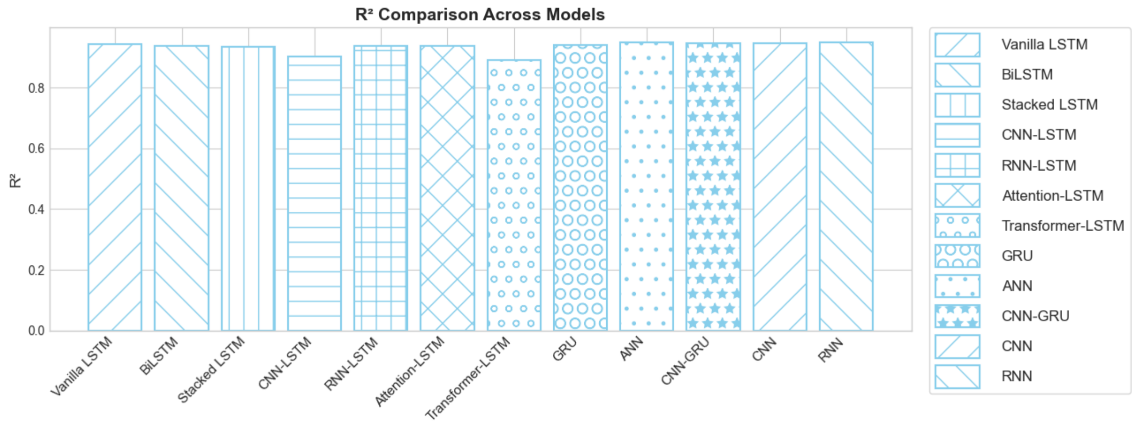

Figure 8: Comparison of R^2 across models

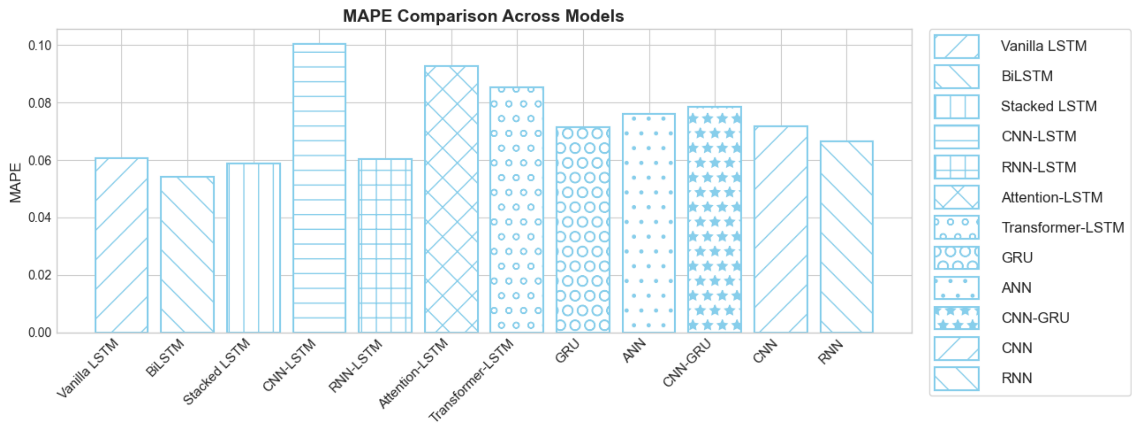

Figure 9: Comparison of MAPE across models

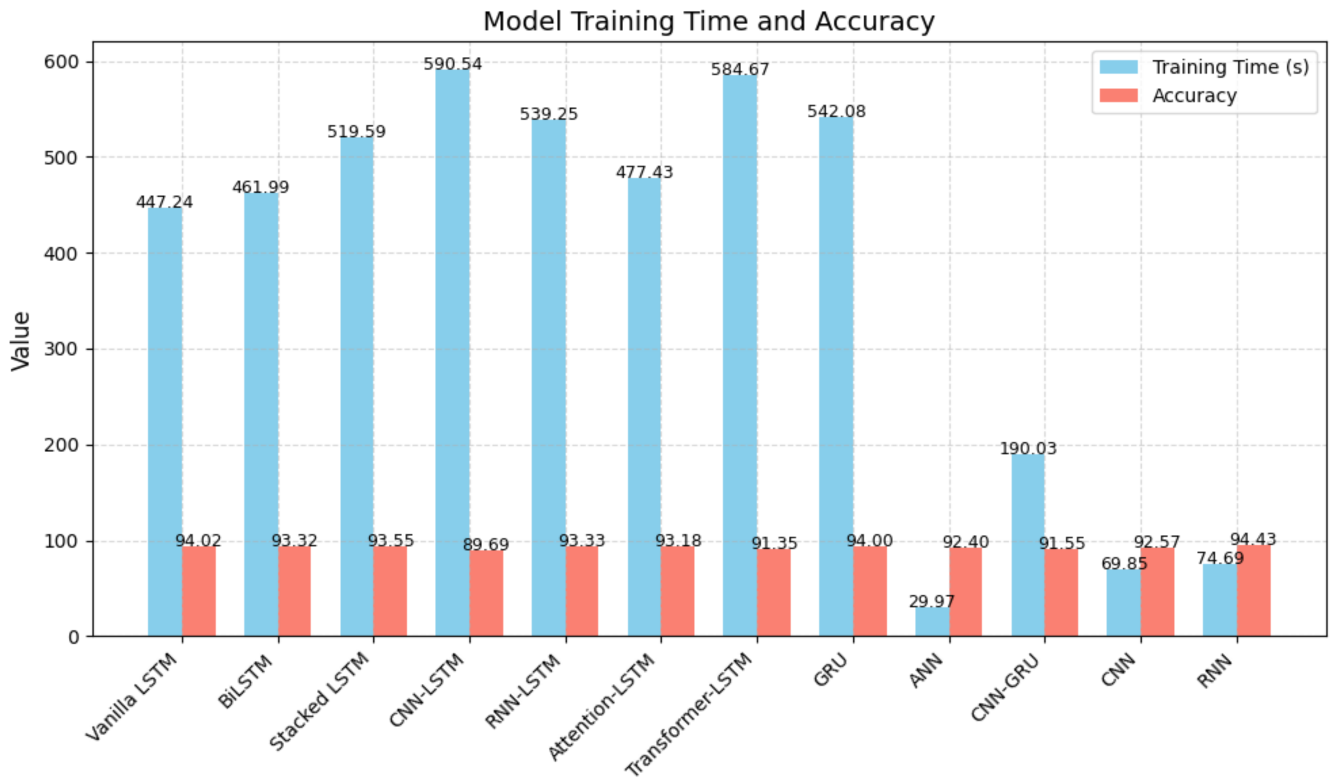

Figure 10: Training Time and Accuracy

not outperformed the basic LSTM variants which suggest that complexity of architectural structure does not always guarantee better results. This statement is supported by the performance accuracy of the models which illustrated in Fig. 10 despite taking longer time in training.

Table II: Evaluation metrics for all prediction model

Neural Network	RMSE	MSE	MAE	R^2	MAPE
Vanilla LSTM	1.0689	1.1530	0.3410	0.9375	0.0598
BiLSTM	1.0895	1.1942	0.3726	0.9352	0.0668
Stacked LSTM	1.1757	1.4469	0.3679	0.9216	0.0645
CNN-LSTM	1.3324	1.7764	0.5368	0.9037	0.1031
RNN-LSTM	1.1972	1.4678	0.3678	0.9205	0.0667
Attention-LSTM	1.1118	1.2509	0.3740	0.9322	0.0682
Transformer-LSTM	1.3786	1.9007	0.5374	0.8970	0.0865
CNN-GRU	1.0473	1.0984	0.3968	0.9405	0.0845
GRU	1.0282	1.0593	0.3331	0.9426	0.0600
ANN	0.9564	0.9148	0.3417	**0.9504**	0.0760
CNN	1.0104	1.0222	0.3926	0.9446	0.0743
RNN	**0.9563**	**0.9145**	**0.2406**	**0.9504**	**0.0557**

The best overall performing model is RNN attaining the lowest RMSE, MAE and MAPE which indicate that this model is the most consistent with least deviating predictions among sequence-based models. The highest accuracy achieved reflect to the ability to near-realistic prediction while taking the least time in model training. We can conclude that RNN is the most balanced model offering a solid prediction accuracy with low error while effective computation. ANN illustrate comparable results to RNN in term of training efficiency and higher MAE which suggest that this method has slightly less precision compared to RNN considering that it is not specifically design for time-series data. Next, GRU also achieve phenomenal results and served as a good alternative for LSTM-like temporal learning but with better convergence. In contrast, the variation of LSTM hybrid especially Transformer-LSTM and CNN-LSTM, which in theory are more powerful failed to showcase a good results due to complexity in architecture and overfitting. Conclusively, traditional method such as RNN and GRU are far more affective in predicting univariate prediction that utilizes sequence data.

The study also finds that the Vanilla LSTM serves as a strong and reliable baseline, show casting high prediction accuracy and relatively lower computational cost. While GRU and RNN performed better, other LSTM hybrid failed to outperformed Vanilla LSTM to process sequential patterns especially when compared in terms of error rates and computation runtime. BiLSTM and Attention-LSTM delivered comparable results to the baseline model by capturing the forward and backward temporal dependencies while reinforcing LSTM variations robustness. Future works may explore the integration of ensemble methods with LSTM variations, application to more complex sequential datasets and exploration of optimization methods tailored to specific techniques.

VI. Conclusion

In this study, we compare the performance of several neural networks models which include complex hybridization of LSTM architecture and non-LSTM models while the Vanilla LSTM model serves as the baseline model. The models computes rainfall dataset of 30 years that aggregated into monthly values. The experiments environment are the same across every models and evaluated using a few criteria. This experiment concludes that traditional methods such as RNN and GRU can outperformed the variation LSTM integration in predicting sequential patterns. CNN-LSTM and Transformer-LSTM show poor performance highlighting the importance of selecting architectures that align with nature of time-series data. Vanilla LSTM also showcase its potential as a strong benchmark for time-series prediction tasks. This finding may provide an insight into improving neural network-based forecasting model to ensure more reliable prediction systems.

References

[1] Chu Mei Mei, "Malaysia Forecasts Peak of El Niño to Hit in Early 2024," Reuters, December 4, 2004. https://www.reuters.com/world/asia-pacific/malaysia-forecasts-peak-el-nio-hit-early-2024-2023-07-07/

[2] Imran Hilmy, "Padi Farmers Struggling Amid Water Crisis in Perlis," The Star, December 4, 2024. https://www.thestar.com.my/news/nation/2024/08/19/perlis-padi-farmers-feel-the-heat

[3] Malaysian Metereological Department, MetMalaysia, "Weather Phenomena," MetMalaysia, December 4, 2024. https://www.met.gov.my/en/pendidikan/fenomena-cuaca

[4] Huang, Ruijie, Chenji Wei, Baohua Wang, Jian Yang, Xin Xu, Suwei Wu, and Suqi Huang, "Well Performance Prediction Based on Long Short-Term Memory (LSTM) Neural Network," Journal of Petroleum Science and Engineering, vol. 208, 2022.

[5] Nguyen, Huu Nam, Thuy-Anh Nguyen, Hai-Bang Ly, Van Quan Tran, Long Khanh Nguyen, Minh Viet Nguyen, and Canh Tung Ngo, "Prediction of Daily and Monthly Rainfall Using a Backpropagation Neural Network," Journal of Applied Science and Engineering, vol. 24, no. 3, pp367–379, 2021.

[6] Chicco, Davide, Matthijs J. Warrens, and Giuseppe Jurman, "The Coefficient of Determination R-squared is More Informative Than SMAPE, MAE, MAPE, MSE, and RMSE in Regression Analysis Evaluation," PeerJ Computer Science, vol. 7, ppe623, 2021.

[7] Harilal, Geethu Thottungal, Aniket Dixit, and Giovanni Quattrone, "Establishing Hybrid Deep Learning Models for Regional Daily Rainfall Time Series Forecasting in the United Kingdom," Engineering Application of Artificial Intelligence, vol. 133, 2024.

[8] Frank, Ray J., Neil Davey, and Stephen P. Hunt, "Time Series Prediction and Neural Networks," Journal of Intelligent and Robotic Systems, vol. 31, pp91-103, 2001.

[9] Jansson, Peter A., "Neural Networks: An Overview," Analytical Chemistry, vol. 63, no. 6, pp357A-362A, 1991.

[10] Bishop, Chris M., "Neural Networks and Their Applications," Review of Scientific Instruments, vol 65, no. 6, pp1803-1832, 1994.

[11] Fischer, Thomas, and Christopher Krauss, "Deep Learning with Long Short-Term Memory Networks for Financial Market Predictions," European Journal of Operational Research, vol. 270, no. 2, pp654-669, 2018.

[12] Endalie, Demeke, Getamesay Haile, and Wondmagegn Taye, "Deep Learning Model for Daily Rainfall Prediction: Case Study of Jimma, Ethiopia," Water Supply, vol. 22, no. 3, pp. 3448–3461, 2021.

[13] Ebtehaj, Isa, and Hossein Bonakdari, "CNN vs. LSTM: A Comparative Study of Hourly Precipitation Intensity Prediction as a Key Factor in Flood Forecasting Frameworks," Atmosphere, vol. 15, no. 9, pp1082, 2024.

[14] Salma, Subia, and A. Ashwitha, "Hybrid CNN-LSTM model: Rainfall Analysis and Prediction for Karnataka Region," Journal of Theoretical and Applied Information Technology, vol. 100, no. 22, pp6715–6716, 2022.

[15] Aderyani, Fatemeh Rezaei, S. Jamshid Mousavi, and Fatemeh Jafari, "Short-Term Rainfall Forecasting Using Machine Learning-Based Approaches of PSO-SVR, LSTM, and CNN," Journal of Hydrology, vol. 614, pp128463, 2022.

[16] Barrera-Animas, Ari Yair, Lukumon O. Oyedele, Muhammad Bilal, Taofeek Dolapo Akinosho, Juan Manuel Davila Delgado, and Lukman Adewale Akanbi, "Rainfall Prediction: A Comparative Analysis of Modern Machine Learning Algorithms for Time-Series Forecasting," Machine Learning with Applications, vol. 7, pp100204, 2022.

[17] Tao, Lizhi, Xinguang He, Jiajia Li, and Dong Yang, "A Multiscale Long Short-Term Memory Model with Attention Mechanism for Improving Monthly Precipitation Prediction," Journal of Hydrology, vol. 602, pp126815, 2021.

[18] Nayak, GH Harish, Wasi Alam, K. N. Singh, G. Avinash, Mrinmoy Ray, and Rajeev Ranjan Kumar, "Modelling Monthly Rainfall of India Through Transformer-Based Deep Learning Architecture," Modeling Earth Systems and Environment, vol. 10, no. 3, pp3119–3136, 2024.

[19] Billah, Mustain, Md Nasim Adnan, Mostafijur Rahman Akhond, Romana Rahman Ema, Md Alam Hossain, and Syed Md. Galib, "Rainfall Prediction System for Bangladesh Using Long Short-Term Memory," Open Computer Science, vol. 12, no. 1, pp323–331, 2022.

[20] Zhang, Jianjing, Peng Wang, Ruqiang Yan, and Robert X. Gao, "Long Short-Term Memory for Machine Remaining Life Prediction," Journal of Manufacturing Systems, vol. 48, pp78–86, 2018, doi: 10.1016/j.jmsy.2018.05.011.

[21] Graves, Alex, "Generating Sequences With Recurrent Neural Networks," 2013, http://arxiv.org/abs/1308.0850.

[22] Schmidhuber, Jürgen, "Tutorial on LSTM Recurrent Nets," 2003, www.idsia.ch

[23] Sherstinsky, Alex, "Fundamentals of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) Network," Physica D, vol. 404, 2020.

[24] Abduljabbar, Rusul L., Hussein Dia, and Pei-Wei Tsai,

"Unidirectional and Bidirectional LSTM Models for Short-Term Traffic Prediction," Journal of Advanced Transportation, vol. 2021, no. 1, pp5589075, 2021.

[25] Bin Zhang, Dan Yang, and Zhihuang Lin, "Medicine and Disease Association Prediction via Attention-Based Medical Heterogeneous Information Network Representation Learning," IAENG International Journal of Computer Science, vol. 49, no. 1, pp69-78, 2022.

[26] Xin Gao, Chen Xue, Wenqiang Jiang, and Bao Liu, "Attention-GRU Based Intelligent Prediction of NOx Emissions for the Thermal Power Plants," IAENG International Journal of Computer Science, vol. 51, no. 8, pp1171-1181, 2024.

[27] Sinha, Ujjwal, Vishal Thakur, Sammed Jain, M. Parimala, and S. Kaspar, "Rainfall Prediction Using Ensembled-LSTM and Dense Networks," International Journal of Engineering Systems Modelling and Simulation, vol. 14, no. 2, pp59–70, 2023.

[28] Cui, Zhiyong, Ruimin Ke, Ziyuan Pu, and Yinhai Wang, "Deep Bidirectional and Unidirectional LSTM Recurrent Neural Network for Network-Wide Traffic Speed Prediction," arXiv preprint arXiv:1801.02143, 2018.

[29] Karevan, Zahra, and Johan AK Suykens, "Spatio-Temporal Stacked LSTM for Temperature Prediction in Weather Forecasting," arXiv preprint arXiv:1811.06341, 2018.

[30] Lu, Wenjie, Jiazheng Li, Yifan Li, Aijun Sun, and Jingyang Wang, "A CNN-LSTM-Based Model to Forecast Stock Prices," Complexity, vol. 2020, 2020.

[31] Karpathy, Andrej, Justin Johnson, and Li Fei-Fei, "Visualizing and Understanding Recurrent Networks," 2015, http://arxiv.org/abs/1506.02078

[32] Willmott, Cort J., and Kenji Matsuura, "Advantages of The Mean Absolute Error (MAE) Over The Root Mean Square Error (RMSE) in Assessing Average Model Performance," Climate Research, vol. 30, no. 1, pp79–82, 2005.

[33] Tealab, Ahmed, Hesham Hefny, and Amr Badr, "Forecasting of Nonlinear Time Series using ANN," Future Computing and Informatics Journal, vol. 2, no. 1, pp9–47, 2017.