

Precipitation Nowcasting in Dar es Salaam: Comparative Analysis of LSTM and Bidirectional LSTM for Enhancing Early Warning Systems

Innocent J. Junior^{1*}, Jacqueline Benjamin Tukay², Abraham Okrah³, Genesis Magara³, Daniel J. Masunga²

¹Department of Meteorology, School of Atmospheric Science, Nanjing University of Information Science and Technology, Nanjing, China

²Tanzania Meteorological Authority, Dodoma, Tanzania

³School of Ecology and Applied Meteorology, Nanjing University of Information Science and Technology, Nanjing, China

Email: *innocent.juniour@nuist.edu.cn

How to cite this paper: Junior, I. J., Tukay, J. B., Okrah, A., Magara, G., & Masunga, D. J. (2025). Precipitation Nowcasting in Dar es Salaam: Comparative Analysis of LSTM and Bidirectional LSTM for Enhancing Early Warning Systems. *Journal of Geoscience and Environment Protection*, 13, 327-342. <https://doi.org/10.4236/gep.2025.134018>

Received: March 25, 2025

Accepted: April 19, 2025

Published: April 22, 2025

Copyright © 2025 by author(s) and Scientific Research Publishing Inc. This work is licensed under the Creative Commons Attribution International License (CC BY 4.0). <http://creativecommons.org/licenses/by/4.0/>



Open Access

Abstract

Accurate precipitation forecasting is crucial for mitigating the impacts of extreme weather events and enhancing disaster preparedness. This study evaluates the performance of Long Short-Term Memory and Bidirectional LSTM models in predicting hourly precipitation in Dar es Salaam using a multivariate time-series approach. The dataset consists of temperature, pressure, U-wind, V-wind, and precipitation, preprocessed to handle missing values and normalized to improve model performance. Performance metrics indicate that BiLSTM outperforms LSTM, achieving lower Mean Absolute Error and Root Mean Squared Error by 6.4% and 6.5%, respectively along with improved threshold scores. It demonstrated better overall prediction accuracy. It also improves moderate precipitation detection (TS3.0) by 16.9% compared to LSTM. These results highlight the advantage of bidirectional processing in capturing complex atmospheric patterns, making BiLSTM a more effective approach for precipitation forecasting. The findings contribute to the development of improved deep learning models for early warning systems and climate risk management.

Keywords

Precipitation Prediction, Long Short-Term Memory, Bidirectional LSTM, Dar es Salaam

1. Introduction

Precipitation forecasting is a very important key in water resource management, agricultural planning, and disaster preparedness (Moeletsi et al., 2013; Ali et al., 2020; Piran et al., 2024). Accurate predictions help to mitigate the adverse effects of extreme weather conditions such as floods and drought which directly impact the life of the people and economy of the country. Rainfall forecasts are typically issued in different timescales, including nowcasting (0 to 6 hrs), short-range forecast (6 hrs to 3 days), medium-range forecast (4 to 10 days) and long-range forecast (10 days to several months) (Mason, 2016). Rainfall prediction relies on data collected from multiple sources, including ground-based observation stations, radar systems, satellite imagery, and radiosonde measurements. These data are analyzed collectively using different tools such as numerical weather prediction models to generate reliable forecasts (Wu & Xue, 2024). Short-term rainfall forecasting remains a significant operational challenge due to the highly dynamic and nonlinear nature of precipitation systems. Rainfall events have rapid spatiotemporal variability, with localized convective processes capable of intensifying from negligible to extreme rates within minutes (Barros & Lettenmaier, 1994).

Dar es Salaam is among the cities close to the equatorial coast with over 5 million people (NBS, 2022), which faces serious flooding problems due to heavy rain seasons. It experiences bimodal rainfall patterns, long rains from March to May (MAM) and short rains from October to December (OND) (Owiti, 2012). The rain season is primarily influenced by the Inter-tropical convergence zone (ITCZ) and other contributing factors such as monsoon winds prevailing circulation patterns and cyclones (Kai, Ngwali, & Faki, 2021). The rain seasons ensure water supplies but sometimes they flood lowlands like the Msimbazi River basin (Sakijege, Lupala, & Sheuya, 2012). The Msimbazi Basin originates from the Kisarawe Highlands, whose topography influences local weather patterns by obstructing coastal winds and sometimes causing unseasonal rainfall. These devastating floods over the basin bring loss of lives, destruction of infrastructure, and significant economic setbacks due to trade disruptions and damage to key revenue-generating sectors (Jerome Glago, 2021). Accurate and reliable rainfall prediction systems are vital to provide early warnings, enhance disaster preparedness, and mitigate extreme weather events. Tanzania Meteorological Authority (TMA) uses traditional forecasting methods together with NWP models, which rely on physical equations to simulate atmospheric processes, and have long been the standard for precipitation forecasting. However, these models often require extensive computational resources and struggle with short-term rainfall prediction, especially in tropical climates (Hess & Boers, 2022). Recent studies (Waqas & Humphries, 2024; Ebtahaj & Bonakdari, 2024) have demonstrated that deep learning models, particularly LSTM-based architectures, can provide more accurate and computationally efficient nowcasting solutions.

In recent years, deep learning techniques have gained popularity, and researchers have conducted experiments on different models due to their ability to learn complex temporal dependencies in weather data. One of the advances is in Recurrent

Neural Networks (RNNs). These are a class of neural networks designed to process sequential data (Lipton, 2015). Unlike traditional feedforward networks, RNNs have recurrent connections that allow them to maintain a memory of past inputs. This memory enables RNNs to capture temporal dependencies and learn patterns that evolve over time (Elman, 1990). However, basic RNNs suffer from the vanishing gradient problem, which makes it difficult to train them on long sequences (Bengio, Simard, & Frasconi, 1994). Hochreiter & Schmidhuber (1997) addresses the vanishing gradient problem through a special type of RNN called Long Short-Term Memory (LSTM) networks. LSTMs have a unique gate structure that regulates the flow of information into and out of the memory cell (Gers, Schmidhuber, & Cummins, 2000; Graves, 2012). These gates allow LSTMs to selectively remember or forget information, enabling them to capture long-range dependencies in time series data. LSTMs have been successfully applied to precipitation prediction, demonstrating their ability to learn complex patterns and make accurate forecasts (Priatna & Djamal, 2020; Xu et al., 2022). One of the key advantages of LSTM is its ability to retain long-term dependencies, making it particularly useful for rainfall forecasting (Kratzert et al., 2018). However, even though LSTM has its limit, it can only look at data in one direction (Sherstinsky, 2020) and thus can miss patterns depending on the past and the future. To overcome this limitation Bidirectional LSTM (BiLSTM) was developed. BiLSTM networks are an extension of LSTMs that process the input sequence in both forward and backward directions (Schuster & Paliwal, 1997) This allows the BiLSTM to capture both past and future dependencies in the data, providing a more complete context for prediction. In precipitation prediction, BiLSTMs can capture the influence of both preceding and subsequent weather events, leading to improved accuracy (Zhang et al., 2023). The ability to consider both past and future context makes BiLSTMs particularly well-suited for precipitation prediction tasks where weather patterns are influenced by a complex integrated factor. In this paper, BiLSTM will be used to improve precipitation accuracy over Dar es Salaam leading to better preparedness and early warning systems. In Dar es Salaam, these strategies can help capture the intricate relationships between meteorological factors driving rainfall.

2. Methodology

2.1. Data Set Description

This study utilized the ERA5 hourly dataset from the Copernicus Climate Data Store (CDS) (<https://cds.climate.copernicus.eu/datasets>). The dataset provides high-resolution meteorological data, including total precipitation, temperature, humidity, wind speed, and atmospheric pressure, with a spatial resolution of $0.25^\circ \times 0.25^\circ$ and a temporal resolution of 1 hour. Data formats include GRIB2 and NetCDF (Hersbach et al., 2020).

Hourly precipitation data for Dar es Salaam from 2010 to 2023 were extracted for model training. To enhance the model's predictive capability, additional variables including temperature, Uwind and Vwind components, and surface pressure were used as input features.

2.2. Study Area

The study focuses on Dar es Salaam as in **Figure 1**, a region characterized by a subtropical monsoon climate and ITCZ with significant seasonal variations in precipitation. It lies within latitude 6.6° S - 7.1° S and longitude 39.1° E - 39.5° E. The region experiences two distinct rainfall seasons: the OND and MAM.

2.3. Model Architecture and Development

LSTM

A special cell structure is used to replace the original hidden neurons in the LSTM. An illustration of the LSTM is shown in **Figure 2**, where all the solid arrows mean that the connection weight is 1. C_t is a memory cell, which is a linear element used to store information to guarantee that information can be stored for a long time to retain the correlation among elements in a sequence. g_t is the input node, which denotes the comprehensive interaction of the input at the time step t and the information of the previous network status. Its value can be passed on to a memory cell through the control of an input gate i_t . If W is the weight, b is the threshold (bias), i_t is the input gate, which receives the input at time step t and the network status information at previous time steps and passes the input value of node g_t into a memory cell C_t after the control of the sigmoid function. f_t is the forget gate, which determines whether the value of C_t is stored or not: if the weight is 1, it is stored as it was, and if it is 0, it is cleared. O_t is the output gate, which receives the input at the time step t and the network status information at previous

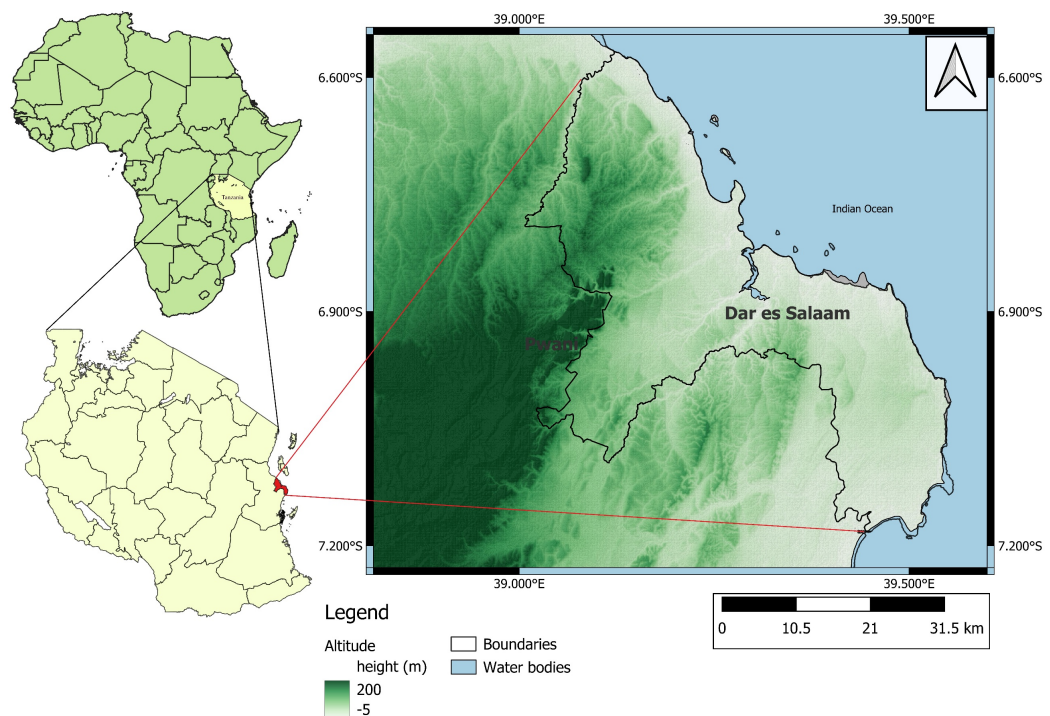


Figure 1. The Map of Dar es Salaam showing altitude (m).

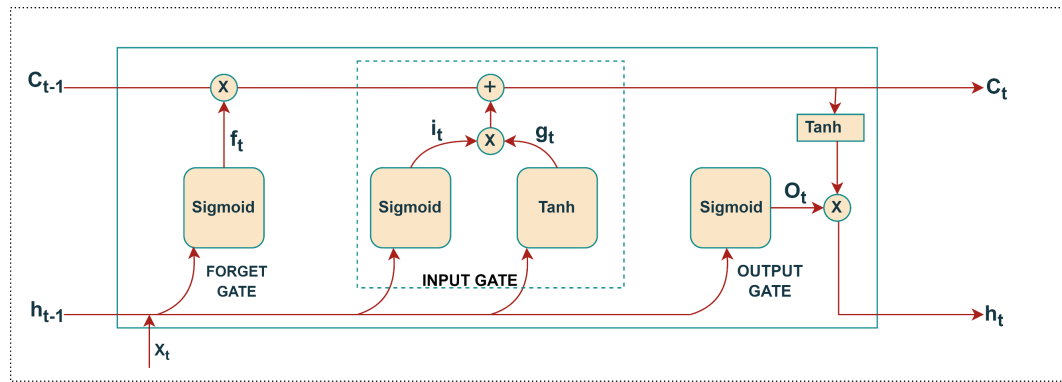


Figure 2. A memory cell of the LSTM.

time step. It controls the output of C_t after the sigmoid function. h_t is the output value. Where X and h vectors to denote the values of each layer.

Forget Gate. The forget gate decides which information from both previous and current sequence values should be retained or discarded in the network. The past hidden state [h_{t-1}], past cell state [c_{t-1}], and present value [x_t] are passed to this gate. They are combined [$x_t + h_{t-1}$] and duplicated. One copy is sent to the input gate. One copy is directed to the input gate, while the other is passed through the sigmoid activation function to update the cell's memory. This updated cell state [c'_{t-1}] is then also sent to the input gate.

Equation in the cell:

$$f_t = \sigma(W_{fx} \cdot x_t + W_{fh} \cdot h_{t-1} + b_f),$$

$$C'_{t-1} = f_t \cdot C_{t-1}$$

Input Gate. The input gate determines useful information from the current element of the sequence. The input to this cell is the updated cell state [c_{t-1}] and the combined [$x_t + h_{t-1}$] from the forget gate. [$x_t + h_{t-1}$] is duplicated again, and one copy is sent through the sigmoid activation function. The other copy is passed through the hyperbolic tangent function (tanh) and merged with the cell state to compute the current cell state [c_t], which is then sent to the output gate along with [$x_t + h_{t-1}$].

Equation in the cell:

$$g_t = \tanh(W_{gx} \cdot x_t + W_{gh} \cdot h_{t-1} + b_g),$$

$$i_t = \sigma(W_{ix} \cdot x_t + W_{ih} \cdot h_{t-1} + b_i),$$

$$C_t = g_t \cdot i_t + C'_{t-1}$$

Output Gate. The past information in the network's memory, i.e., the cell state [c_t], is used to process the current element of the sequence. It is passed through a hyperbolic tangent function (tanh) and combined with [$x_t + h_{t-1}$] which was passed through the sigmoid activation function to generate the final cell state [c_t] and hidden state [h_t]. This can then be forwarded to the output layer or to the next LSTM neuron for processing the next element in the sequence.

Equation in the cell:

$$O_i = \sigma(W_{ox} \cdot x_i + W_{oh} \cdot h_{i-1} + b_o),$$

$$h_i = \tanh(C_i) \cdot O_i$$

Dropout Layer is a regularization technique that helps prevent overfitting by randomly dropping out some LSTM units from our network during training. It reduces the coadaptation of neurons, which can lead to overfitting. By randomly dropping units in the layer, the network is encouraged to learn more robust representations that are less reliant on the specific training data, improving the model's ability to generalize. This layer is inserted immediately after the LSTM1 layer when Batch normalization has already been done and input it to LSTM layer 2. The final model is described in **Figure 3**.

LSTM MODEL Process

The spitted preprocessing data of Temperature, pressure, Uwind, Vwind and precipitation are passed in the model. The sequence is then fed to an LSTM layer with 64 units, which learns patterns over time while maintaining long-term dependencies. To prevent overfitting, batch normalization is applied to stabilize learning, followed by another LSTM layer with 64 units to further refine temporal features. The model output is generated through a dense layer, predicting future precipitation values for the defined forecasting step. The entire network is optimized using the Adam optimizer with a mean squared error (MSE) loss function, ensuring efficient learning while minimizing prediction errors.

BiLSTM

The Long Short-Term Memory (LSTM) model is a specialized type of recurrent neural network (RNN) that provides feedback to each neuron. Its distinctive gating mechanism addresses the issues of vanishing and exploding gradients that occur when RNNs process long sequences. The process is done by processing sequences in both directions simultaneously. Forward LSTM processes the sequence from start to end as shown in **Figure 4** capturing past to present context

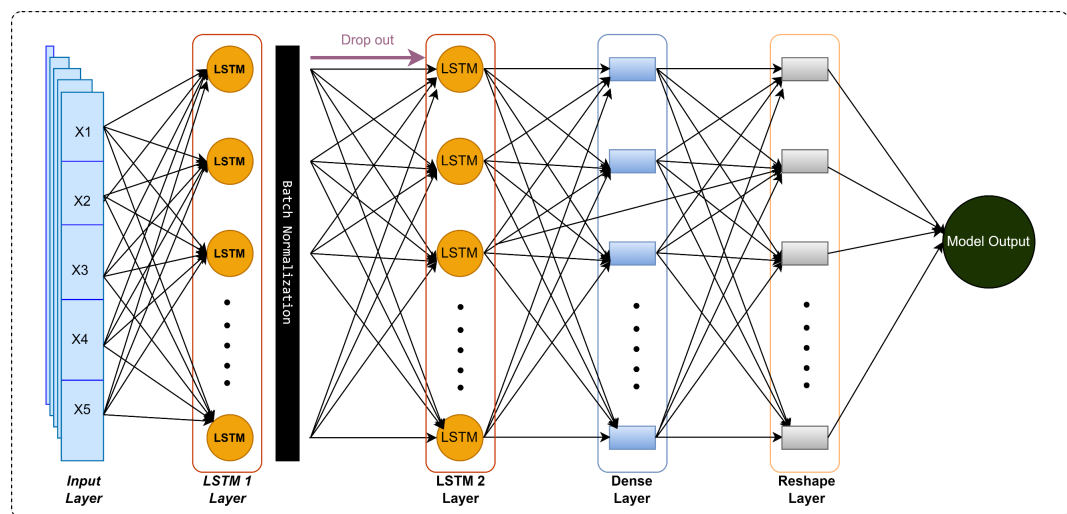


Figure 3. LSTM model architecture for precipitation nowcasting using ERA5 dataset.

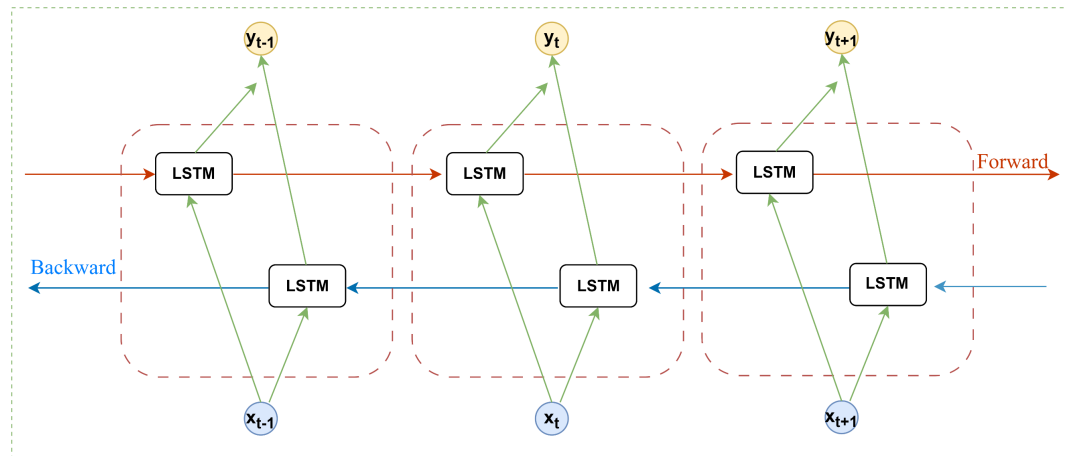


Figure 4. BiLSTM cell.

while the backward LSTM processes the sequence from end to start capturing present to past context. Each BiLSTM layer contains two sets of gates (input, forget, output) for both directions. During forward pass at each timestep, the forward LSTM updates its hidden state based on current input and past memory. During back pass the backward LSTM processes the sequence in reverse, updating its hidden state based on the future inputs and past memory. The outputs of both directions are concatenated and fed into the next layer. The merged representation captures temporal dependencies in both directions, enabling richer feature extraction. BiLSTM takes into account both the preceding and following correlations in the sequence, while also addressing the prediction lag issue that can occur in unidirectional LSTM. The structure of BiLSTM.

BiLSTM Model Process

The input data, consisting of temperature, pressure, U-wind, V-wind, and precipitation, is first processed through a masking layer to handle missing values. The sequence is then passed through a Bidirectional LSTM layer with 64 units, which allows the model to learn patterns in both past and future directions simultaneously as in **Figure 5**. To improve stability and convergence, batch normalization is applied after each BiLSTM layer. The network includes two stacked BiLSTM layers, each refining the learned temporal features before reaching the dense output layer, which predicts future precipitation values. The model is trained using the Adam optimizer with mean squared error (MSE) loss and mean absolute error (MAE) as an evaluation metric.

3. Experiment

3.1. Data Preprocessing

Deep learning models rely heavily on the quality and consistency of input data to make accurate predictions. To ensure high-quality data before training, several preprocessing techniques were applied, including handling missing values, outlier detection, normalization, and sequence generation. These steps help improve model stability, reduce bias, and enhance generalization.

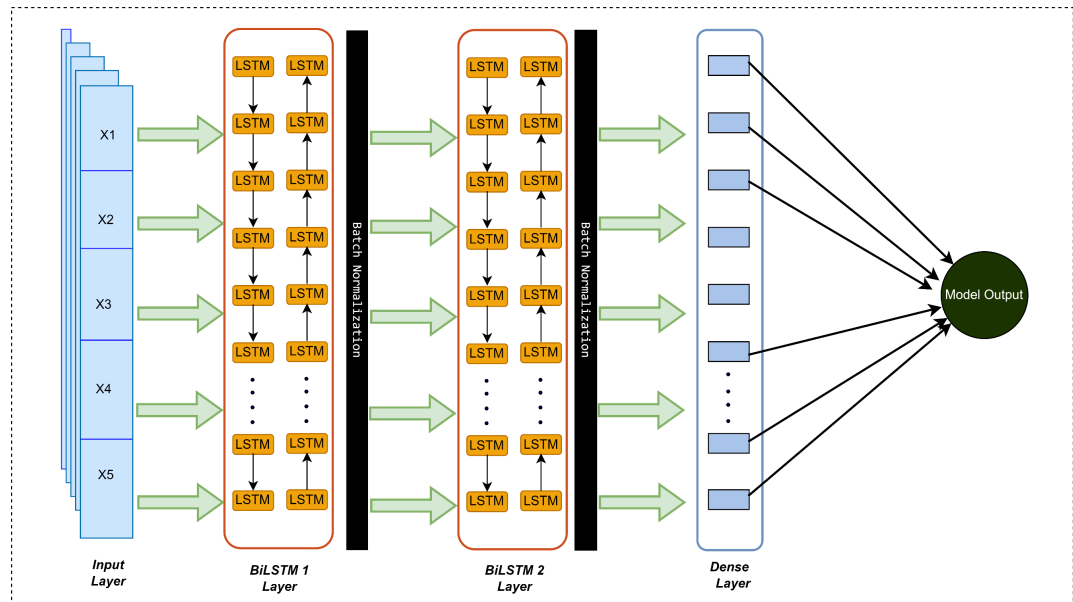


Figure 5. BiLSTM model architecture for predicting Dar es Salaam rainfall.

Missing values. Missing values in time-series data can introduce bias or disrupt learning in deep learning models. To ensure data continuity linear interpolation was used to estimate missing values by fitting the linear relationship between the near points.

$$x_t = x_{t_1} + \frac{(t - t_1)}{(t_2 - t_1)} \times (x_{t_2} - x_{t_1})$$

where:

x_{t_1} and x_{t_2} are the nearest valid values before and after t respectively.

t_1 and t_2 are their respective time indices.

x_t is the interpolated value at time t .

Outlier Detection and Removal. Isolation Forest algorithm was applied to Identify and remove outliers that might lead to poor predictions. The algorithm works by detecting rare values by evaluating how easily they can be separated. A contamination parameter of 0.01 was set to limit the fraction of detected anomalies. And lastly, identified outliers were removed before further preprocessing.

Normalization. Since deep learning models work better when features are on similar scales, all five features are normalized using min-max scaling of the data between [0, 1] also negative precipitations do not exist.

Minmax equation:

$$x' = \frac{x - x_{\min}}{x_{\max} - x_{\min}}$$

where:

x' : The normalized value, which is scaled between 0 and 1.

x : The original value from the dataset.

x_{\min} : The minimum value in the dataset.

x_{\max} : The maximum value in the dataset.

Temporal sequence generation to enable deep learning models to learn temporal dependencies, data was converted into sequences using sliding window processing (Nambirajan & Rajalakshmi, 2024). Past time steps were used as input, and future steps were predicted. Sequence generation ensured the model learned both short- and long-term dependencies. For a given past window size P and future step size F , the sequences were constructed as

$$X = x_t, x_{t+1}, \dots, x_{t+P-1}, Y = x_{t+P}, \dots, x_{t+P+F-1}$$

where X represents input features over past P timesteps, and Y represents the future values the model learns to predict. Then the data will be divided into training 80% and validation 20%.

3.2. Evaluation Indicators

To evaluate the BiLSTM and LSTM model's performance in rainfall nowcasting for flood-prone Dar es Salaam, three metrics were prioritized based on their relevance to operational forecasting and disaster preparedness:

Threat Score (TS) TS measures the model's ability to correctly predict rainfall events exceeding critical thresholds (0.1 mm, 3.0 mm, 5.0 mm). For flood regions like Dar es Salaam's Msimbazi Basin, TS at 5.0 mm is particularly important, as it reflects the model's skill in detecting heavy rainfall events that trigger urban flooding. A higher TS indicates fewer missed alarms (e.g., undetected storms) and reduced false alerts, which are critical for maintaining public trust in early warnings.

$$TS = \frac{hits}{hits + false\ alarms + missed}$$

Mean Absolute Error (MAE) MAE quantifies the typical deviation between predicted and observed rainfall amounts. This metric is essential for water resource planners, as even small persistent errors (e.g., 1 - 2 mm) in daily rainfall accumulation can lead to incorrect reservoir management or crop irrigation decisions.

$$MAE = \frac{1}{N} \sum_{i=1}^N |y_{\text{true}}^{(i)} - y_{\text{pred}}^{(i)}|$$

Root Mean Squared Error (RMSE) RMSE penalizes large errors disproportionately, making it sensitive to extreme rainfall mispredictions. In Dar es Salaam, where short-duration, high-intensity rains (e.g., 10 mm/hour) overwhelm drainage systems, RMSE directly correlates with the model's ability to mitigate catastrophic flood losses.

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (y_{\text{true}}^{(i)} - y_{\text{pred}}^{(i)})^2}$$

Coefficient of Determination (R^2): R^2 measures the proportion of variance in observed precipitation that is explained by the model predictions. A higher R^2 indicates better alignment between predicted and actual rainfall values. In the con-

text of Dar es Salaam, where rainfall variability is high due to convective systems and localized weather events, a strong R^2 reflects the model's ability to capture key temporal patterns and improve forecast reliability for early warning and flood preparedness.

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

where:

y_i = reanalysis values.

\hat{y}_i = Model prediction.

\bar{y} = Mean of reanalysis values.

n = number of data points.

To ensure transparency and reproducibility, we provide the detailed parameter settings used in our experiments. As shown in **Table 1**, the data were divided into training and testing sets using an 80:20 ratio and sequence length of 175,296 time steps, representing data from 2010 to 2023 provide an appropriate foundation for the model's temporal learning capabilities. The models were trained to predict 30 hours into the future. The selected learning rate of 0.0001 falls within the optimal range for LSTM/BiLSTM architectures, as demonstrated by [Siami-Namini, Tavakoli, & Namin \(2019\)](#) in their comparative analysis of time series prediction models. The batch size of 64 has an effective balance between computational efficiency and model stability, supported by recent findings in batch size optimization research ([Hwang et al., 2024](#)). Z-score normalization was applied to all input features to ensure consistent scaling ([Kim et al., 2025](#)). The architecture's use of two recurrent layers with 64 hidden units each, combined with a masking layer (mask value is 0.0) for handling missing values, follows established best practices documented by [Che et al. \(2018\)](#). The 200-epoch training duration and 30-hour prediction horizon represent a well-balanced approach to achieving model convergence while maintaining forecast accuracy.

Table 1. BiLSTM and LSTM parameters settings.

CATEGORY	DESCRIPTION	CONFIGURATION
DATA & TRAINING	Split fraction (train/test)	0.8
	Sequence step size	1
	Input sequence length (past)	175,296
	Prediction horizon (future step)	30
	Learning rate	0.0001
	Batch size	64
	Training epochs	200
DATA PREPROCESSING	Target variable	precip (precipitation)
	Scaling method	Z-score (Z-score normalization)

Continued

MODEL ARCHITECTURE	LSTM MODEL	2 LSTM LAYERS (64 UNITS EACH)
	Masking layer (mask 0.0)	
	BiLSTM Model	2 Bidirectional LSTM layers (64 units each)
METRICS	LSTM: MSE loss	
	BiLSTM: MSE loss, MAE metric	
SPATIAL PARAMETERS	Bounding box (Dar es Salaam)	[39.1, -7.1, 39.5, -6.6]
TEMPORAL PARAMETERS	Date range	2019 to 2024 (Hourly)

4. Results and Discussion

4.1. Training Loss Curves

The training loss curves for LSTM and BiLSTM models in **Figure 6**, indicate a clear difference in learning efficiency and overall performance for precipitation prediction. The BiLSTM model consistently achieves a lower loss compared to the LSTM model throughout the 200 training epochs. Initially, both models exhibit a steep decline in loss, indicating effective learning, but BiLSTM converges faster, reducing its loss at a more rapid rate than LSTM. This suggests that BiLSTM benefits from its bidirectional structure, capturing dependencies from both past and future time steps, making it more effective in learning precipitation patterns. The training curves for both models are relatively smooth, indicating stable learning without severe fluctuations or instability. Overall, BiLSTM outperforms LSTM by demonstrating better convergence, lower training loss, and improved learning efficiency.

4.2. Prediction Curves

The graph in **Figure 7** presents a comparative analysis of precipitation forecasting using LSTM and BiLSTM models against the true precipitation values. The x-axis represents the forecast hours, while the y-axis denotes the precipitation. The true precipitation is depicted as a solid blue line, while the LSTM and BiLSTM forecasts are represented by dashed red and magenta lines, respectively.

The reference precipitation pattern is relatively stable values between 1 and 16 hours, followed by a sharp decline beyond 16 hours. This suggests a period of consistent precipitation then possibly the precipitation stopped and started again. The LSTM model successfully tracks general fluctuations in precipitation but tends to underestimate sudden drops and exhibits lag in responding to sharp transitions. Its forecasts are relatively smoothed, particularly around the sharp dip at hour 17 and during the rising trend beyond hour 20, indicating potential difficulty in adapting to abrupt atmospheric changes.

In contrast, the BiLSTM model displays a more stable and align with the reference precipitation timeline across most forecast hours. Even during the sharp drop around hour 17 and the rise that follows, BiLSTM adjusts well without big errors. This shows that it can handle sudden changes in weather while still giving

reliable forecasts. This improved performance can be attributed to bidirectional processing, which allows the model to integrate information from both past and future time steps.

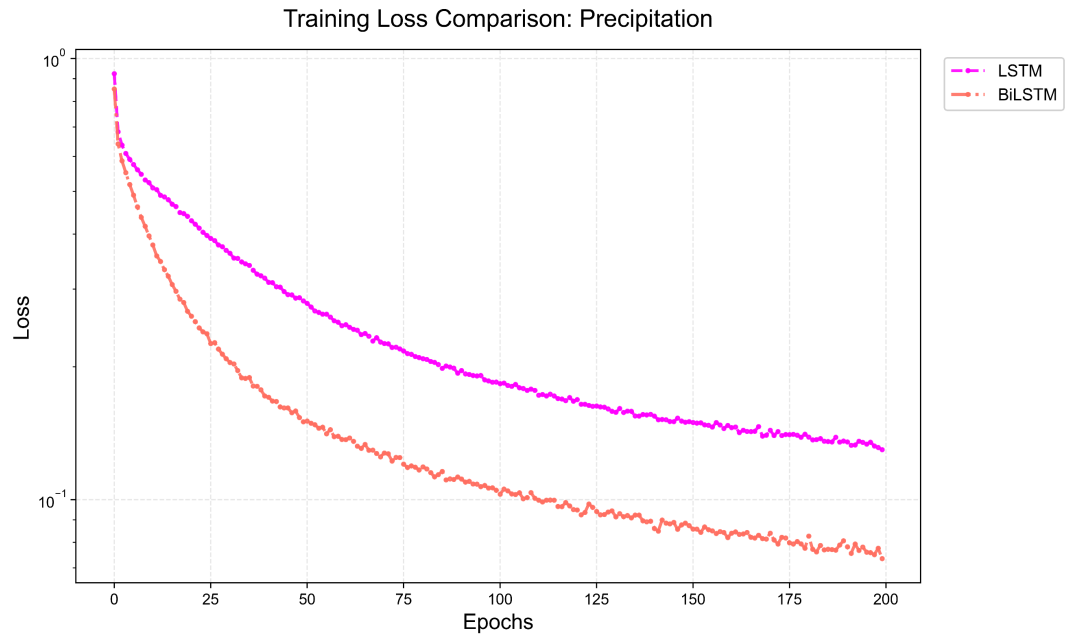


Figure 6. This figure compares the training loss of LSTM and BiLSTM models for precipitation prediction over 200 epoch.

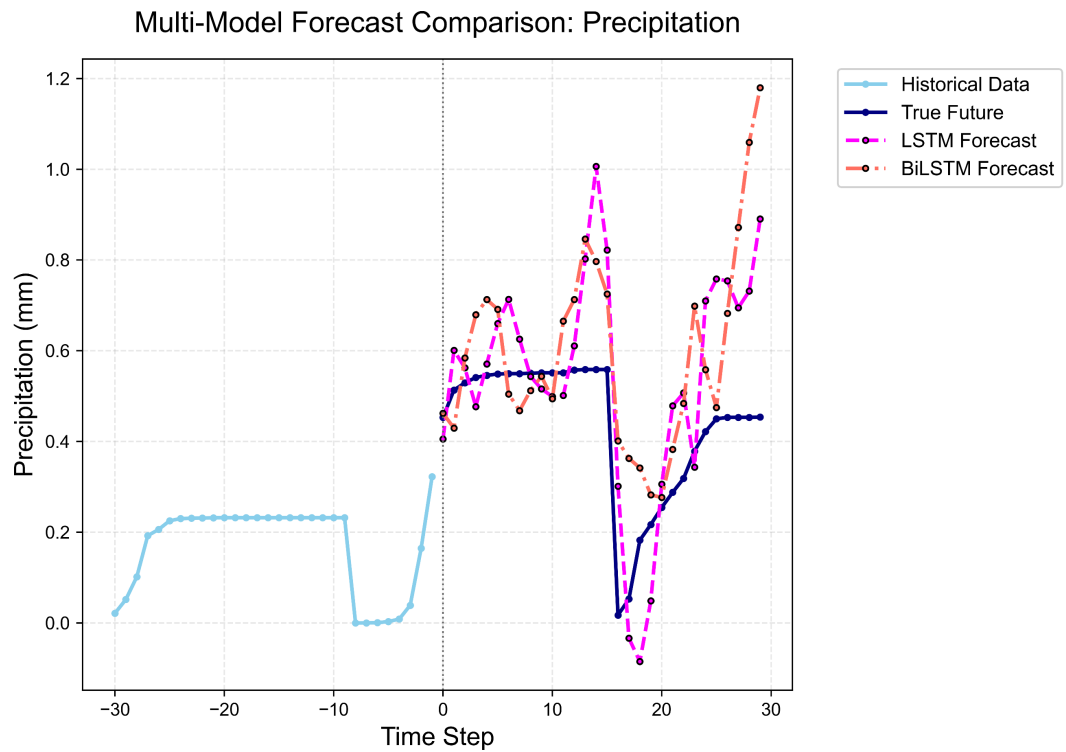


Figure 7. This Plot shows the comparison between True values, BiLSTM and LSTM hourly forecast for 30 hours.

A significant deviation occurs beyond 26, where both models struggle to accurately represent the observed precipitation drop-off. The LSTM model fails to capture the rapid decrease, while the BiLSTM model overshoots the decline, leading to higher variability. This suggests that while BiLSTM is more adaptive to sudden changes, it may also introduce excessive fluctuations in relatively stable conditions, potentially indicating overfitting to short-term variations.

4.3. Statistical Evaluation Indicators

The performance of LSTM and BiLSTM models is evaluated using key error metrics: Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and Threat Scores (TS) at different precipitation thresholds (1.0 mm, 5.0 mm, and 10.0 mm) as shown in **Figure 8**. These metrics provide insight into the overall accuracy, error distribution, and skill in predicting different rainfall intensities.

Error Metrics (MAE & RMSE) The Mean Absolute Error (MAE) for BiLSTM (0.465) is lower than that of LSTM (0.496), indicating that BiLSTM produces more accurate precipitation forecasts on average. Similarly, the Root Mean Squared Error (RMSE), which penalizes larger errors more heavily, is also lower for BiLSTM (0.888) compared to LSTM (0.931). This suggests that BiLSTM not only provides a better overall fit to the data but also reduces the impact of extreme errors, making it more reliable for operational forecasting.

Threat Scores (TS) for Different Rainfall Thresholds. Threat Scores (TS) measure the model's ability to correctly identify precipitation events at different intensity levels:

TS1.0 (Light Rain Detection): BiLSTM (0.554) outperforms LSTM (0.544), showing a slightly better skill in detecting light precipitation. This indicates that BiLSTM is better at capturing weak but significant precipitation events, which are often influenced by small-scale atmospheric processes like moisture convergence.

TS5.0 (Moderate Rain Detection): BiLSTM (0.320) performs better than LSTM (0.313) at detecting moderate rainfall events. This is crucial in meteorology, as moderate rainfall is often associated with frontal systems, convective clusters,

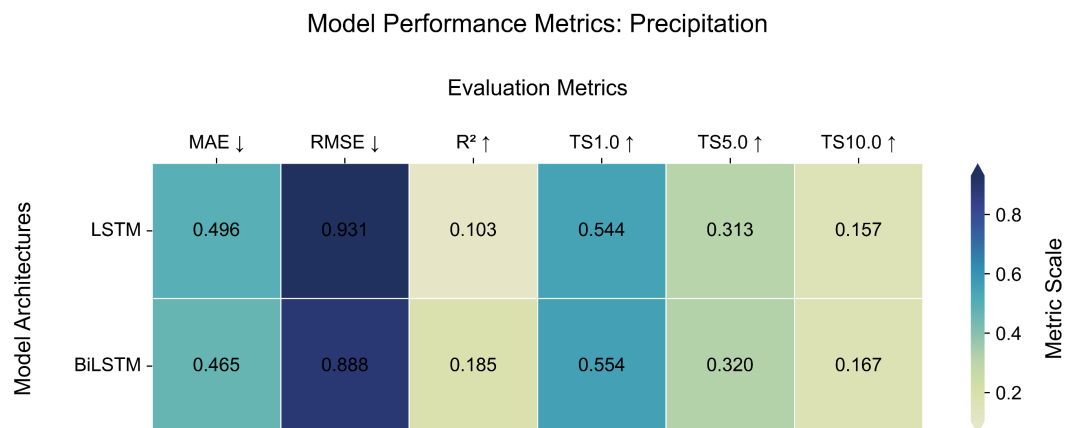


Figure 8. Different evaluation metrics for LSTM and BiLSTM models.

and mesoscale dynamics. The improvement suggests that BiLSTM's ability to process bidirectional information allows it to better track the development and dissipation of such precipitation systems.

TS10.0 (Heavy Rain Detection): Interestingly, for heavy precipitation (>10 mm), BiLSTM (0.167) outperforms BiLSTM (0.157). This suggests that BiLSTM captures well temporal features when there are enough heavy rainfall patterns in the data to prevent the model from introducing noise or overfitting. Bidirectional architecture likely enhances the model's ability to recognize complex rainfall patterns by incorporating both forward and backward dependencies, which are particularly important for identifying the build-up and dissipation phases of heavy rainfall.

5. Conclusion

In this article, LSTM and BiLSTM architectures were developed and compared for precipitation nowcasting using ERA5 hourly data over Dar es Salaam region. The models were trained with an extended dataset from 2010 to 2023 to better capture long-term precipitation patterns and improve performance on extreme events. The results demonstrated that the BiLSTM model consistently outperforms the standard LSTM in terms of overall accuracy, mainly due to its ability to capture both past and future temporal dependencies. However, in smaller datasets, the bidirectional nature of BiLSTM may introduce noise and increase the risk of overfitting. With the extended training period, BiLSTM demonstrated improved generalization.

A significant improvement in results could be achieved by incorporating features at different pressure levels to better represent the atmospheric influences. The use of BiLSTM is scientifically supported due to its lower error rates and better TS scores for precipitation. However, attention should be paid to prevent overfitting when training with limited data. Future improvements may include integrating attention mechanisms to strengthen the model's ability to capture extreme rainfall events while preserving generalization.

Author Contribution

All authors contributed to the study's conception and design. I.J.J., J.B.T. and A.O. conceived the idea of the study; I.J.J. and D.J.M. designed the model and plotted the figures; I.J.J., J.B.T. and G.M. collected the preliminary data and wrote the manuscript. All authors read and agree to the published version of the manuscript.

Data Availability Statement

The datasets to train and prove the neural network were obtained from Copernicus Climate Change Service (C3S) on website <https://cds.climate.copernicus.eu/datasets>.

Conflicts of Interest

The authors declare there are no conflicts.

References

- Ali, M., Deo, R. C., Xiang, Y., Li, Y., & Yaseen, Z. M. (2020). Forecasting Long-Term Precipitation for Water Resource Management: A New Multi-Step Data-Intelligent Modelling Approach. *Hydrological Sciences Journal*, *65*, 2693-2708. <https://doi.org/10.1080/02626667.2020.1808219>
- Barros, A. P., & Lettenmaier, D. P. (1994). Dynamic Modeling of Orographically Induced Precipitation. *Reviews of Geophysics*, *32*, 265-284. <https://doi.org/10.1029/94rg00625>
- Bengio, Y., Simard, P., & Frasconi, P. (1994). Learning Long-Term Dependencies with Gradient Descent Is Difficult. *IEEE Transactions on Neural Networks*, *5*, 157-166. <https://doi.org/10.1109/72.279181>
- Che, Z., Purushotham, S., Cho, K., Sontag, D., & Liu, Y. (2018). Recurrent Neural Networks for Multivariate Time Series with Missing Values. *Scientific Reports*, *8*, Article No. 6085. <https://doi.org/10.1038/s41598-018-24271-9>
- Ebtehaj, I., & Bonakdari, H. (2024). CNN Vs. LSTM: A Comparative Study of Hourly Precipitation Intensity Prediction as a Key Factor in Flood Forecasting Frameworks. *Atmosphere*, *15*, 1082. <https://doi.org/10.3390/atmos15091082>
- Elman, J. L. (1990). Finding Structure in Time. *Cognitive Science*, *14*, 179-211. https://doi.org/10.1207/s15516709cog1402_1
- Gers, F. A., Schmidhuber, J., & Cummins, F. (2000). Learning to Forget: Continual Prediction with Lstm. *Neural Computation*, *12*, 2451-2471. <https://doi.org/10.1162/089976600300015015>
- Graves, A. (2012). Supervised Sequence Labelling. In *Studies in Computational Intelligence* (pp. 5-13). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-24797-2_2
- Hersbach, H., Bell, B., Berrisford, P., Hirahara, S., Horányi, A., Muñoz-Sabater, J. et al. (2020). The ERA5 Global Reanalysis. *Quarterly Journal of the Royal Meteorological Society*, *146*, 1999-2049. <https://doi.org/10.1002/qj.3803>
- Hess, P., & Boers, N. (2022). Deep Learning for Improving Numerical Weather Prediction of Heavy Rainfall. *Journal of Advances in Modeling Earth Systems*, *14*, e2021MS002765. <https://doi.org/10.1029/2021ms002765>
- Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term Memory. *Neural Computation*, *9*, 1735-1780. <https://doi.org/10.1162/neco.1997.9.8.1735>
- Hwang, J., Lee, S., Gil, J., & Lee, C. (2024). Determination of Optimal Batch Size of Deep Learning Models with Time Series Data. *Sustainability*, *16*, Article 5936. <https://doi.org/10.3390/su16145936>
- Jerome Glago, F. (2021). Flood Disaster Hazards; Causes, Impacts and Management: A State-of-the-Art Review. In *Natural Hazards—Impacts, Adjustments and Resilience*. Intech Open. <https://doi.org/10.5772/intechopen.95048>
- Kai, K. H., Ngwali, M. K., & Faki, M. M. (2021). Assessment of the Impacts of Tropical Cyclone Fantala to Tanzania Coastal Line: Case Study of Zanzibar. *Atmospheric and Climate Sciences*, *11*, 245-266. <https://doi.org/10.4236/acs.2021.112015>
- Kim, Y., Kim, M. K., Fu, N., Liu, J., Wang, J., & Srebric, J. (2025). Investigating the Impact of Data Normalization Methods on Predicting Electricity Consumption in a Building Using Different Artificial Neural Network Models. *Sustainable Cities and Society*, *118*, Article 105570. <https://doi.org/10.1016/j.scs.2024.105570>
- Kratzert, F., Klotz, D., Brenner, C., Schulz, K., & Herrnegger, M. (2018). Rainfall-Runoff Modelling Using Long Short-Term Memory (LSTM) Networks. *Hydrology and Earth System Sciences*, *22*, 6005-6022. <https://doi.org/10.5194/hess-22-6005-2018>
- Lipton, Z. C. (2015). *A Critical Review of Recurrent Neural Networks for Sequence Learn-*

ing. <http://export.arxiv.org/pdf/1506.00019>

- Mason, S. (2016). *Guidance on Verification of Operational Seasonal Climate Forecasts*. <https://doi.org/10.7916/d8-gh4a-ex60>
- Moeletsi, M. E., Mellaart, E. A. R., Mpandeli, N. S., & Hamandawana, H. (2013). The Use of Rainfall Forecasts as a Decision Guide for Small-Scale Farming in Limpopo Province, South Africa. *The Journal of Agricultural Education and Extension*, *19*, 133-145. <https://doi.org/10.1080/1389224x.2012.734253>
- Nambirajan, V., & Rajalakshmi, V. (2024). Climatological Rainfall Forecasting Using LSTM: An Analysis of Sequential Input and Data Window Input Approaches. In *Lecture Notes in Networks and Systems* (pp. 311-321). Springer. https://doi.org/10.1007/978-981-99-7814-4_25
- NBS (2022). *Tanzania Population and Housing Census 2022. National Demographic Socio-Economic Profile*. <https://microdata.nbs.go.tz/index.php/catalog/45>
- Owiti, Z. (2012). Spatial Distribution of Rainfall Seasonality over East Africa. *Journal of Geography and Regional Planning*, *5*, 409-421. <https://doi.org/10.5897/jgrp12.027>
- Piran, M. J., Wang, X., Kim, H. J., & Kwon, H. H. (2024). Precipitation Nowcasting Using Transformer-Based Generative Models and Transfer Learning for Improved Disaster Preparedness. *International Journal of Applied Earth Observation and Geoinformation*, *132*, Article 103962. <https://doi.org/10.1016/j.jag.2024.103962>
- Priatna, M. A., & Djamal, E. C. (2020). Precipitation Prediction Using Recurrent Neural Networks and Long Short-Term Memory. *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, *18*, 2525. <https://doi.org/10.12928/telkomnika.v18i5.14887>
- Sakijege, T., Lupala, J., & Sheuya, S. (2012). Flooding, Flood Risks and Coping Strategies in Urban Informal Residential Areas: The Case of Keko Machungwa, Dar Es Salaam, Tanzania. *Jambá: Journal of Disaster Risk Studies*, *4*, a46. <https://doi.org/10.4102/jamba.v4i1.46>
- Schuster, M., & Paliwal, K. K. (1997). Bidirectional Recurrent Neural Networks. *IEEE Transactions on Signal Processing*, *45*, 2673-2681. <https://doi.org/10.1109/78.650093>
- Sherstinsky, A. (2020). Fundamentals of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) Network. *Physica D: Nonlinear Phenomena*, *404*, Article 132306. <https://doi.org/10.1016/j.physd.2019.132306>
- Siami-Namini, S., Tavakoli, N., & Namin, A. S. (2019). *A Comparative Analysis of Forecasting Financial Time Series Using ARIMA, LSTM, and BiLSTM*. <https://arxiv.org/abs/1911.09512v1>
- Waqas, M., & Humphries, U. W. (2024). A Critical Review of RNN and LSTM Variants in Hydrological Time Series Predictions. *MethodsX*, *13*, 102946. <https://doi.org/10.1016/j.mex.2024.102946>
- Wu, Y. T., & Xue, W. (2024). Data-Driven Weather Forecasting and Climate Modeling from the Perspective of Development. *Atmosphere*, *15*, Article 689. <https://doi.org/10.3390/atmos15060689>
- Xu, Y., Hu, C., Wu, Q., Jian, S., Li, Z., Chen, Y. et al. (2022). Research on Particle Swarm Optimization in LSTM Neural Networks for Rainfall-Runoff Simulation. *Journal of Hydrology*, *608*, Article 127553. <https://doi.org/10.1016/j.jhydrol.2022.127553>
- Zhang, X., Shi, J., Chen, H., Xiao, Y., & Zhang, M. (2023). Precipitation Prediction Based on CEEMDAN-VMD-BiLSTM Combined Quadratic Decomposition Model. *Water Supply*, *23*, 3597-3613. <https://doi.org/10.2166/ws.2023.212>