



Prediction of Rainfall Using Long Short-Term Memory Using Encoder-Decoder

Kaushlendra Yadav^{1,*} and Arvind Kumar Tiwari¹

ARTICLE INFO

Article history:

Received: 23 May 2024

Revised: 14 September 2024

Accepted: 2 February 2025

Online: 31 October 2025

Keywords:

Rainfall prediction

LSTM encoder-decoder

Deep learning

Machine learning

Convolutional neural networks

ABSTRACT

Rainfall prediction has been a critical area of research due to its significant impact on several industries such as agriculture, water resource management, and catastrophe preparedness. Anticipating rainfall provides crucial awareness, enabling proactive measures to be taken in advance. Precise estimations of daily rainfall not only enhance agricultural productivity but also play a pivotal role in securing food and water supplies. Machine Learning and the more recent Deep Learning based approaches have shown promising results in modelling complex patterns inherent in meteorological data. This paper presents an in-depth exploration and evaluation of various ML and DL algorithms, encompassing Naïve Bayes, Decision Trees, Support Vector Machines, K-Nearest Neighbour, Artificial Neural Network, XgBoost, Convolutional Neural Networks, Long Short-Term Memory Network, RNN with LSTM and LSTM Encoder-Decoder. The evaluation metrics of Mean Square Error (MSE), Root Mean Squared Error (RMSE), Matthews Correlation Coefficient (MCC), Accuracy, and Precision are employed to assess the performance of the model. Here, the proposed approach LSTM Encoder-Decoder achieves accuracy 92.13%. Here, the comparative analysis is performed with machine learning and demonstrated how well the suggested method predicts rainfall when compared to alternative methods.

1. INTRODUCTION

Predicting rainfall is a pivotal yet inherently challenging task within the domain of meteorology. The accurate prediction of the rainfall increases the output produced in agricultural field, ensure proper food and water supply, providing it to a healthy population. Anticipating rainfall can help avoid flooding, save human lives and property. The shortage of rainfall adversely affects the aquatic ecosystem, water supply quality, and agricultural productivity. Hence, forecasting is essential for understanding the state of the atmosphere. Predictive analytics emerges as a cutting-edge analytical approach for conducting such predictions and anticipating future events based on historical datasets. It plays a crucial role in prompting various human activities, including the generation of power, production related to agriculture, tourism, and forestry [1]. The prediction of meteorological phenomena is intricate, primarily attributed to the challenge of obtaining accurate data, with instances where data collection is either impossible or may yield inaccurate information from highly sensitive sensors [2]. Therefore, predictions are made using probabilistic models [3]. Lately, there has been a widespread adoption of machine learning and deep learning algorithms for modelling phenomena that involve substantial and diverse datasets.

Many factors are responsible for this trend. Firstly, machine learning algorithms demonstrate proficiency in handling large volumes of data [4]. Secondly, they exhibit the capability to unveil patterns of behaviour or implicit relationships within processed data that may not be directly apparent [5]. For identifying patterns in past weather data, machine learning algorithms like Random Forests, Support Vector Machines (SVM), and Gradient Boosting work well. However, deep learning algorithms are adept at modelling transitory dependencies in sequence data, especially Recurrent Neural Networks (RNNs) and Long Short-Term Memory networks (LSTMs). As a result, these algorithms are excellent at processing big datasets, identifying intricate patterns, and forecasting future events based on past patterns. This paper's goal is to perform an exploratory evaluation with an emphasis on modelling rainfall prediction using machine learning and deep learning algorithms. It explores the challenges associated with conventional methods and highlights the potential of data-driven approaches in enhancing prediction accuracy. The utilization of various climatic parameters, such as speed of the wind, temperature, atmospheric pressure, and humidity, in combination with historical rainfall data, forms the basis for training these models.

¹KNIT Sultanpur, Uttar Pradesh, India.

*Corresponding author: Kaushlendra Yadav; Email: kaushal202@gmail.com.

2. RELATED WORKS

The literary survey encompassed a wide array of intriguing techniques aimed at conducting weather forecasts using various machine learning and deep learning algorithms. This paper presents study of multiple algorithms used in past by various other authors. Recent research has brought attention to the prevailing weather conditions can perhaps affect various things like increase in air pollution [6], solar radiations [7] and several additional factors. The research and literary survey have been made for the rainfall prediction model that widely affects or is a major reason behind many catastrophic disasters like flood [8], drought [9], landslide [10], rise in local mean sea level [11] and so on. Although a substantial part of historical forecasting methods relies on differential equations and physics-based simulations, recent studies show the advancements that have introduced alternative approaches from machine learning and deep learning. These methods primarily leverage machine learning techniques such as Bayesian network or Naïve Bayes [12], SVM [13], decision trees [14], K-nearest neighbor [15], ANN while CNN included [16], RNN [17], and further advancements in these neural networks using long-short term memory (LSTM) algorithms. In paper [18], the authors combined 5 machine learning (ML) algorithms Decision trees, KNN, Random Forest, SVM, and AdaBoost) together and used a hybrid approach of Principal Component Analysis (PCA) with the stochastic machine learning (ML) algorithms. In the recent studies it has been seen that authors tend more towards the deep learning model rather than stochastic and mathematical machine learning models. The parameters such as precipitation, humidity, wind speed at different times, sunshine, cloud characteristics and direction have been used for rainfall prediction and weather forecasting. Authors used Artificial Neural Network (ANN) algorithm to enhance accuracy by taking into account the mentioned parameters and factors to predict weather fluctuations [19]. In paper [20], the authors studied and

implemented various methodologies such as gradient boosting, polynomial regression, recurrent neural network neural and prophet models to the intricate process of weather forecasting. The comprehensive comparative analysis entailed the utilization of various techniques such as ANN, Auto Regression, RNN with the LSTM model to capture non-linear relationships among weather variables and patterns within the dataset. In paper [21], In order to create a sophisticated and statistically sound model for predicting univariate weather variables, the authors employed LSTM. Using a single-layer Long Short-Term Memory (LSTM) model and a multi-layer-LSTM model, the authors investigated the impact of intermediate weather variables on prediction accuracy using data from an Indonesian airport area. In paper [22], The weather translation, which changes an image's weather condition from one category to another, was created by the authors using Generative Adversarial Networks (GAN). For this a large-scale weather image dataset was used which comprised of 5 weather categories. An attention module was integrated to concentrate the image translation process on the region of interest, effectively minimizing superfluous alterations or artifacts. The authors used LSTM to optimize the accuracy of model forecasting by using multiple meteorological factors. The authors employed a real-time dataset covering eight meteorological parameters from the European Centre for Medium-Range Weather Forecasting (ECMWF). The relationships between meteorological elements and the predicted rainfall were identified through the use of feature extraction and continuous debugging throughout the LSTM model's training, allowing for precise modifications to the model's rainfall forecasts. [23].

The author mentioned the various models and their algorithms in terms of progress of research with description. See Table 1.

Table 1. Summary of Machine Learning based approaches for weather prediction

Models	Year	Author	Dataset	Result	Description
ANN	2012	Kumar Abhishek, et. al.	Data available from the station Toronto Lester B. Pearson Int'l A, Ontario, Canada	MSE=1.52	According to the findings, the model performs better overall when there are more neurons per layer because this lowers Mean Squared Error (MSE) [24].
	2015	Mislana, et. al.	Rainfall dataset from BMKG Tenggarong Station and Universitas Mulawarman	MSE=0.0009631	The study's findings suggest that the Artificial Neural Network (ANN) model can function as a powerful prediction algorithm with a high degree of predicted accuracy [25].

KNN	2008	Zahoor Jan, et. al.	National Climatic Data Center (NCDC)	Accuracy=96.66%	In the experiment, it was observed that for a smaller dataset size of 40,000, the range of k values from 5 to 10 yielded accurate results. However, for a larger dataset size of 80,000, the accurate results were obtained within the range of k values from 35 to 40 [26].
	2013	Jian Hu, et. al.	Nanjing city and the Dahuofang reservoir basin.	RMSE=327.7 MAE=247.3	This work presents a nonparametric model that replicates relevant statistical aspects of the observed time series to forecast yearly average rainfall [27].
	2017	Mingming, et. al.	Rainfall dataset Beijing	Accuracy=48.37%	When working with precipitation datasets that have irregular distributions, the KNN algorithm's resilience to different neighborhood size selections is particularly clear [28].
Naïve Bayes	2019	Li Tiancheng, et. al.	China Meteorological Data Network	Accuracy=76.9%	Experimental results demonstrated that the Naive Bayesian algorithm-based sandstorm prediction model outperformed other approaches in terms of prediction accuracy [29].
	2019	Wei Chen, Yang Li, Weifeng Xue, et. al.	Jiangxi Province Meteorological Bureau	Accuracy=83.5%	The results showed that Naïve Bayes, along with all other models, demonstrated favorable outcomes.[30]
	2020	A U Azmi, et. al.	BMKG Banyuwangi Meteorological Station	Accuracy=95.91%	The test outcomes indicate a relatively high accuracy in classifying rainfall categories [31].
RNN	2021	Han, J. M., et. al.	Harvard Gund Hall data (Jan to May)	MSE= 0.3969	Validation of the model showed how important it is to use localized weather data in building performance simulations. The GRU, one of the RNN models, performed better than the regular models [32].
LSTM	2019	Pradeep Hewage, Ardhendu et.al.	Surface Weather Data Global Forecast System 0.25° dataset	MSE: 0.0168 Accuracy: 79%	For the Snow and Soil Moisture (SMOIS) variables, the Weather Research and Forecasting (WRF) model produces the best results [33].
	2020	Shweta Mittal, et.al.	Integrated Global Radiosonde Archive (IGRA) dataset	MAE : 0.04	To evaluate their impact on the accuracy of the network, the number of hidden layers was increased to three. A network with two hidden layers was found to attain acceptable accuracy with less learning time. Additionally, experimental findings indicate that the Adam optimizer outperforms SGD and RMSProp optimizers, delivering superior results [34].
Encoder - Decoder	2021	Kumar, R., et.al.	Real-time data from sensor locations at the farm	Accuracy = 85.97	Utilizing DeepMC for micro-climate predictions enables cost-effective forecasting using accessible IoT sensors. This empowers partner farmers to optimize chemical applications, ensuring timely and effective usage [35].
Multiple models	2021	Schultz, M. G., Betancourt, C., Gong, B., Kleinert, F., Langguth, et.al.	Three datasets including meteorological time series dataset	---	Considering the ambition to substitute computationally intensive NWP models with DL algorithms, it's crucial to reassess weather forecasting goals and articulate

					precise criteria for alternative methods [36].
LSTM + Attention model	2018	Yirui Wu, <i>et.al.</i>	Flood dataset of Changhua River	Accuracy:67.59%	Experiments on the Changhua dataset show that the suggested approach outperforms a number of comparison approaches, demonstrating the potency of the context-aware attention model presented in this work [37].
	2022	Suleman, M. A. R., & Shridevi	Real meteorological data of weather station at Saskatoon John G. Diefenbaker Intl. Airport	MSE = 0.0871	Accurate predictions are made even when input sequences abruptly shift because the spatial feature attention method quantifies the mutual influence of input features on the target feature [38].
Transformer	2022	Chattopadhyay, A., Mustafa, <i>et.al.</i>	ECMWF Reanalysis 5 (ERA5) dataset	RMSE (m2 s2) at 3d = 294	The relative spatial correlations of the features in the spatiotemporal data are encoded by integrating a deep spatial transformer into the network architecture's latent space [39].
	2022	Alerskans, E., Nyborg, J., <i>et.al.</i>	GFS dataset with resolution of 0.25	RMSE (C) = 1.50	Even though the Transformer post-processing model was trained using data from less than two years ago, it shows encouraging results [40].
CNN + Attention + LSTM	2023	Hou, X., <i>et. al.</i>	GHI dataset for solar radiance	CNN-A-LSTM RMSE= 0.076	To increase the accuracy of solar irradiance forecast and accomplish the objective of better power system optimal dispatching and management, an enhanced CNN-A-LSTM and attention mechanism are suggested. GBT, SVM, ANN, DNN, LSTM, CNN-DNN, and CNN-LSTM models are contrasted with the suggested model [41].
Deep LSTM	2023	Ilias C, Georgios T., <i>et.al.</i>	Nisorys rainfall dataset since 2017	Accuracy = 96.45%	The suggested machine learning pipeline initiates the deep LSTM regressor's execution solely upon the classifier's prediction of a rainfall event. This approach reduces computational demands while enabling the utilization of a dataset exclusively comprised of rainfall events during the regressor's training phase. Consequently, this method enhances forecasting performance [42].
	2023	Sunkavalli Akash, Sarikonda S, <i>et.al.</i>	Weather dataset over Delhi between 1996 and 2017.	--	The types of different machine learning methods were discussed to find the best model [43].
Generative Adversarial networks	2023	Sigg, C., <i>et. al.</i>	dataset with Sentinel-2 photos that correspond to climatic and topographic factors.	Accuracy = 59%	Propose a method for visualizing weather forecasts using photographic images, a complex task requiring realism and alignment with predictions. Seamless transitions from observed to forecasted conditions and visual consistency between consecutive images are crucial. Leveraging conditional generative adversarial networks (cGANs) enables synthesis of realistic visualizations [44].

3. METHODOLOGY

Rainfall prediction using traditional artificial intelligence relies on predefined rules and explicit programming. It works like step-by-step instructions, guiding the AI system to solve problems or perform tasks. Traditional AI systems, however, struggle with adapting to new or unexpected situations and handling complexity, as they rely heavily on predefined rules and explicit programming. Initial weather forecasting methods were constrained by their dependence on static rules and lacked flexibility. This approach was limited in handling complex, unpredictable weather patterns. Traditional programming laid the groundwork for AI in weather forecasting. It provided a systematic approach to interpreting meteorological data, setting the stage for more dynamic solutions.

Rainfall prediction using Machine Learning (ML) uses algorithms to analyze historical data and predict future patterns. ML models in weather forecasting consider factors like temperature, wind speed, and precipitation, leading to improved accuracy, faster response times, and more efficient forecasting operations. ML algorithms can automate aspects of the forecasting process, thereby enhancing efficiency and accuracy. As weather phenomena became more erratic the need for a more adaptable and accurate forecasting method became apparent. Machine Learning algorithms, applied in institutions like ECMWF, began to analyse historical weather data, enhancing prediction accuracy and responsiveness. This was a significant step towards more reliable and efficient weather forecasting.

Unlike traditional programming or even basic machine learning algorithms that require feature engineering or explicit instructions, deep learning systems automatically identify and learn the relevant features from the data itself. Text, music, and image data are examples of unstructured data that this works particularly well with. Using data instead of mathematical formulas, deep learning creates weather forecasting systems. It uses historical weather data to learn the cause-and-effect links that govern Earth's weather evolution. Deep learning offers more accurate forecasts in less time than traditional systems, making it essential for mitigating the consequences of severe weather conditions. Deep Learning technologies, exemplified by models like Google DeepMind's GraphCast, were able to process vast amounts of historical data, uncovering complex weather patterns with greater precision and speed.

In the context of climate science and related fields, Large Language Models (LLMs) offer unique advantages. They can process and summarize large volumes of scientific literature, making it easier for researchers and policymakers to stay informed. LLMs can also assist in drafting reports, simplifying complex scientific information for public communication, and even generating predictive narratives based on climate data. Large Language Models offered a solution by processing and generating human-like text. They

played a vital role in making climate data more accessible and understandable, thus enhancing the overall utility of weather forecasts.

One significant advancement in artificial intelligence is the capacity of generative AI to synthesis and create new material from learned data. Generative artificial intelligence is based on the ability to process and integrate vast volumes of data, often millions of data points, and generate concise, logical, and often original ideas or content. This is achieved through the use of sophisticated machine learning techniques including Generative Adversarial Networks (GANs), autoencoders, and transformer models. These techniques enable the AI to learn from a dataset and then generate new data instances that are similar but not identical to the original data. For weather forecasting, Generative AI brings a transformative approach. Traditional models in meteorology rely on analyzing historical data to predict future conditions. Generative AI, however, synthesizes this data to provide not just predictions but also actionable insights and comprehensive summaries. This ability makes it particularly valuable for businesses and organizations that require quick, accurate interpretations of complex data to make informed decisions. Generative AI's uniqueness lies in its capability to create outputs that are both informative and easy to assimilate, bridging the gap between complex data analysis and practical, everyday decision-making. This makes it an invaluable tool in fields where data is abundant, but clarity and conciseness are essential. The evolving climate demanded not just accurate forecasts but also actionable insights for better decision-making.

Prediction Techniques

Here, this paper presents some machine learning and deep learning based techniques used for rainfall prediction.

Naive Bayes: The Naive Bayes classifier works on Bayes' theorem, which is used to determine that whether an event will occur or not when another event has already occurred. In a classification, the objective of the task is to assign the input to a single predefined category [45]. The Naive Bayes classifier operates by computing the probabilities that are associated with the input belonging to each category and subsequently selecting the category that has the highest probability as the predicted output. The term "naive" comes from the assumption that, given the class label, the qualities used to describe the input are conditionally independent of one another. According to this, the presence or absence of one attribute has no bearing on the presence or absence of another. [46]. Although, this presumption may not be applicable to all the cases, but it eases the computation and often performs effectively, even when dealing with attributes that are correlated.

Decision Tree: It utilizes tree architecture to represent the relationship among variables [47]. It facilitates the partitioning of data into two or more interconnected sets based on important factors. Initially, the entropy of each

attribute is computed, and subsequently, the data is partitioned based on predictors that exhibit the highest information gain or minimal entropy [48]. Building a model that forecasts the value of the target variable is the primary goal. With each leaf node denoting a distinct class label and the inside nodes representing attributes, the decision tree employs a tree structure to solve the given problem. [49].

K-Nearest Neighbours: The K-Nearest Neighbours (KNN) predict the label or category of a given data point by taking into account the labels of its nearest neighbours in the attribute space. KNN assumes that data points with similar attributes tend to have similar labels [50]. The KNN method typically uses a constant k value and runs several experiments with various k values to produce the best prediction results. The method works by gradually enlarging the area surrounding test sample point y until k training sample points are covered. [51]. Primary challenges with KNN is its heightened susceptibility to hyper parameter configurations, including factors such as the selection of the number of nearest neighbours (k), the choice of distance function, and the type of weighting function used. The neighbours are selected from the dataset and their classes or object property estimates are known [52].

Extreme Gradient Boosting: Extreme Gradient Boosting (Xgboost) is used in regression, classification, ranking, and prediction activities [53]. It represents an enhanced iteration of the gradient boosting algorithm, with a primary objective of improving model efficiency and accuracy by reducing the computation time required for gradient calculations [54]. The capacity of Xgboost to efficiently handle big datasets, attain high accuracy, and handle missing values has led to its rise in popularity. Since decision tree are weak learner, are methodically integrated into the model during the process in which each successive tree is trained to address errors made by its predecessors. Thus, Xgboost constructs a collection of decision trees, with each tree dedicated to capturing patterns in the data that may have been overlooked by the preceding trees [55].

Long Short-Term Memory: One kind of recurrent neural network (RNN) that works well for sequence prediction tasks, such as time series forecasting, is called an LSTM. [56]. Because LSTM can detect long-range dependencies, it performs more accurately than conventional RNNs. Unlike traditional RNNs, LSTM's recurrent hidden layer includes unique units known as memory blocks. [57]. Within these memory blocks, there are memory cells that feature self-connections, serving to preserve the transistor network state. Additionally, specialized multiplicative units, referred to as gates are included to regulate the flow of information. LSTM effectively addresses the issue of error backflow, ensuring that the algorithm utilizes only the relevant error feedback for making more accurate predictions [58]. LSTM can be applied effectively to rainfall prediction problem.

Recurrent Neural Networks: it suffers from the limitation of short-term memory. Consequently, when making predictions with extensive data, RNN has various variations designed to provide solutions. Thus, it commonly encounters the vanishing gradient problem, in which the information gradually tends to fade depending on the activation function. The non-linearity term is frequently insufficient for preserving long-term memory [59]. To address this issue, a Long Short-Term Memory network has been used. LSTMs play a crucial role in retaining issues that are capable of being back-propagated through both time and layers. This preservation of issues enables RNN algorithm to enhance learning efficiency over numerous time-steps [60]. Therefore, the model can capture temporal patterns of rainfall data that include both short-term and long-term dependencies thanks to the integration of RNNs and LSTMs. The sequential nature of RNNs enables the network to process the time series data, while the LSTM architecture helps mitigate the vanishing gradient problem, allowing for better retention of relevant information over extended sequences [61].

4. PROPOSED APPROACH

4.1 Data Set Description

In this comprehensive dataset, an extensive collection of daily weather observations spanning approximately a decade across various locations in Australia is meticulously documented. The focal point of interest is the target variable "Rain Tomorrow," which serves as the key element for predictive modeling. The objective is to forecast whether it will rain on the subsequent day, with binary outcomes indicating either a positive affirmation ("Yes") or a negative one ("No"). Specifically, the affirmative response is assigned when the recorded rainfall for a given day surpasses or equals 1mm. This dataset offers a rich and diverse repository of meteorological information, such as temperature of the place, humidity, speed of the wind, and atmospheric pressure of a particular location, providing an invaluable resource for the development and evaluation of classification models aimed at predicting rainfall patterns in Australia.

4.1.1 Data Collection

The data was obtained from Kaggle (<https://www.kaggle.com/datasets/jsphyg/weather-dataset-rattle-package>), making it a valuable resource for academics, analysts, and enthusiasts. The collection consists of daily weather observations for about 10 years. These observations were made in a variety of Australian places; therefore, the dataset is rich and varied for study. The collection contains 23 features, each of which captures a different aspect of the weather. The dataset, which has 145460 rows, is divided into 20% for testing and 80% for training. These features include temperature, wind speed,

humidity, wind direction and atmospheric pressure. The temporal nature of the data allows for time series analysis, making it particularly relevant for modeling using techniques like LSTM (Long Short-Term Memory) Encoder-Decoder.

4.1.2 Data pre-processing and data analysis:

In preparing the Rain in Australia dataset for analysis, several essential preprocessing steps are undertaken. Initially, missing values are addressed by employing techniques like imputation or removal, depending on the extent of missing data. Additionally, categorical variables, such as “Rain Today”, underwent encoding to facilitate their use in machine learning models. Standardization or normalization is applied to ensure consistent feature scaling, a crucial aspect, especially for models like LSTM Encoder-Decoder, which benefit from uniform scales. Outliers, if present, are identified and treated using robust techniques to prevent their undue influence on model performance. Given the dataset's temporal nature, temporal aggregation is performed, creating sequences of observations to capture time-dependent trends effectively. Lastly, a prudent train-test split is implemented to configure the performance of the model on that data which has not been trained, ensuring unbiased assessments.

During the data training phase for the LSTM Encoder-Decoder model on the Rain in Australia dataset, the pre-processed data underwent careful preparation for model learning. This involved structuring the entire data into input sequences and its corresponding target values, essential for training a time series prediction model. The LSTM architecture is configured with suitable parameters, including the number of LSTM units, activation functions, and input dimensions. Model compilation is done using the Adam optimizer and the Binary cross-entropy loss function. The data was split into 80% for training set, and 20% of the dataset for testing. To avoid over-fitting, early termination criteria were included to the training process, which started over several epochs. To guarantee ideal generalization, the model's performance was continuously tracked on a validation subset. Through this training process, the LSTM Encoder-Decoder learned temporal patterns and dependencies within the data, ultimately enhancing its capability to predict rainfall occurrences effectively.

The testing of the model is performed on the left out 20% of the dataset. The testing phase of the dataset using LSTM (Long Short-Term Memory) Encoder-Decoder involves evaluating the performance of the trained model in predicting “Rain Tomorrow”. After preprocessing the data, splitting it into training sets and testing sets, and training the LSTM Encoder Decoder model, the testing phase is crucial for the assessment of the model's generalization to unseen data. The testing process involves feeding the model with sequences of input features from the testing set and

comparing the predicted “Rain Tomorrow” values with the true labels. Here the is done on 10 number of epochs, sequence length is 10, LSTM units are 100 number, batch size are 128 number and learning rate is 0.7. The 23 number of features are used for rainfall prediction these are: Date, Location, MinTemp, MaxTemp, Rainfall, Evaporation, Sunshine, WindGustDir, WindGustSpeed, WindDir9am, WindDir3pm, WindSpeed9am, WindSpeed3pm, Humidity9am, Humidity3pm, Pressure9am, Pressure3pm, Cloud9am, Cloud3pm, Temp9am, Temp3pm, RainToday and RainTomorrow. The efficacy of the model is assessed using performance indicators like accuracy, precision, and the Matthews Correlation Coefficient (MCC).

4.2 Model architecture

In the LSTM Encoder-Decoder model architecture for rainfall prediction, the LSTM Layer records the temporal dependencies and encodes the information into a fixed-size context vector, while the Encoder's Input Layer processes the time series data of previous rainfall observations. In order to return the complete sequence for every time step, it sets the number of layers and LSTM units according to the problem's complexity and return sequences to true. Then in Intermediate Context Vector the Repeat Vector Layer repeats the encoded sequence to match the length of the decoder input sequence, this helps in providing the context vector for each time step of the decoder. After this in the Decoder the LSTM Layers decodes the context vector into a sequence of future rainfall predictions. It configures the amount of LSTM units and the number of layers similarly to the encoder and return sequences set to true to output a sequence for each time step [62] (See Fig.1).

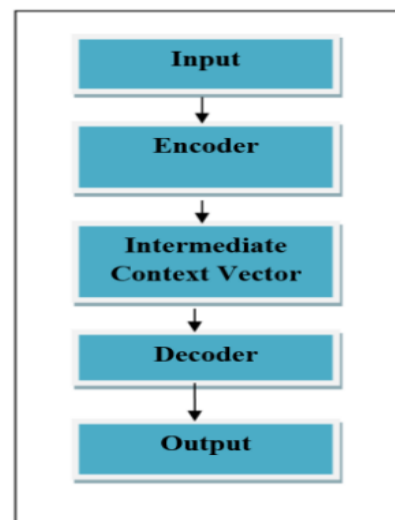


Fig. 1. LSTM Encoder-Decoder Model Architecture.

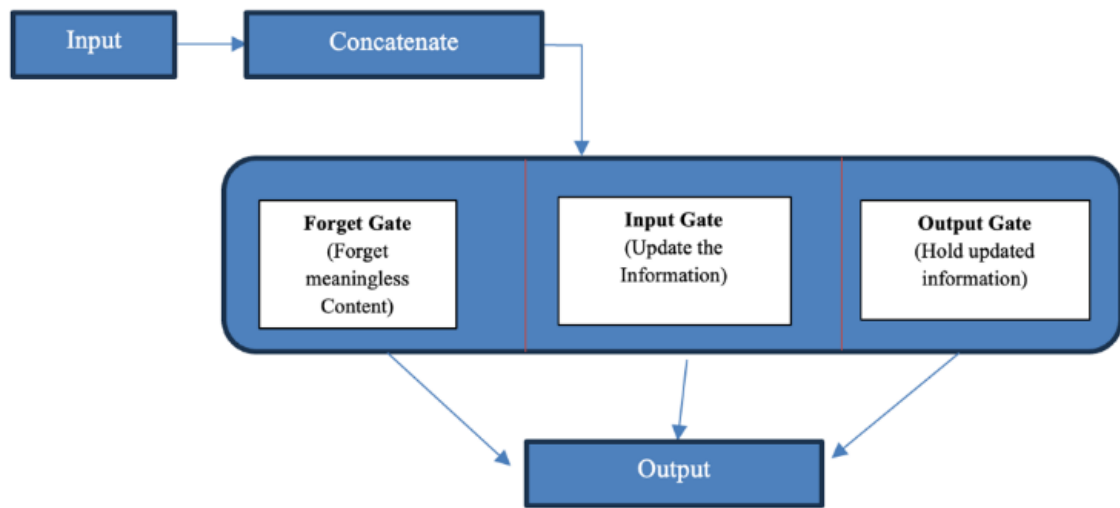


Fig. 2. LSTM Architecture.

In an LSTM cell, input, output, and forget gates are coupled. (See Fig.2). Which data should be deleted from the cell state is determined by the forget gate. It receives as inputs the current input and the prior cell state and outputs a forgetting factor. Input Gate picks the fresh information kept in the cell state. It uses the previous cell state and the current input to determine likely new values that could be added to the cell state. What information should be shown as the output is decided by the output gate. To provide the output for the current time step, it takes into account the current input and the updated cell state.

In an LSTM cell, a cell state c_t is employed to retain information from previous time steps. This information is regulated by the: Forget gate unit f_t which is responsible for carrying some of the portion of past information that is to be retained from c_{t-1} , Input gate unit i_t which incorporates the quantity of new information c'_t and Output gate unit o_t which triggers the activation of the new cell state c_t to generate the hidden state h_t . In each time step the following calculations occur:

$$f_t = \sigma(U_f x_t + W_f h_{t-1} + b_f) \quad (1)$$

$$i_t = \sigma(U_i x_t + W_i h_{t-1} + b_i) \quad (2)$$

$$o_t = \sigma(U_o x_t + W_o h_{t-1} + b_o) \quad (3)$$

$$c'_t = \varphi(U_c x_t + W_c h_{t-1} + b_c) \quad (4)$$

$$c_t = f_t \odot c_{t-1} + i_t \odot c'_t \quad (5)$$

$$h_t = o_t \odot \Phi(c_t) \quad (6)$$

where, $\odot$ represents the element wise vector product, σ denotes the sigmoid function, Φ represents an activation function.

Here, in Output Layer the Dense Layer produces the final rainfall prediction for each time step. Further the model is trained on the historical rainfall data, adjusting hyper parameters as needed. Then this trained model is used to predict future rainfall based on new input sequences [63]. The LSTM Encoder-Decoder structure is particularly beneficial in handling the sequence-to-sequence nature of rainfall prediction. It allows the model to consider historical rainfall patterns, learn temporal dependencies, and make accurate predictions for future time steps [64] (See Fig. 3). LSTMs have the advantage of being able to capture long-term dependencies, which makes them ideal for time series data tasks like rainfall prediction. The difficulties posed by different input and output sequence lengths in rainfall forecasting are successfully addressed by this architecture. [65] (See Fig.4-8). In figure 4, one can see as the time steps are increasing, the corresponding rainfall start showing a fixed kind of pattern. Figure 5 describes the correlation between the actual and predicted rainfall, as a color changes from blue to red indicates the low correlation to high correlation. Figure 6 explains, as the frequency becomes low, prediction error becomes high. Figure 7 explains that a will fit models have small residuals for all data points. Figure 8 explains that the variance is regularly distributed.

4.3 Performance Evaluation Metrics

Performance metrics are essential tools used to evaluate the accuracy and efficacy of predictive models. These metrics furnish quantitative measures for evaluating the model's predictive capabilities. The research presents the performance metrics specifically for the LSTM encoder-decoder model. For evaluating the performance of the model, the various parameters are involved in this model

such as Accuracy, RMSE (root mean squared error), MSE (mean squared error), Precision and MCC (Matthews correlation coefficient). These parameters give the models predictive capability:

Mean Squared Error: A loss function called MSE (Mean Squared Error) is applied to regression issues. Its function in this situation is to calculate the average squared difference. between the predicted values (y_b) and the actual values (y_a) in the dataset.

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_a - y_b)^2 \quad (7)$$

In this formula, n represents the total number of samples, y_a denotes the observed (actual) value of i^{th} sample and y_b signifies the predicted value for the i^{th} sample.

Root Mean Squared Error: The square root of Mean Squared Error is known as RMSE. It calculates the residuals' standard deviation.

Accuracy: Accuracy is a measure that represents the proportion of accurately predicted instances to the total number of instances in the dataset. It is frequently used for evaluating performance in classification problems.

$$\text{Accuracy} = \frac{\text{Number of correct predictions}}{\text{Total number of predictions}} \quad (8)$$

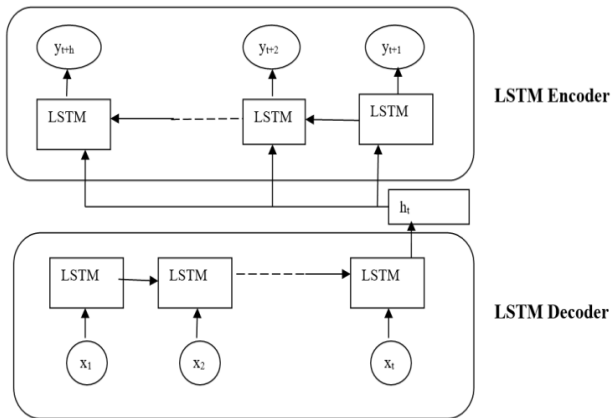


Fig. 3. LSTM Encoder-Decoder Model.

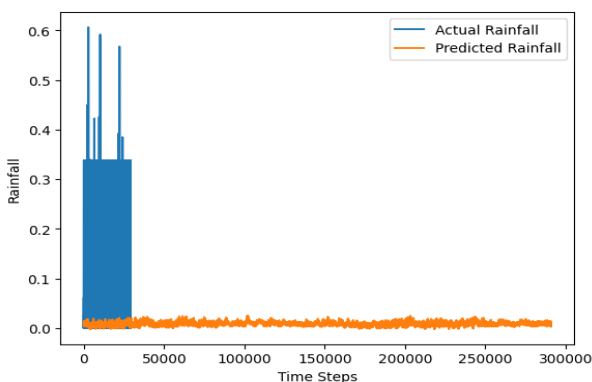


Fig. 4. Time Series Plot for rainfall.

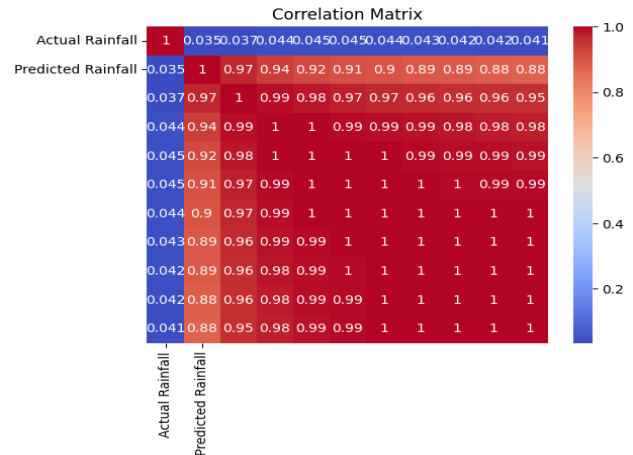


Fig. 5. Correlation matrix for actual and predicted rainfall.

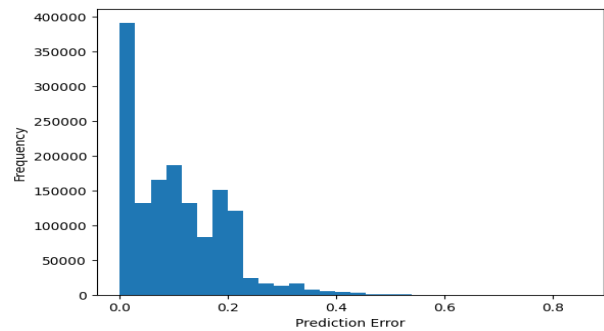


Fig. 6. Histogram between the frequency and the predicted error for the given dataset.

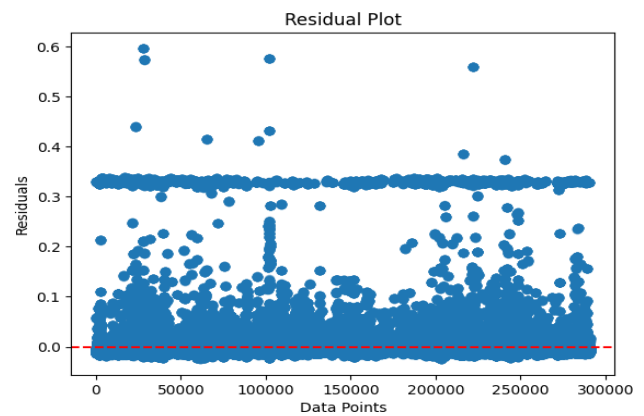


Fig. 7. Residual Plot.

Precision: By calculating the proportion of accurately predicted positive observations to all predicted positives, precision evaluates the accuracy of positive predictions. A high value of precision indicates a lower rate of false positives.

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}} \quad (9)$$

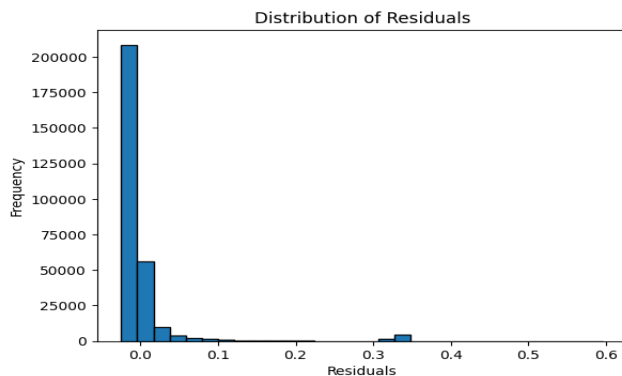


Fig. 8. Histogram plotted between Frequency and Residuals.

Matthews Correlation Coefficient (MCC): By taking into account both true and false positives as well as negatives, MCC acts as a balanced metric. Its utility is especially pronounced when handling imbalanced datasets.

5. RESULTS AND COMPARATIVE ANALYSIS

The LSTM encoder and decoder model's flawless integration has produced rainfall pattern predictions with previously unheard-of accuracy, setting a new benchmark for forecasting precision. In order to assess the proficiency and precision of the model's predictions, quantitative metrics are chosen and applied in the context of the LSTM Encoder-Decoder model for rainfall prediction. Three assessment measures are used in this research to evaluate the LSTM-ED model's efficacy. The recurrent structure of the LSTM-ED model is complex. Using an optimizer such as Adam, which is well-known for its first-order training properties, reduces the complexity of the training procedure and makes model training easier overall. The ReLU (Rectified Linear Unit) activation function was essential to our modeling strategy for maximizing our neural network's performance and training effectiveness. This paper offered a non-linearity that allowed the network to comprehend intricate patterns and correlations within the data by using ReLU as the activation function in this model neural network design. The key characteristic of ReLU is its ability to output the input directly if it is positive, and zero otherwise. This introduces a desirable sparsity in the network, promoting faster convergence during training. The LSTM encoder and decoder model employed in our study demonstrated exceptional accuracy, particularly at a critical juncture during the training process: at epochs with a value of 100 and for larger values of epochs it started to decrease and then became constant.

For comparative analysis, this paper involved working in the various models and comparing their Accuracy, Precision and MCC values. The study involved, statistical as well as neural network models such as Naïve Bayes, SVM, Decision Tree, KNN, Xgboost, ANN, CNN, RNN, DNN, LSTM, LSTM+RNN, and the proposed model combining LSTM with the encoder and decoder (LSTM-ED). Here, it is found

that the LSTM-ED model achieves an Accuracy= 92.13%, Precision 0.7282 and MCC 0.6545 the prediction of rainfall on the given dataset (See Table 2).

Here, in this paper presents the evaluation temporal patterns, comparing predictions with actual values, assessing variable importance, conducting threshold analyses. The model worked with the binary cross entropy and the ADAM optimizer function, apt for classification task. The model has been regularized and validated using early stopping.

Table 2. Comparative analysis of different models for rainfall prediction

Methods	Accuracy (in %)	Precision	MCC
SVM	83.5354	0.7438	0.4089
KNN	78.9858	0.5351	0.3434
Decision Tree	77.6143	0.4814	0.3524
Naïve Bayes	72.9883	0.4319	0.3881
ANN	84.3672	0.6645	0.5263
Xgboost	85.2734	0.7317	0.5448
RNN	82.7329	0.7707	0.3477
CNN	77.9321	0.5241	0.3989
LSTM	77.5926	0.6338	0.3478
LSTM Encoder-Decoder	92.13812	0.7282	0.6545
DNN	85.7763	0.7506	0.5494
RNN+LSTM	86.08	0.6578	0.3456

On performing the evaluation and running the model on the different values of the epochs for determining the best accuracy score of the model, the values are taken starting from epochs=50 (Accuracy=92.1270%), then gradually decreased and the best accuracy is achieved at epochs=10 (Accuracy=92.13812%) i.e., it reached its maximum and then started decreasing as further epochs are increased for instance at epochs=150 the accuracy started decreasing. (See Table 3.).

Table 3. Accuracy for different number epochs with patience = 3

Epochs	10	20	50
Accuracy	92.1381	92.1276	92.1270

Here, the EarlyStopping callback is used to validate the model after every epoch. Its hyperparameters include

monitor='val_loss', which monitors the validation loss and stops training if the validation loss does not improve after a predetermined number of epochs (patience=5). When restore_best_weights=True is selected, the model weights are returned to the lowest validation loss. (See Fig. 9.).

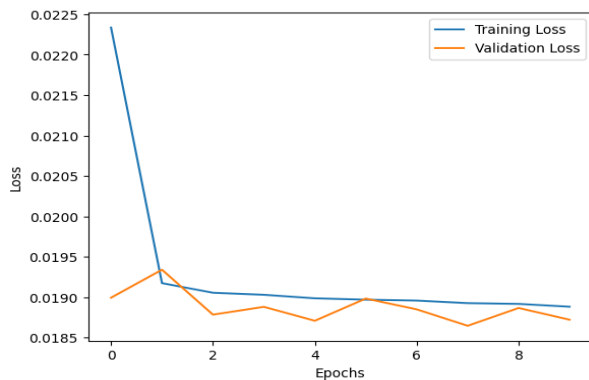


Fig. 9. Validation loss and Training loss with increase in value of epochs.

The learning rate is crucial for achieving accurate and efficient training in the LSTM-ED model, and thus the used hyper parameter is involved in finding the best accuracy for the given model. The accuracy of the LSTM-ED has been evaluated for different values of the learning rate starting from learning rate = 0.1 to 0.9. On performing the evaluation, it is found that the highest Accuracy 92.13% is achieved at learning rate= 0.7 (See Table 4.) (See Fig. 10.).

Table 4. Calculation of accuracy for changing values of learning rate

Learning Rate	Accuracy
0.1	90.6497
0.2	86.5012
0.3	84.6319
0.4	81.4861
0.5	86.0479
0.6	86.2002
0.7	92.13812
0.8	89.656
0.9	87.6389

The dataset included various parameters such as the wind speed at different timings, temperature, humidity at different timings, precipitation, sunshine, and so on. While evaluating the model, in order to find the optimized accuracy, the combination out of these given parameters has been used and the model was implemented on these combinations. The

best result was concluded when parameters including rainfall, sunshine, maximum temperature, minimum temperature and evaporation were used as selected features. Due to their high correlation with the optimal results, the authors applied a principal component analysis using the combination of these features in the proposed model.

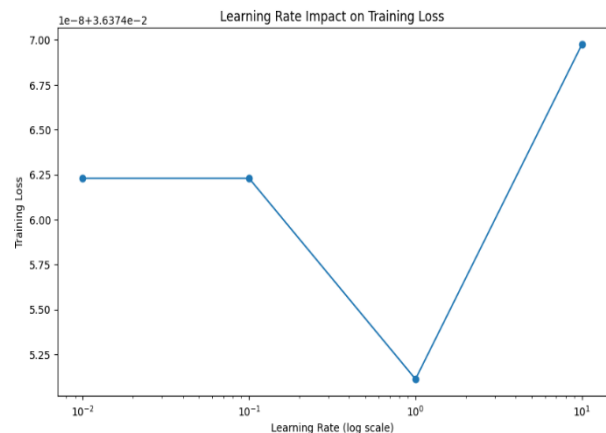


Fig. 10. The impact of learning rate on Training loss.

Here, it is find that that the best accuracy for the proposed model has been evaluated to be 92.138% on 10 number of epochs and learning rate is 0.7 with early stopping callback used for validation.

The goal of training was to minimize the loss . Monitor (quantity to be measured ,defaults to “value_loss”) and min_data (minimum changes in the measured quantity to consided as improvement)were used as early stopping callback parameters.

6. CONCLUSION

In order to accurately analyze and predict rainfall using meteorological data, this paper has employed a variety of machine learning and deep learning techniques. The application of the suggested LSTM encoder-decoder model for rainfall prediction has been discovered to represent a noteworthy development in weather forecasting. Here, the suggested LSTM Encoder-Decoder method has proved successful. 92.13% accuracy. Here, a comparative examination using machine learning demonstrates that the suggested method outperforms alternative methods for rainfall prediction. The proposed approach maintained its accuracy across additional epochs, showcasing a decrease in loss, root mean square error (RMSE), and learning rate. The capacity of the proposed models to analyze complex temporal patterns in meteorological data has displayed encouraging outcomes in reliably forecasting rainfall patterns over extended periods. Thus, the model's capability to glean insights from historical data and generate dependable predictions contributes to better preparedness for extreme weather events, aiding in effective risk management, agriculture planning, water resource

management, and disaster mitigation strategies. This research can be further extended to predict multivariate predictions with transfer learning, and can be integrated with several ensemble models to achieve higher forecasting efficiency.

REFERENCE

- [1] Hernández, E., et al. 2016. Rainfall prediction: A deep learning approach. *Hybrid Artificial Intelligent Systems: 11th International Conference, HAIS 2016, Seville, Spain, April 18–20, 2016, Proceedings 11*. Springer International Publishing.
- [2] Datta, A., Si, S., and Biswas, S. 2020. Complete statistical analysis to weather forecasting. *Computational Intelligence in Pattern Recognition: Proceedings of CIPR 2019*. Springer Singapore.
- [3] Mohammed, M., et al. 2020. Prediction of rainfall using machine learning techniques. *International Journal of Scientific and Technology Research*, 9(1): 3236–3240.
- [4] Jolliffe, I. T., and Stephenson, D. B. (Eds.). 2012. *Forecast verification: A practitioner's guide in atmospheric science*. John Wiley & Sons.
- [5] Tanessong, R. S., et al. 2017. Bayesian processor of output for probabilistic quantitative precipitation forecast over Central and West Africa. *Atmospheric and Climate Sciences*, 7(3): 263–286.
- [6] Czarnecka, M., and Nidzgorska-Lencewicz, J. 2011. Impact of weather conditions on winter and summer air quality. *International Agrophysics*, 25(1).
- [7] Sharma, N., et al. 2011. Predicting solar generation from weather forecasts using machine learning. *2011 IEEE International Conference on Smart Grid Communications (SmartGridComm)*. IEEE.
- [8] Bezak, N., Šraj, M., and Mikoš, M. 2016. Copula-based IDF curves and empirical rainfall thresholds for flash floods and rainfall-induced landslides. *Journal of Hydrology*, 541: 272–284.
- [9] Deo, R. C., et al. 2018. Drought prediction with standardized precipitation and evapotranspiration index and support vector regression models. In *Integrating Disaster Science and Management*. Elsevier, 151–174.
- [10] Abedini, M., et al. 2019. A novel hybrid approach of Bayesian logistic regression and its ensembles for landslide susceptibility assessment. *Geocarto International*, 34(13): 1427–1457.
- [11] Senior, C. A., et al. 2002. Predictions of extreme precipitation and sea-level rise under climate change. *Philosophical Transactions of the Royal Society A*, 360(1796): 1301–1311.
- [12] Kwon, Y., Kwasinski, A., and Kwasinski, A. 2019. Solar irradiance forecast using naïve Bayes classifier based on publicly available weather forecasting variables. *Energies*, 12(8): 1529.
- [13] Radhika, Y., and Shashi, M. 2009. Atmospheric temperature prediction using support vector machines. *International Journal of Computer Theory and Engineering*, 1(1): 55.
- [14] Xu, W., Ning, L., and Luo, Y. 2020. Wind speed forecast based on post-processing of numerical weather predictions using gradient boosting decision tree algorithm. *Atmosphere*, 11(7): 738.
- [15] Findawati, Y., et al. 2019. Comparative analysis of Naïve Bayes, K Nearest Neighbor and C.45 method in weather forecast. *Journal of Physics: Conference Series*, 1402(6). IOP Publishing.
- [16] Kareem, S., Hamad, Z. J., and Askar, S. 2021. An evaluation of CNN and ANN in weather forecasting: A review. *Sustainable Engineering and Innovation*, 3(2): 148–159.
- [17] Biswas, S. K., et al. 2014. Weather prediction by recurrent neural network dynamics. *International Journal of Intelligent Engineering Informatics*, 2(2–3): 166–180.
- [18] Sen, S., et al. 2021. Analysis of PCA-based AdaBoost machine learning model for mid-term weather forecasting. *Computational Intelligence and Machine Learning*, 2(2): 41–52.
- [19] Patkar, D. U., et al. 2022. Machine learning for weather forecasting using freely available weather data in Python. *GIS Science Journal*, 8(12).
- [20] Solomon, W. 2023. A comparative analysis of the use of deep learning and machine learning in weather forecasting: Using meteorological dataset on Vaasa.
- [21] Salman, A. G., et al. 2018. Single layer and multi-layer long short-term memory (LSTM) model with intermediate variables for weather forecasting. *Procedia Computer Science*, 135: 89–98.
- [22] Li, X., Kou, K., and Zhao, B. 2021. Weather GAN: Multi-domain weather translation using generative adversarial networks. *arXiv preprint arXiv:2103.05422*.
- [23] Chang, L., et al. 2019. Feature extraction and continuous debugging training of the LSTM model. <https://doi.org/10.1002/met.1852>.
- [24] Abhishek, K., et al. 2012. Weather forecasting model using artificial neural network. *Procedia Technology*, 4: 311–318.
- [25] Hardwinarto, S., and Aipassa, M. 2015. Rainfall monthly prediction based on artificial neural network: A case study in Tenggarrong Station, East Kalimantan–Indonesia. *Procedia Computer Science*, 59: 142–151.
- [26] Jan, Z., et al. 2009. Seasonal to inter-annual climate prediction using data mining KNN technique. In *Wireless Networks, Information Processing and Systems: IMTIC 2008, Jamshoro, Pakistan, April 11–12, 2008, Revised Selected Papers*. Springer Berlin Heidelberg.
- [27] Hu, J., et al. 2013. EMD-KNN model for annual average rainfall forecasting. *Journal of Hydrologic Engineering*, 18(11): 1450–1457.
- [28] Huang, M., et al. 2017. A novel approach for precipitation forecast via improved K-nearest neighbor algorithm. *Advanced Engineering Informatics*, 33: 89–95.
- [29] Li, T., Ren, Q., and Qiu, Y. 2019. Application of improved naïve Bayesian–CNN classification algorithm in sandstorm prediction in Inner Mongolia. *Advances in Meteorology*, 2019: 1–13.
- [30] Chen, W., et al. 2020. Modeling flood susceptibility using naïve Bayes tree, alternating decision tree, and random forest methods. *Science of The Total Environment*, 701: 134979.
- [31] Azmi, A. U., et al. 2021. Naïve Bayes methods for rainfall prediction classification in Banyuwangi. *Journal of Physics: Conference Series*, 1872(1). IOP Publishing.
- [32] Han, J. M., Ang, Y. Q., Malkawi, A., and Samuelson, H. W. 2021. Using recurrent neural networks for localized weather prediction with public airport data and on-site measurements. *Building and Environment*, 192: 107601.

- [33] Hewage, P., Behera, A., Trovati, M., and Pereira, E. 2019. Long-short term memory for an effective short-term weather forecasting model using surface weather data. https://doi.org/10.1007/978-3-030-19823-7_32.
- [34] Mittal, S., and Sangwan, O. P. 2020. Big data analytics using deep LSTM networks: A case study for weather prediction. <https://dx.doi.org/10.25046/aj050217>.
- [35] Kumar, P., et al. 2021. Micro-climate prediction: Multi-scale encoder-decoder based deep learning framework. <https://doi.org/10.1145/3447548.3467173>.
- [36] Schultz, M. G., et al. 2021. Can deep learning beat numerical weather prediction? *Philosophical Transactions of the Royal Society A*, 379(2194): 20200097.
- [37] Wu, Y., et al. 2018. Context-aware attention LSTM network for flood prediction.
- [38] Suleman, M. A. R., and Shridevi, S. 2022. Short-term weather forecasting using spatial feature attention-based LSTM model. *IEEE Access*, 10: 82456–82468. DOI: 10.1109/ACCESS.2022.3196381.
- [39] Chattopadhyay, A., et al. 2022. Towards physics-inspired data-driven weather forecasting: Integrating data assimilation with a deep spatial-transformer-based U-NET. *Geoscientific Model Development*, 15(5): 2221–2237.
- [40] Alerskans, E., et al. 2022. A transformer neural network for predicting near-surface temperature. *Meteorological Applications*, 29(5): e2098. DOI: 10.1002/met.2098.
- [41] Hou, X., Ju, C., and Wang, B. 2023. Prediction of solar irradiance using convolutional neural network and attention mechanism-based LSTM. *Heliyon*, 9(11): e21484.
- [42] Chamatidis, I., Tzanes, G., Istrati, D., and Nikos, D. 2023. Short-term forecasting of rainfall using sequentially deep LSTM networks. <https://doi.org/10.3390/environsciproc2023026157>.
- [43] Sunkavalli, A., et al. 2023. Weather prediction using LSTM. *European Chemical Bulletin*. <https://www.eurchembull.com/uploads/paper/1670a57f18202ee44ce5b6602f83e6c6.pdf>.
- [44] Sigg, C., et al. 2023. Photographic visualization of weather forecasts with generative adversarial networks. *Artificial Intelligence for the Earth Systems*, 2(1): e220028.
- [45] Sena, I. G. W., Dillak, J. W., Leunupun, P., and Santoso, A. J. 2020. Predicting rainfall intensity using Naïve Bayes and information gain methods. *Journal of Physics: Conference Series*, 1577(1): 012011.
- [46] Ali, A., Khairan, A., Tempola, F., and Fuad, A. 2021. Application of Naïve Bayes to predict the potential of rain in Ternate City.
- [47] Patel, H. H., and Prajapati, P. 2018. Study and analysis of decision tree-based classification algorithms. *International Journal of Computer Sciences and Engineering*, 6(10): 74–78.
- [48] Fiarni, C., Sipayung, E. M., and Tumundo, P. B. 2019. Academic decision support system for choosing information systems sub-majors using decision tree algorithm. *Journal of Information Systems Engineering and Business Intelligence*, 5(1): 57.
- [49] Kumar, V., Yadav, V. K., and Dubey, S. 2022. Rainfall prediction using machine learning. *International Journal for Research in Applied Science and Engineering Technology*, 10: 42876.
- [50] Zhang, S., et al. 2018. A novel kNN algorithm with data-driven k parameter computation. *Pattern Recognition Letters*, 109: 44–54.
- [51] Pandey, A., and Jain, A. 2017. Comparative analysis of KNN algorithm using various normalization techniques. *International Journal of Computer Network and Information Security*, 9: 36.
- [52] Deng, Z., et al. 2016. Efficient kNN classification algorithm for big data. *Neurocomputing*, 195: 143–148.
- [53] Anwar, M. T., Winarno, E., Hadikurniawati, W., and Novita, M. 2021. Rainfall prediction using extreme gradient boosting. *Journal of Physics: Conference Series*, 1869(1): 01207.
- [54] Kumar, V. P., et al. 2023. Real-time rainfall prediction for Indian states using XGBoost and random forest approach.
- [55] Hussain, S. A., and Fernandez, T. F. 2023. Early prediction of rainfall using XGBoost algorithm in comparison with logistic regression.
- [56] Bui, D. L., et al. 2024. Evaluating an effectiveness of a solar power plant output forecasting model based on LSTM method using validation in different seasons of a year in Vietnam. *GMSARN International Journal*, 18: 114–122.
- [57] Salman, A. G., Heryadi, Y., Abdurahman, E., and Suparta, W. 2018. Single-layer and multi-layer LSTM model with intermediate variables for weather forecasting. *Procedia Computer Science*, 135: 89–98.
- [58] Salehin, I., et al. 2020. Artificial intelligence-based rainfall prediction using LSTM and neural network. *IEEE International Women in Engineering (WIE) Conference on Electrical and Computer Engineering (WIECON-ECE)*, 5–8. IEEE.
- [59] Han, Q. H. Giap, et al. 2025. Toward enhancing mid-term load forecasting: RNN-based models vs transformer-based models. *GMSARN International Journal*, 19: 397–404.
- [60] Poornima, S., and Pushpalatha, M. 2019. Prediction of rainfall using intensified LSTM-based recurrent neural network with weighted linear units. *Atmosphere*, 10(11): 668.
- [61] Priatna, M. A., and Djamal, E. C. 2020. Precipitation prediction using recurrent neural networks and long short-term memory. *Telkomnika (Telecommunication Computing Electronics and Control)*, 18(5): 2525–2532.
- [62] Zhang, Y., Ragettli, S., Molnar, P., Fink, O., and Peleg, N. 2022. Generalization of an encoder-decoder LSTM model for flood prediction in ungauged catchments. *Journal of Hydrology*, 614: 128577.
- [63] Kumar, P., et al. 2021. Micro-climate prediction: Multi-scale encoder-decoder based deep learning framework. *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, 3128–3138.
- [64] Khafaga, D. S., Ibrahim, A., Towfek, S. K., and Khodadadi, N. 2023. Data mining techniques in predictive medicine: An application in hemodynamic prediction for abdominal aortic aneurysm disease. *Journal of Artificial Intelligence and Metaheuristics*, 5(1): 29–37.
- [65] Zhang, Y., Ragettli, S., Molnar, P., Fink, O., and Peleg, N. 2022. Generalization of an encoder-decoder LSTM model for flood prediction in ungauged catchments. *Journal of Hydrology*, 614: 128577.