A Fine-Grained, Multi-Model System to Predict Rainfall Across Diverse Datasets

Shubham Gade[1], Alfiya Shahbad[2], Pradnya Randive[3], Chanchal Vakte[4]

[1]AI/ML Engineer, AI/ML Department, EnerSys, PA, USA
[2]Asst. Professor, Computer Engineering, MESWCOE, Pune, India
[3]Asst. Professor, Computer Engineering, Dr. D. Y. Patil Institute of Technology, Pune, India
[4]Asst. Professor, Computer Engineering, MESWCOE, Pune, India

Abstract

Rainfall is crucial to agriculture and a farmer's means of subsistence. It has been very helpful to farmers, especially in the agricultural society of the Indian Subcontinent, where it is heavily used for agricultural production, ground water replenishment, and irrigation. Rainfall is therefore a vital and important event for a sizable section of the Indian population. Therefore, rainfall prediction is by far the most effective and advantageous method that has the power to drastically improve the lives of a great number of people. The process of estimating the probability of rainfall at a given place, forecasting future precipitation, and figuring out how much rain will fall in particular locations is known as rainfall prediction. Along with the probability of rainfall at that specific place, it takes into account the precipitation volume assessment, forecast accuracy, and prediction error. Consequently, an efficient method for rainfall prediction is presented in this research effort. The proposed method uses Artificial neural network (ANN) and Long short term memory (LSTM) model to estimate the rainfall prediction. This research used many datasets from the Indian subcontinent and other continent-wide datasets to estimate the rainfall. Results are measured for the Root mean square error (RMSE) parameter on a variety of datasets, demonstrating how well the system performs in comparison to some previous efforts.

Keywords: Rainfall prediction, Deep Learning models, Artificial Neural network, long short term memory.

1. Introduction

Because ocean water is saline, most land-dwelling creatures and plants cannot reach the vast majority of the earth's surface, which is covered in water. Rain is therefore necessary for life to exist on Earth. Rainfall supplies salt-free water necessary for ground-based life. Accurate weather forecasting is essential to our everyday activities. Farmers want information to help them get ready for planting and harvesting crops. Weather forecasting helps us avoid hurricanes and droughts, reduces agricultural loss, and helps us make important decisions. Currently, forecasting the weather involves the use of mathematical models, trend and pattern recognition, and observation. An organic phenomenon, precipitation is the consequence of multiple complex weather systems interacting with one another.One of the most challenging aspects of developing a rainfall prediction model is the significant unpredictability involved in determining the contribution of natural occurrences. It is highly challenging to model weather forecasting due to the intricate nonlinear structure of the relevant meteorological systems. Moisture content, wind direction, wind speed, cloud cover, temperature, and other factors are some of the factors that significantly affect the frequency of rainfall.

Forecasting rainfall is crucial information for irrigation, agriculture, and water security. It also serves a significant purpose. It is also crucial for emergency preparedness in the event that heavy rain causes landslides. Models for predicting rainfall can help save lives and sustain livelihoods, which has a positive implicit economic impact on the country. In order to provide accurate and reliable forecasting, forecast methodologies have been developed and implemented using statistical modeling and regression techniques.Rain or precipitation prediction has been sought for since antiquity and has always attracted the interest of a wide range of stakeholders, including the research community, the farming industry, transportation agencies, and policymakers. This is true whether it is a matter of academic study or practical analysis. A precise rainfall forecast provides crucial information for policy makers to warn of approaching natural disasters and aids in human planning for outdoor pleasure. In government organizations, rainfall forecasting—like short-term forecasting—is a big problem.

In the domains of computing and meteorology, developing a reliable rainfall prediction system that generates accurate forecasts is a challenging ongoing research problem. The collection of measurable data that is both broad and of high quality is essential to the study of rainfall prediction.Since weather is so important to human existence, weather forecasting is crucial throughout the Indian subcontinent. The challenging task for the meteorological organization is to predict the unpredictable regularity of precipitation. It's challenging to accurately predict rainfall in a changing environment. Rainfall is unpredictable in both the warm and rainy seasons. Numerous methods have been developed to estimate rain fall by researchers worldwide; the majority of these methods employ random values and are equivalent to meteorological data.

[1] S. Poornima et al. Depending on the area, the current years' weather unpredictability causes either droughts or floods. Both lives and property may be lost as a result of these natural disasters. They also have a variety of other effects on agriculture, including crop failure, agricultural stress, and a lack of water. Managing the nation's food supply for its citizens becomes increasingly difficult as a result of this fall in food production. Planning water and farm management in agriculture so requires forecasting weather, particularly rainfall. By predicting floods and droughts, necessary precautions can be made in light of the future possibilities. SPI is used to estimate drought, and LSTM is used to anticipate monthly long-term rainfall. The LSTM's performance metrics, including RMSE, loss, and learning rate, are suitable for prediction at 0.059, 0.0036, and 10^{-5}, respectively.

[2] Gomathy, C. K., et al. The focus of this project was rainfall estimation, and it is predicted that SVR is a useful and flexible method that may assist the client manage the challenges associated with the geometry of the data, the typical problem of model overfitting, and the distributional features of fundamental components. For SVR display, the bit capacity choice is fundamental. For both direct and non-straight relationships, we advise tenderfoots to use both straight and RBF pieces separately. It is evident that SVR performs better as an expectation technique than MLR. When a data collection has non-linearity, SVR becomes useful because MLR is unable to detect it. In order to evaluate the models' execution, we also process Mean Absolute Error (MAE) for the MLR and SVR models. Finally, we examine the SLR, SVR, and tuned SVR model presentations. The tuned SVR model provides the best expectation, true to form.

[3] Appiah-Badu, Nana K. A. et al. Using five (5) classification algorithms—Decision Tree, Random Forest, Multilayer Perceptron, Extreme Gradient Boosting, and KNearest Neighbour—this study

predicted rainfall in Ghana across all ecological zones. This study made use of 41 years of historical climate data from the Ghana Meteorological Service, covering the years 1980–2019. Precision, f1-score, and recall were the assessment measures used to assess the classifiers' performance; the results are shown in tables. Additionally, the execution times of each model and the model's overall accuracy were determined; the outcomes are displayed in figures. In order to guarantee accurate rainfall prediction, the input datasets underwent exploratory data analysis. Outliers were eliminated from the datasets and standardized prior to the classification stage, and missing data was replaced using the multiple imputation by chained equations approach. Two categories of datasets were created: training datasets and testing datasets. To examine the effectiveness of the classification algorithms on various training and testing ratios, we used three distinct types of training and testing ratios (training data: testing data): 70:30, 80:20, and 90:10. The study's conclusions demonstrated unique traits for the rain and no-rain classes' classification in each of the nation's ecological zones. No, classifiers in the coastal zone performed a better job classifying the rain class than they did the rain class. In the woodland zone, there was, however, the opposite reaction. Regarding the rain class, the classifiers' performance was best in the woodland zone. All classifiers on the three training and testing ratios in the savannah zone did a good job of classifying the no-rain class, which is consistent with the region's low rainfall trend. Decision trees stood up as the model with the fastest execution time across all ecological zones and when all training and testing ratios were combined, but multilayer perceptrons fared poorly. In general, multilayer perceptrons, random forests, and extreme gradient boosting all fared well in all cases, suggesting that ensemble and deep learning models are viable options for rainfall prediction. However, K-Nearest Neighbor underperformed on all training and testing ratios in all zones, which calls for additional research. Further research is being considered to predict rainfall in all of Ghana's ecological zones using other classification algorithms and a hybrid model at various training and testing ratios.

This paper's second section reviews earlier research on models for predicting Rainfall. Section 3 provides a detailed explanation of the deployed approach. Section 4 is utilized to assess the results acquired, and Section 5 concludes this study.

2. Literature Survey

[4] Poornima S. et al. Using the provided dataset, which contained roughly 12,000 data points, ARIMA and LSTM models were used. While LSTM uses a multivariate method, ARIMA uses a univariate approach. The two factors that had a positive association with the two indices but were not taken into account in the SPI and SPEI calculations were temperature and humidity. As a result, they were incorporated into the LSTM neural network's multivariate method. It can be inferred that for forecasts made over longer timescales, such as six and twelve months, the long short term memory model has performed better than the ARIMA model. Additionally, it was shown that adding temperature and relative humidity—two additional variables that exhibited a positive association with both SPI and SPEI—to the training set improved LSTM performance. On the other hand, LSTM calculation requires more resources than ARIMA modeling. Additionally, it can be seen that ARIMA was computationally lighter than LSTM networks and offered a strong short-term prediction answer. Future research in the field can make use of a larger dataset, which could produce better results, as well as ensemble deep learning techniques, which have a higher likelihood of capturing the subtleties of a challenging pattern like drought. Depending on the overall precipitation cycle, the linked factors may vary at particular locations. El Nino and other weather phenomena might also need to be taken into account.

[5] Muluken Chalachew Liyew and others. The application of data science and machine learning to forecast the state of the atmosphere is called rainfall prediction. For the efficient use of water resources and agricultural production to lower food-related and rain-related disease mortality, it is critical to forecast the intensity of the rainfall. This study examined many machine learning techniques for predicting rainfall. A variety of tree machine learning algorithms, including MLR, FR, and XGBoost, were demonstrated and evaluated with data gathered from the Bahir Dar City meteorological station in Ethiopia. The Pearson correlation coefficient was used to identify environmental characteristics that were pertinent for rainfall prediction. The machine learning model employed in this work used the chosen characteristics as input variables. When the three machine learning algorithms (MLR, RF, and XGBoost) were compared, it became clear that, when it came to predicting the quantity of rainfall that would occur each day based on specific environmental factors, XGBoost performed better. If sensor data is included in the study, the rainfall amount estimate may become more accurate. However, this study did not take sensor data into account. By utilizing sensor and meteorological datasets with more diverse environmental parameters, the accuracy of the rainfall prediction can be increased. Therefore, if sensor and meteorological datasets are employed for the daily rainfall amount prediction study, big data analysis can be applied for rainfall prediction in future work.

[6] Poornima S. et al. The primary conclusions drawn from this research are as follows: (1) the annual rainfall level in India is declining at a rate of 0.04% annually; (2) the nation has been experiencing mild to moderate drought for the past 50 years, particularly from 1980 onwards; (3) mild drought may affect the country in the upcoming years (2021 to 2027); and (4) certain regions may experience moderate to severe drought in the future. The country's actual precipitation and drought situation for 2021 are in line with the work's forecast. The precipitation projection for 2021 is 99.46% accurate. This work estimates the rainfall for future periods, which is used to determine the drought for next seasons, in contrast to other studies that do drought analysis using the rainfall data that is now available. The government may be able to implement required measures, such rainwater collection, irrigation, and other farm management practices, with the aid of insights gleaned from the development of data analytics. These measures will lessen the likelihood that food production will fall. Future research will concentrate on applying the enhanced LSTM to predict rainfall in India and other regions, as well as adding soil properties to the forecast algorithm for drought situations so that the best crops may be suggested for the upcoming seasons. Incorporating soil properties into Internet of Things (IoT) technology will lead to a major improvement in rainfall forecasting and drought estimate. Eventually, a mobile application with the analysis results will be made available to farmers so they may utilize it to select which crops to plant for the next growing season.

[7] T. A method for predicting rainfall in Sri Lanka's Badulla district is proposed by Dananjali et al. To that purpose, the rainfall data gathered from Badulla District was used to train three data mining algorithms. The M5P model tree outperformed the linear regression and SMO regression models, according to the evidence. In both the training and testing phases, the M5P model exhibited significantly lower MAE, RMSE, RRSE, RAE, and higher DA values. Between actual and forecast rainfall values, only the M5P model has a positive correlation of 0.41; the other two models do not display any association at all. Moreover, the M5P gives the error distribution more unpredictability. As a result, the M5P model tree is suggested as the most effective model for predicting weekly rainfall in the Badulla district. In summary, the M5P model tree can reasonably accurately predict the rainfall in Badulla District for the next six months. This research is helpful for a variety of industries, especially the

Badulla region's agricultural industry.[8] Liu Huang et al. noted that overall accuracy is increased and that the relative error of the combined model's prediction results typically exhibits a declining trend when compared to those of a single SVR model or RBF model. It demonstrates how the combination approach can use the benefits of each individual model to produce the best possible forecast outcomes. A statistical analysis is performed on the relative error of the prediction outcomes of the three models mentioned above. The average relative error of the combined model is 18.21%, showing that the combined model may realize a better overall forecasting impact. The average relative error of the SVR model is 28.02%, the average relative error of the RBF model is 37.26%.

[9] Joshi Yogesh Kumar et al. This article uses bar and line graphs to visualize data and illustrate all the different trends in rainfall across the States and Union Territories during the last century. This document presents the annual rainfall as well as the top ten highest and lowest rainfall in the States and Union Territories of India, as well as the highest and lowest rainfall in the West Rajasthan and Coastal Karnataka regions. There has been discussion on the areas of India with the most and least rainfall. The data visualization findings indicate a negligible downward trend in the yearly average rainfall in the West Rajasthan and Coastal Karnataka regions. It is discussed how the West Rajasthan region is experiencing a protracted drought and how the coastal Karnataka region is experiencing extremely damaging rainfall. Data visualization has significantly increased data understandability and reduced data complexity. It will be possible to determine other features of rainfall through future research. Regression analysis using the findings of this work can be used to predict rainfall, which has significant agricultural applications.

[10] Zne-Jung Lee et al. suggested an innovative way to predict landslide rainfall. The platform in the suggested approach is built on Apache Spark. SVR is used in Apache Spark to predict landslide rainfall. When comparing the RMSE findings with the outcomes of the earlier techniques, the suggested strategy outperforms the other approaches and has the best value. [11] Yukata Nakagawa et al. discuss how rainfall attenuation has been researched for a long time from the perspective of quality assessment in satellite system communication link design. In this study, the estimation of rainfall using both left- and right-hand circularly polarized signals is reported experimentally. In addition to using multiple regression to estimate heavy rainfall, smoothing techniques and outlier removal can improve estimation accuracy.

[12] Despite using the WRF model, V. A. P. C. Perera et al. discovered that six model simulations had to be eliminated. Additionally, it has been discovered that using Time Series Plots in conjunction with forecasts derived from the average of the model simulations is not a very reliable way to predict a value from the twelve stimulations that were examined. Additionally, considering all of the WRF model's model simulations, fitted stepwise regressions for the climate zones were unable to produce a forecast that could be trusted. Similar results to the model simulations were also obtained using the MSE values of the fitted multiple linear regressions. The fitted stepwise regressions also had to eliminate certain model runs since they were not significant. Therefore, it cannot be regarded as an advancement of this technique. Principal component analysis also revealed that a new regression may be fitted with just two principal components when taking the Wet zone into account. A new regression can be fitted for the Inter zones and Dry zone using just three principal components. Fitted principal component regressions demonstrated much lower mean square error (MSE) than all model simulations when Sri Lanka's three distinct climate zones were taken into separate consideration. The final findings showed that all twelve

model simulations of the WRF model may be used to generate a dependable rainfall forecast for the Sri Lankan region by applying principal component analysis and principal component regression.

[13] The Imrus Salehin et al. model, which is based on LSTM and RNN, was created to forecast the amount of rainfall both annually and monthly. The system uses pressure, temperature, wind speed, and wind direction data to anticipate the quantity of precipitation that will fall in a month or a year with accuracy. One of the most important aspects of agriculture is rainfall. Despite the availability of numerous contemporary technologies, farmers primarily rely on rainfall [13]. However, they are costly and challenging to administer. Therefore, farmers stand to gain from early weather forecasts and knowledge of when and how much rainfall is expected. The primary objective of the suggested approach in this work is to identify ways that rainfall forecasting can assist farmers

[14] Sri Dewi Sartika Syarifuddin et al. converted the hardware rainfall gauge results from earlier study to successfully simulate a website-based rainfall information system. The measurement parameters that are shown on the operational website. A geometric method is used to translate the data, and it determines the volume of rainfall by using the cube volume and triangular prism. Performance validation was done in order to test the planned system by contrasting the website's rainfall measurement findings with calculations made by hand. Based on this validation test, there is 100% correctness in the system. Additionally, delay testing has been done, and the average generated delay is 2.5 seconds. It is envisaged that this proposed system can serve as a supporting system for monitoring and managing rainfall in order to prevent future tragedies. [15] According to Lince Rachel Varghese et al., Kerala's rainfall rate has altered over time, with varying seasons. In the South-East and North-West monsoons, there is an increase in the pattern of daily, weekly, and monthly precipitation. The soil has almost completely lost its fundamental texture and ability to hold water. All of this causes chaos. Frequent periods of rain and drought impact rubber output. Unpredictable rainfall patterns have a negative impact on yield and spread illness. Rain guarding should be employed to collect produce if there are morning showers before the rubber tree's tapping hours. Better prediction models provide information to both regulatory agencies and the general public. Not just the media, but every public agency must adequately address the impact of climate change. A more efficient medium should be used to convey climate risk information. Kerala may experience droughts and floods in the upcoming years as a result of these changes in rainfall patterns

3. Proposed Models

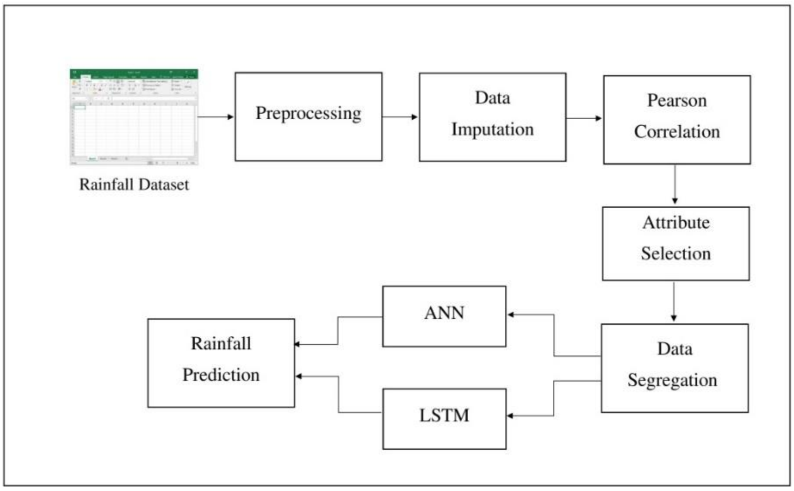


Figure 1: Proposed model for Rainfall Prediction

The proposed model for the rainfall prediction is carried out on two datasets, one is on Non-Indian subcontinent and other is for the Indian region dataset with the above model as depicted in the above figure 1. The proposed model follows series of the steps to carry out the prediction process on rainfall datasets as mentioned below.

Step 1: Dataset collection*:* For the purpose of the rainfall prediction the proposed model uses non- Indian dataset from the URL https://www.kaggle.com/datasets/jsphyg/weather-dataset-rattle-package/. About ten years' worth of daily weather observations from several places in Australia are included in this collection. Numerous weather stations provided observations. Rain tomorrow is the variable that needs to be predicted. It indicates whether or not it rained the following day. If there was at least 1 mm of rain that day, the answer to this column is yes.

This dataset contains some attributes which are described as follows: "Date" this attribute indicates about" The date of observation", "Location" attribute indicates about" The common name of the location of the weather station "."Mini Temp" this attribute indicates about" The minimum temperature in degrees Celsius ","Max Temp " attribute indicates about" The maximum temperature in degrees Celsius "," Rainfall" attribute indicates about" The amount of rainfall recorded for the day in mm"," Evaporation" attribute indicates about" The so- called Class A pan evaporation (mm) in the 24 hours to 9am"," Sunshine" attribute indicates about" The number of hours of bright sunshine in the day", "Wind Gust Dir " attribute indicates about" The direction of the strongest wind gust in the 24 hours to midnight"," Wind Gust Speed" attribute indicates about" The speed (km/h) of the strongest wind gust in the 24 hours to midnight "," WindDir9am" attribute indicates " Direction of the wind at 9am", and other attributes like WindDir3pm, WindSpeed9am, WindSpeed3pm, Humidity9am, Humidity3pm, Pressure9am, Pressure3pm, Cloud9am, Temp9am, Rain Today, Rain Tomorrow. And this dataset contains around 145460 Rows. Then another dataset for Indian sub-continent is collected from the URL : http://data.icrisat.org/icrisatweather/ which contains some attributes like Station, Date, max temperature, min temperature,RH1,RH2,Wind, Rain, SSH, Evaporation, Radiation, FAO56_ET, Latitude, Longitude and cum rain. This dataset contains around 14853 data.

Step 2: Preprocessing- Once both datasets are obtained, they tend to be applied to the process of preprocessing. In this pre-processing step, the attributes are collected in a double-dimensional list after the dataset has been read from the given path. To read the dataset from the given path, the Pandas library has been utilized in Python. This double-dimensional list data is used to estimate the early parameters of the attributes, like the mean and standard deviation, to describe the dataset characteristics. After this process, the dataset attribute information is being obtained for different data types, like string and float values, to assert the entropy of the data types in the dataset. The attributes like rain today are labeled as 1 for yes and 0 for no; in the same way, rain tomorrow is also labeled as 1 for yes and 0 for no. Once the dataset is labeled, their frequency is estimated to check the balance of the dataset for the labeled classes. These labeled classes are oversampled to balance them equally. This process eventually enhances the preprocessing process to yield good results in rainfall prediction.

Step 3: Data Imputation – A heat map for all the attributes is estimated using the oversampled data. This is done using the comparison of the transition data through the sorting process to get the total missing data and their respective percentages. These missing data is imputed using the fillna() function for attributes like 'Date', 'Location', 'WindGustDir', 'WindDir9am', and 'WindDir3pm'. For the purpose of imputation, a label encoder is employed in conjunction with the fit transform function for every object in the attributes as a formal parameter. After transforming the characteristics using the fit transformer, the

IterativeImputer() function is used to deploy multiple imputation by chained equations, producing the object of mouse imputation. Under particular assumptions about the data missingness mechanism, the multiple imputation technique known as MICE is used to replace absent data values in a data collection. The model can produce numerous copies if it starts with a dataset that has missing values in one or more of its variables. Following the application of mice imputation, the missing values for the specific combination are identified, and the missing values are predicted using known values from other features in the data as predictors. Next, the interquartile range (IQR) ranges from 0.75 to 0.25 are estimated using the imputated values that were obtained. The dataset containing the imputation of missing values is ultimately produced by this technique.

Step 4: Pearson Correlation – The process of Mice imputation yields a comprehensive dataset list, which is then subjected to Pearson Correlation analysis. The values of the Pearson Correlation are used to discover the attributes with the lowest correlation. Pearson Correlation establishes the correlation between the attributes. As a consequence, a correlation matrix is produced, which may be helpful in deciding which combination of the traits to choose. Following this discovery, the characteristics with lesser correlation are disregarded and new correlation values are computed. Equation 1, which is shown below, is used to perform the Pearson correlation.

$$r = \frac{\sum(x_i-\bar{y})(y_i-\bar{y})}{\sqrt{\sum(x_i-\bar{x})^2}\sum(y_i-\bar{y})^2} \text{ -------- } (1)$$

Where,

x_i=values of x (independent) variable

y_i= values of y (Dependent) variable

$\bar{x}$= mean of x variable values

$\bar{y}$= mean of y variable values

Following the compilation of the dataset in the double-dimensional list, after the correlation step, a few superfluous columns are eliminated from the list. The obtained list is then fed to the next step of data segmentation, as explained in the next step.

Step 5: Attribute Selection – In this process of attribute selection the imputated and preprocessed is used to apply the minMaxscaler function. Minmaxscaler Scale each feature to a specified range to transform the features. Each feature is scaled and translated separately by this estimator so that it falls into the designated range on the training set, for example, between zero and one. Although MinMaxScaler linearly scales outliers into a set range, where the largest data point corresponds to the maximum value and the smallest one to the minimum value, it does not lessen the impact of outliers. Consult Compare MinMaxScaler with other scalers for an illustration of a visualization. The minmaxscaler transformation can be given by the equation 2 and 3.

$$\mathrm{x_{std}} = \frac{(\mathrm{x - x.min(axis{=}0)})}{(\mathrm{x.max(axis{=}0) - x.min(axis{=}0)})} \text{-----}(2)$$

$$\mathrm{x_{\,scaled}} = \mathrm{x_{\,std}} * (\max - \min) + \min \text{—}(3)$$

where min, max = feature_range.

Following this procedure, two lists are produced for the "RainTomorrow" prediction property. Next, the features with value 10 are chosen based on the k highest scores. Next, using the random state zero and an estimators value of 100, the chosen attributes are fitted for the transform. The important attributes that will be trained in the subsequent steps are produced by this process. The selected important attributes are 'Rainfall', 'Sunshine', 'WindGustSpeed', 'Humidity9am', 'Humidity3pm', 'Pressure9am', 'Pressure3pm', 'Cloud9am', 'Cloud3pm' and 'RainToday'.

Step 6:Data Segregation – Here all the attributes like 'Location', 'MinTemp', 'MaxTemp', 'Rainfall', 'Evaporation', 'Sunshine', 'WindGustDir', 'WindGustSpeed', 'WindDir9am', 'WindDir3pm', 'WindSpeed9am', 'WindSpeed3pm', 'Humidity9am', 'Humidity3pm', 'Pressure9am', 'Pressure3pm', 'Cloud9am', 'Cloud3pm', 'Temp9am', 'Temp3pm' and 'RainToday' are kept in a list called X. On the other hand 'RainTomorrow' is kept in list called Y. In order to divide our data into train and test sets, we utilize the train_test_split() function. Our data must first be separated into features (X) and labels (y). The dataframe is split up into four sections: y_train, y_test, X_train, and X_test. The model is trained and fitted using the X_train and y_train sets. To determine whether the model is correctly predicting the outputs and labels, utilize the X_test and y_test sets. The size of the test sets and the train can be explicitly tested. Maintaining our train sets larger than the test sets is advised. *Train Set:* The set of data used to fit the model is known as the training dataset. The dataset used to train the model. The model sees and learns from this data. *Test set:* To provide an accurate assessment of the final model fit, a subset of the training dataset is used as the test dataset. Proposed system uses 33% of our data for test sets and 67% for training sets.

validation set: When adjusting the model's hyper parameters, a validation dataset is a sample of data taken from the training set. It is used to estimate the model's performance.

After this process data scaling is applied, setting the minimum and maximum values of each feature to zero and one, respectively. The data is shrunk by MinMax Scaler within the specified range, which is typically 0 to 1. By scaling features to a specified range, it modifies data. It maintains the original distribution's structure while scaling the numbers to a predetermined range. To do this, the MinMaxScaler() method is defined by the Sklearn preprocessor

As a result of this MinMaxScaler() method yields for lists like train_X, test_X, train_Y and test_Y. These four lists are used to apply the Artificial Neural network as described in the next step.

Step 7: Artificial Neural network- This step configures an Artificial neural network object for sequential type for the list train_X, test_X, train_Y and test_Y using sequential() method of keras. Sequential API gets its name from the basic principle of sequencing the Keras layers in a consecutive manner. The majority of ANNs also feature layers arranged in sequential sequence, with data moving sequentially from one layer to the next until it reaches the output layer. After that, a dense layer of 11 kernel units in an input layer is initialized using the 'uniform' parameter. Initializers specify how to set the Keras layers' initial random weights. Additionally, this layer has a "relu" activation function with a 21 input dimension. Next, a dense layer with a single unit of kernel—which is initialized with a "uniform" parameter—is added as an output layer. Sigmoid is used as the activation function to strengthen the output layer.

The "adam" optimizer, which applies the Adam method, is used to build the generated ANN model. The adaptive estimation of first- and second-order moments serves as the foundation for the stochastic gradient descent technique known as Adam optimization. The approach employed by the Adam algorithm is "computationally efficient, has little memory requirement, is invariant to diagonal rescaling of gradients, and is well suited for problems that are large in terms of data and parameters." Since there are two classes, such as "rain" or "not rain," the binary cross entropy parameter is utilized during model compilation to assess the model's correctness. With a batch size of 100, the ANN neural network is trained for 100 epochs.

The equations 4 and 5 below display the ReLU and Sigmoid functions respectively.

$f(x) = max(0, x)$ ___ (4)

Where, x is any positive value

$S(X) = \frac{1}{(1+e^{-x})}$ ______(5)

Where,

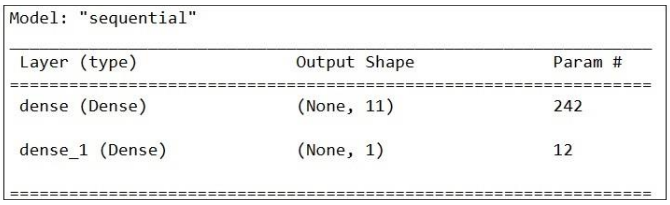
X is the input to a neuron

$f(x) = Relu\ Activation\ Function$

$S(x) = Sigmoid\ Activation\ Function$

e= Euler's Number

The summary of the constructed ANN model and its training results is depicted in figure 2 and 3 respectively.



```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense (Dense)               (None, 11)                242

 dense_1 (Dense)             (None, 1)                 12

=================================================================
```

Figure 2: ANN Model Summary

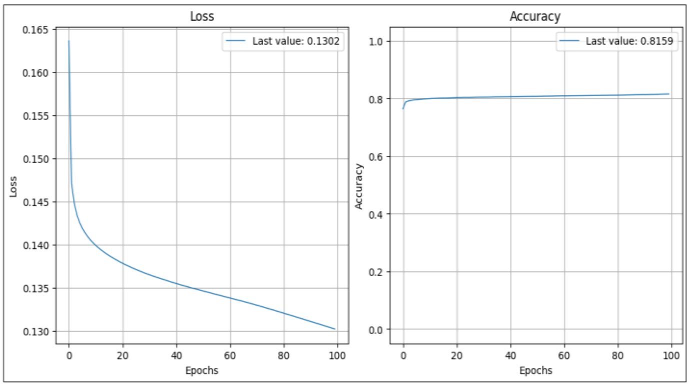
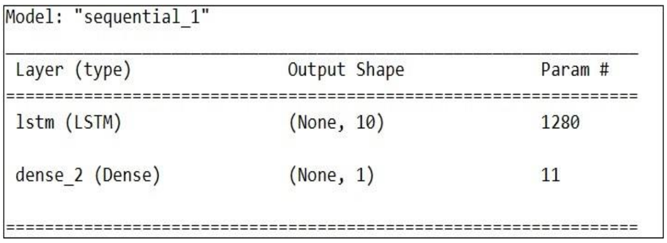


Figure 3: ANN model training Results

Step 8: Long short term memory (LSTM) - Input parameters for the LSTM neural network include test_x, test_y, train_x, and a scalar normalization object. A one-dimensional space with a single feature, ten units of samples, and a TRUE return sequence are used to introduce the LSTM model, which has a few parameters like train_X1.shape[1], train_X1.shape[2]. Then, a Dense layer with a kernel size of 1 and an activation function of "relu" is added. Activation functions on neurons are used by the dense layer of a densely linked neural network to effectively learn new information. Two dense layers make up the basic LSTM neural network, however just one kernel, size 1, is employed for one-dimensional data in this case. When building a neural network, a batch size of 100 and 100 epochs are employed, along with the shuffle parameter set to false.

The summary of the constructed LSTM model and its training results is depicted in figure 4 and 5 respectively.



```
Model: "sequential_1"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 lstm (LSTM)                 (None, 10)                1280

 dense_2 (Dense)             (None, 1)                 11

=================================================================
```

Figure 4: LSTM Model Summary

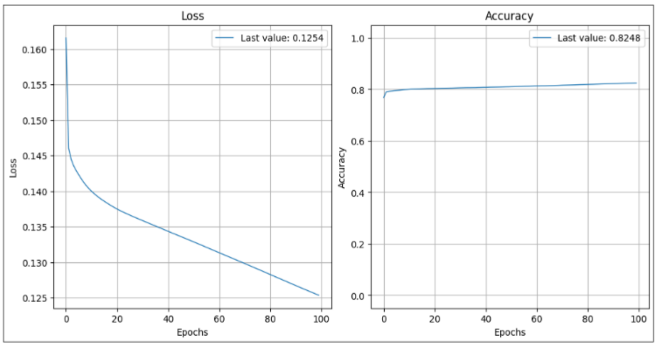


Figure 5: LSTM model training Results

4. Results and Discussions

The proposed Rainfall prediction methodology is implemented in the Python programming language and operates on a Windows-based machine platform. The Jupyter IDE has been used during the development process. The development laptop has an Intel Core i5 processor, 500GB hard disk, and 8GB of RAM. To achieve a successful deployment of a rainfall prediction system, the effectiveness of rainfall prediction needs to be evaluated. This method makes use of a dataset that contains a variety of rainfall-related attributes and is supplied as an input as mentioned in the prior step. As explained in the section below, the ANN and LSTM deep learning models will produce the results required for the performance evaluation.

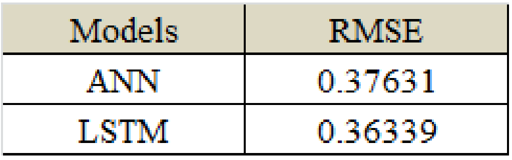
Performance Evaluation based on Root Mean Square Error (RMSE)

Testing that took use of the system's degree of error has allowed for an accurate measurement of the approach. Mean absolute error is used to calculate this error, which effectively shows the inaccuracy in the application of the methodology. To accurately understand the frequency of inaccuracy in continuous attributes, utilize equation 6. Qualities that work well for this assessment are the rainfall prediction scores produced by ANN and LSTM Models. To forecast scores using Root mean Square error (RMSE). To determine the error margin of the recommended procedure, a measurement known as the root mean square error (RMSE) is conducted. The error margin between the actual event detection and the expected event detection that is carried out by the ANN and LSTM model is computed in this inquiry using the root mean square error, or RMSE. The equation 1 supplied below illustrates the RMSE approach.

$$\mathrm{RMSE}_{fo} = \left[\sum_{i=1}^{N} (z_{f_i} - z_{o_i})^2 / N\right]^{1/2} \quad \_\_\_\_(6)$$

Where , ∑ - Summation. , $(Z_{fi} - Z_{oi})^2$ - Differences Squared for the Rainfall Prediction.
N - Number of trails.

Before the Error Rate of the Approach can be evaluated using RMSE, the Mean Square Error, or MSE, must be obtained. The difference between the expected and actual rainfall prediction that was detected is known as the mean square error, or MSE. An increasing number of trails are being used to test the entire project; the outcomes are shown in table 1 below for the ANN and LSTM models for Non-Indian Dataset along with respective graph in figure 6.



Models	RMSE
ANN	0.37631
LSTM	0.36339

Table 1: RMSE results for ANN and LSTM

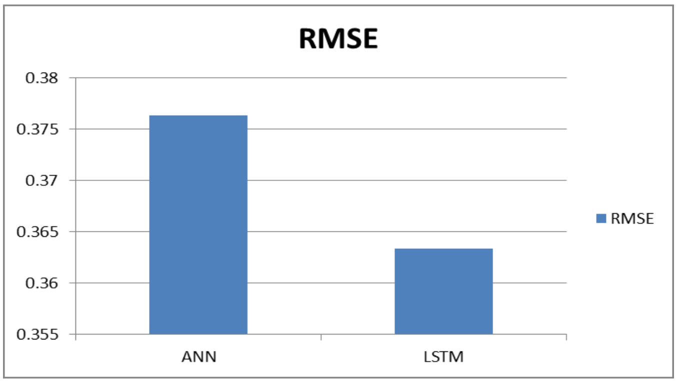


Figure 6:RMSE results for ANN and LSTM

The prediction results for ANN and LSTM for No-Indian and Indian Datasets are shown in the Figure 7,8,9 and 10 respectively.

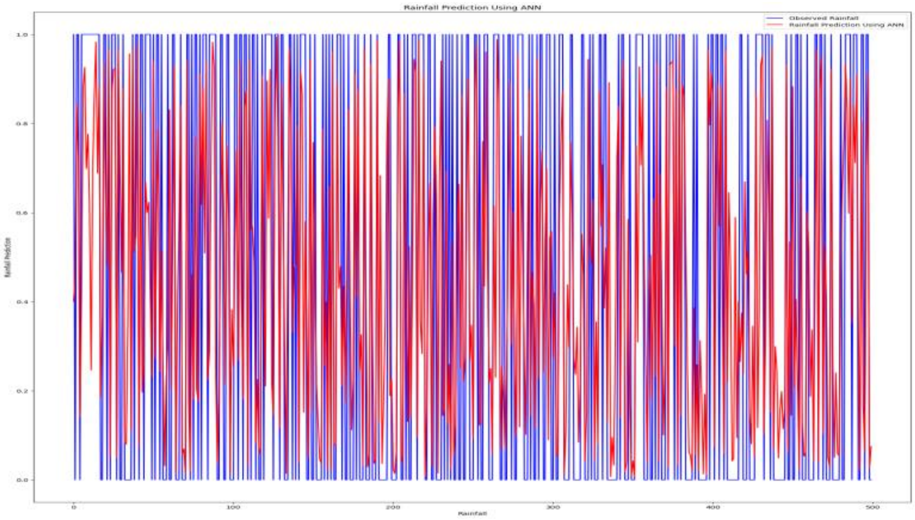


Figure 7: Rainfall Prediction for ANN model for No-Indian Dataset

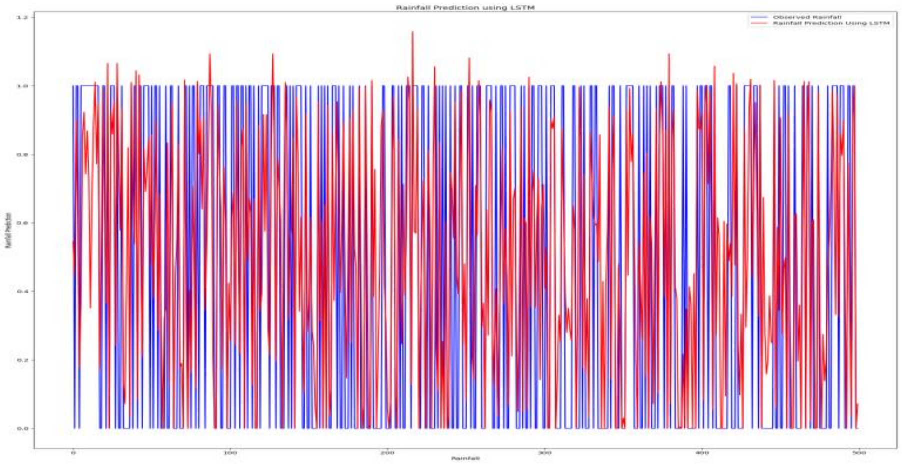


Figure 8: Rainfall Prediction for LSTM model for No-Indian Dataset

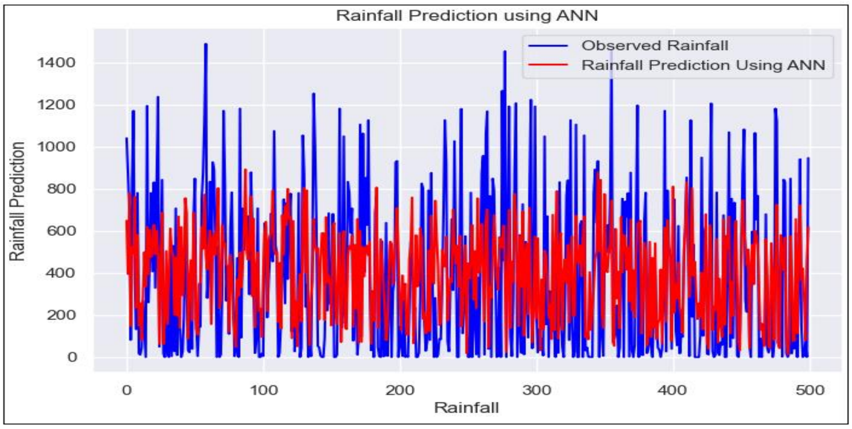


Figure 9: Rainfall Prediction for ANN model for Indian Dataset

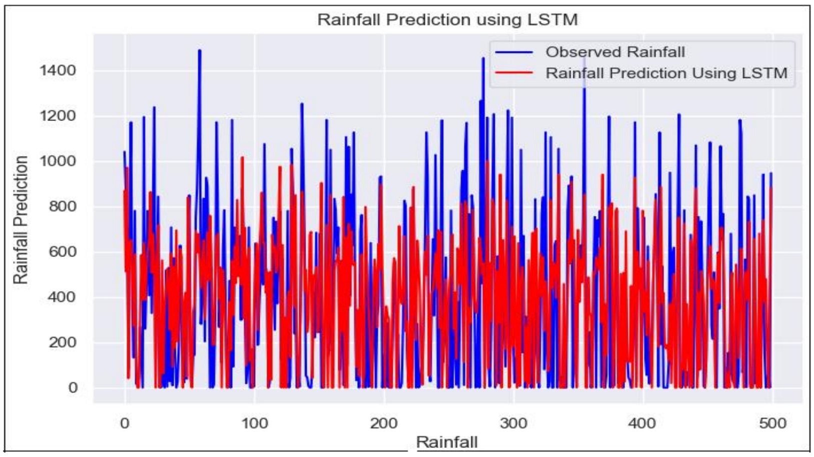


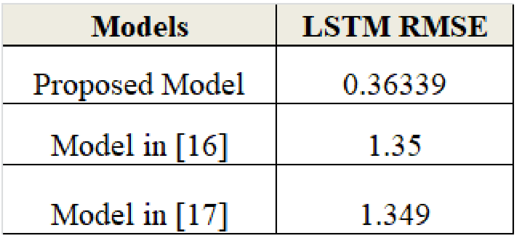
Figure 10: Rainfall Prediction for LSTM model for Indian Dataset

The proposed model is compared with two research articles stated in [16] and [17]. The application of deep learning model for the prediction of monthly rainfall across the Karnataka region is described by Pragati Kanchan et al. in [16]. The data demonstrate that the deep learning technique improved with LSTM yields superior prediction performance. Accuracy matrices have been used to assess this approach’s performances. With a minimal Mean Absolute Percentage Error of 0.79 and a Root Mean

Squared Error of 1.35 for prediction by the LSTM model as mentioned in [16]. The study of Andrea Trucco et al. in [17] shows how important it is to take temporal correlation into account when predicting wind and rainfall based on underwater acoustic noise measurements. While earlier studies have demonstrated that ML techniques can outperform empirical equations in the memoryless prediction of wind and rainfall from underwater noise, this study demonstrates that the LSTM architecture is a supervised ML technique capable of accurately modeling the temporal correlation inherent in the meteorological phenomena mentioned.

In terms of rainfall prediction, performance was noticeably better than that of the RF regression when the five previous spectra—that is, those that go back 50 minutes—were added to the current one. Upon deleting the samples obtained over two days during a flood that occurred in Genoa, which is approximately 80 km away from the sensors, the LSTM architecture was evaluated using tenfold cross-validation without shuffling. This resulted in rainfall intensity forecasts with an RMSE of 1.349 by [17].

On understanding both the earlier works for rainfall prediction using LSTM by [16,17] is summarized with the proposed model in Table 2 along with the subsequent graph in Figure 11.



Models	LSTM RMSE
Proposed Model	0.36339
Model in [16]	1.35
Model in [17]	1.349

Table 2: RMSE comparison for LSTM

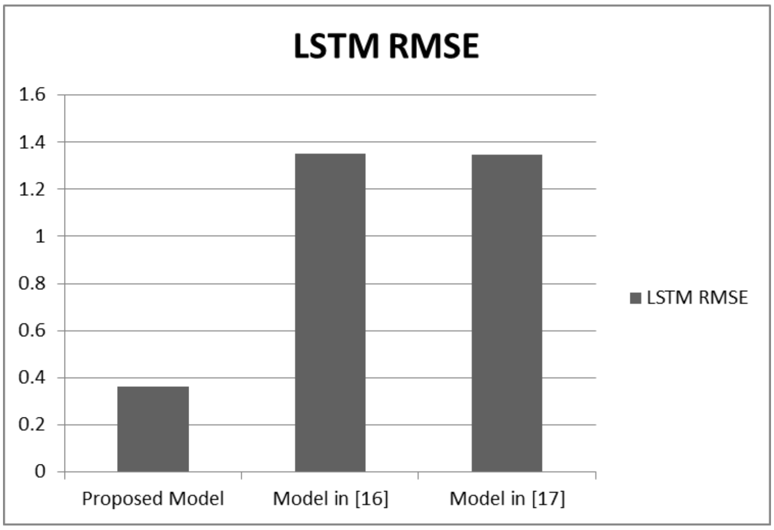


Figure 11: RMSE Comparison for LSTM Model

The test results showed how well the rainfall prediction system created in this study worked. An RMSE of 0.36339 is obtained from the analysis of the system's degree of error in calculating the probability scores prior to and following the deviation, which is modest. These results have a comparable to the LSTM model that is explained in references [16] and [17]. By comparison with [16] and [17], the created LSTM model obtains much better performance. The experimental data displayed in Table 2 and

Figure 11 support this. This illustrates how the rainfall forecasting method has been improved, significantly increasing prediction accuracy.

5. Conclusion and Future Scope

Large-scale human casualties as well as significant harm to infrastructure and natural resources are brought about by natural disasters like landslides. Although it is impossible to completely prevent landslides, their effects can be reduced if they can be anticipated in advance. Since landslides are typically associated with periods of heavy rainfall or swift snowmelt, rainfall forecasting is a useful method for estimating the likelihood of landslides. The goal of this research is to determine the best method for generating early warnings for heavy rainfall by developing and comparing the performance of many machine learning algorithms for rainfall prediction. This research article contains a comprehensive deployment of the rainfall estimation method that is being implemented using the ANN and LSTM deep learning model. The current method provides an excel sheet with a rainfall dataset as an input to the designed system. Rainfall prediction cannot be performed using this workbook file until it has been properly preprocessed. In order to fully segment the preprocessed datasets, the missing values are imputed. The segmented data is then given to deep learning models, such as ANN and LSTM, for a variety of datasets from the Indian subcontinent and other datasets across the continent, in order to predict the rainfall. The test results demonstrated the effectiveness of the rainfall forecast system developed in this study. The investigation of the system's degree of error in determining the probability scores before and after the deviation yields a moderate RMSE of 0.36339. These findings are similar to the LSTM model that the cited sources [16] and [17] explain. In contrast to [16] and [17], the developed LSTM model achieves significantly higher performance. This is corroborated by the experimental findings shown in Table 2 and Figure 11. This demonstrates how improvements have been made to the rainfall forecasting technique, leading to a notable increase in prediction accuracy.

An interactive API can be built for the model's future improvement so that other researchers and developers can use it. Large datasets can also be used to train the model so that it can be utilized in online and mobile applications through the interactive deployment of the same using big data and cloud architectures.

References

1. S. Poornima, M. Pushpalatha, "Prediction of Rainfall Using Intensified LSTM Based Recurrent Neural Network with Weighted Linear Units," in Atmosphere Access, 2019, 10, 668; doi:10.3390/atmos10110668.
2. Dr. C K.Gomathy , ANNAPAREDDY BALA NARASIMHA REDDY, ARAVAPALLI PAVAN KUMAR, AILE LOKESH, "A STUDY ON RAINFALL PREDICTION TECHNIQUES," in ResearchGate Access, Volume. 05 Issue: 10, Oct – 2021.
3. N. K. A. Appiah-Badu, Y. M. Missah, L. K. Amekudzi, N. Ussiph, T. Frimpong and E. Ahene, "Rainfall Prediction Using Machine Learning Algorithms for the Various Ecological Zones of Ghana," in IEEE Access, vol. 10, pp. 5069-5082, 2022, doi: 10.1109/ACCESS.2021.3139312.
4. S. Poornima, M. Pushpalatha, "Drought prediction based on SPI and SPEI with varying timescales using LSTM recurrent neural network," in ResearchGate Access, Volume 23, pages 8399–8412, (2019).
5. Chalachew Muluken Liyew, Haileyesus Amsaya Melese, "Machine learning techniques to predict

daily rainfall amount," in Open Access, Article number: 153 (2021), doi:10.1186, s40537-021-00545-4.

6. S. Poornima, M. Pushpalatha, Raghavendra B. Jana, Laxmi Anusri Patti, "Rainfall Forecast and Drought Analysis for Recent and Forthcoming Years in India," in MDPI Access, Vol. 2023, *15*(3), 592, doi:10.3390, w15030592.
7. T. Dananjali, S. Wijesinghe and J. Ekanayake, "Forecasting Weekly Rainfall Using Data Mining Technologies," 2020 From Innovation to Impact (FITI), Colombo, Sri Lanka, 2020, pp. 1-4, doi: 10.1109/FITI52050.2020.9424877.
8. L. Huang, X. Liu and H. Wei, "Urban Rainfall Forecasting Method Based on Multi-model Prediction Information Fusion," 2020 6th International Conference on Information Management (ICIM), London, UK, 2020, pp. 210-214, doi: 10.1109/ICIM49319.2020.244700.
9. Y. K. Joshi, U. Chawla and S. Shukla, "Rainfall Prediction Using Data Visualisation Techniques," 2020 10th International Conference on Cloud Computing, Data Science & Engineering (Confluence), Noida, India, 2020, pp. 327-331, doi: 10.1109/Confluence47617.2020.9057928.
10. Z. -J. Lee, C. -Y. Lee, X. -J. Yuan and K. -C. Chu, "Rainfall Forecasting of Landslides Using Support Vector Regression," 2020 3rd IEEE International Conference on Knowledge Innovation and Invention (ICKII), Kaohsiung, Taiwan, 2020, pp. 1-3, doi: 10.1109/ICKII50300.2020.9318930.
11. Y. Nakagawa, T. Higashino and M. Okada, "Multiple Regression For Rainfall Estimation Using Right/Left-hand Circularly Polarized Signals," 2020 IEEE 9th Global Conference on Consumer Electronics (GCCE), Kobe, Japan, 2020, pp. 632-633, doi: 10.1109/GCCE50665.2020.9291977.
12. V. A. P. C. Perera and K. G. H. S. Peiris, "An Improved Statistical Method for Rainfall Forecasting in Sri Lanka using the WRF Model," 2020 International Conference and Utility Exhibition on Energy, Environment and Climate Change (ICUE), Pattaya, Thailand, 2020, pp. 1-7, doi: 10.1109/ICUE49301.2020.9307070.
13. I. Salehin, I. M. Talha, M. Mehedi Hasan, S. T. Dip, M. Saifuzzaman and N. N. Moon, "An Artificial Intelligence Based Rainfall Prediction Using LSTM and Neural Network," 2020 IEEE International Women in Engineering (WIE) Conference on Electrical and Computer Engineering (WIECON-ECE), Bhubaneswar, India, 2020, pp. 5-8, doi: 10.1109/WIECON-ECE52138.2020.9398022.
14. S. D. S. Syarifuddin, A. Khurniawan, S. Aulia, D. N. Ramadan and S. Hadiyoso, "Rainfall Information System Using Geometry Algorithm on IoT Platform," 2021 IEEE Asia Pacific Conference on Wireless and Mobile (APWiMob), Bandung, Indonesia, 2021, pp. 195-199, doi: 10.1109/APWiMob51111.2021.9435219.
15. L. R. Varghese and K. Vanitha, "A Time-series based Prediction Analysis of Rainfall Detection," 2020 International Conference on Inventive Computation Technologies (ICICT), Coimbatore, India, 2020, pp. 513-518, doi: 10.1109/ICICT48043.2020.9112488.
16. Pragati Kanchan and Nikhil Kumar Shardoor “ Rainfall Analysis and Forecasting Using Deep Learning Technique”, Journal of Informatics Electrical and Electronics Engineering, Vol. 02, Iss. 02, S. No. 015, pp. 1-11, 2021
17. A. Trucco, A. Barla, R. Bozzano, S. Pensieri, A. Verri and D. Solarna, "Introducing Temporal Correlation in Rainfall and Wind Prediction From Underwater Noise," in IEEE Journal of Oceanic Engineering, vol. 48, no. 2, pp. 349-364, April 2023, doi: 10.1109/JOE.2022.3223406.