International Journal of Computing and Artificial Intelligence

E-ISSN: 2707-658X
P-ISSN: 2707-6571
www.computersciencejournals.com/ijcai
IJCAI 2024; 5(2): 211-219
Received: 13-05-2024
Accepted: 18-06-2024

Chandra Sekhar Sanaboina
Assistant Professor, Department of Computer Science and Engineering, University College of Engineering Kakinada, JNTUK, Kakinada, Andhra Pradesh, India

Corresponding Author:
Assistant Professor, Department of Computer Science and Engineering, University College of Engineering Kakinada, JNTUK, Kakinada, Andhra Pradesh, India

A comparative study of different machine learning techniques for forecasting rainfall

Chandra Sekhar Sanaboina

DOI: https://doi.org/10.33545/27076571.2024.v5.i2c.117

Abstract
The main objective of this paper is to find the best machine-learning technique for rainfall prediction. Predicting rainfall is a challenging and uncertain process that greatly impacts society. Timely and accurate forecasting can proactively reduce human and financial loss. In this article, a prediction model has been developed to forecast whether it will rain or not in major Australian cities based on the weather data on a given day. This data has been used to test the accuracy of several machine learning algorithms, such as logistic regression, random forest classifier, decision trees, neural networks, light GBM, CatBoost, and XGBoost. The results of this research article depicted that CatBoost, XGBoost, and LightGBM performed better when compared to the other models, with accuracy scores of 95%, 94%, and 94%, respectively. However, if speed is a priority, we can continue with LightGBM rather than XGBoost or CatBoost.

Keywords: Rainfall prediction, machine learning, decision tree, neural network, random forest, xgboost, k-nearest neighbor, logistic regression, lightgbm, catboost, Imbalanced data, oversampling, imputation

1. Introduction
Due to climate change, rainfall is becoming a more unpredictable and difficult atmospheric process. Therefore, it is crucial to predict rainfall since it can have a variety of negative effects, including the destruction of farms, crops and property damage. To reduce hazards to life, and property and to manage agricultural fields more efficiently, a better forecasting model is required. For forecasting, rainfall data is collected using the weather AUS dataset. Under the examination, this dataset includes daily weather observations from several weather stations in Australia. Rain tomorrow is the target variable which indicates whether or not it rained the following day. Either yes or no.

For rainfall prediction, numerous machine learning classifiers can be used, including logistic regression, random forest classifiers, decision trees, neural networks, light GBM, and categorical boosting, and the appropriate algorithm must be modeled by the specifications to achieve the desired accuracy model. To predict rainfall, it may be possible to apply preventative and mitigation measures for these natural events.

This study makes use of data exploration, imputation, and transformation techniques. It requires imputing the category columns with the mode and then converting them to numeric values using the label encoder. Once all the columns in the entire data frame have been transferred outliers have been identified and eliminated, and the final working dataset has been received. To impute the missing data, multiple imputations by chained equations (MICE) package has been used. The feature selection process, where we must choose the best characteristics from a list of various accessible attributes, can then be started. The filter technique and the wrapper method are two approaches to the feature selection process. To assess the performance of the models, we used F1-Score, Accuracy, Precision, Recall, and Area under Curve as evaluation metrics.

2. Related Works
Many experts have focused on increasing the accuracy of machine learning methods for predicting rain. Here, several related studies are covered.

Several classification techniques were investigated for predicting rainfall in four ecological areas of Ghana. Among the classification methods are k-nearest neighbor, extreme gradient boosting, multi-layer perceptron, decision tree, and random forest. RF, XGB, and MLP all did well, with KNN performing the least well in all zones [1].

The rainfall forecast model was developed using the convolutional long short-term memory (convLSTM) system-based deep learning network. However, the MapReduce architecture serves as the foundation for the proposed deep learning system. The proposed approach has a minimal mean square error (MSE) of 0.001 and a percentage root mean square difference (PRD) of 0.0021, according to an investigation of the rainfall forecast database [2].

Rainfall forecasting in Odisha for both annual and non-moon sessions (India) was estimated in their research. The maximum precipitation was predicted using support vector regression and multi-layer perception [3]. The outcomes of Multi-Layered Perceptron and simple regression approaches were contrasted with those of support vector regression.

Using multi layered perceptrons the dynamic regional combined short-term precipitation forecasting technique (DRCF) was developed to predict the rainfall. The experimental results reveals that DRCF performs better when compared to the existing techniques in terms of threat score (TS) and the root mean square error (RMSE) [4].

Five data-mining methods were used to produce prediction models [5] they are Support vector machines, neural networks, random forests, classification trees, regression trees, and k-nearest neighbors are some examples. The radar data for Oxford were used to develop the baseline model, Model I. Oxford radar and TB data were used in Model II to forecast rainfall. The radar data gathered at South Amana and lowa city were used to build Model III. Model II outperforms the other two models in terms of rainfall forecasts for upcoming periods.

In their research [6], they intended to develop algorithms that could forecast time series from cumulative monthly rainfall, two non-parametric sub-sampling methods were investigated. Results are assessed using time series with high levels of roughness derived from a single geographic location's cumulative monthly historical rainfall data. The positive results encourage further research and funding for implementation efforts in the target regions.

They investigated a one-dimensional convolutional neural network with a multi-layer perceptron for daily precipitation estimation (1-day to 5-day in advance). The deep learning technique known as Multi-Layered Perceptron (deep MLP) and support vector regression were contrasted with the suggested hybrid approach [7]. Typically, the analysis's conclusions demonstrate that the suggested model does a better job of reflecting the complicated relationships between the causes of rainfall and its daily fluctuation.

To predict amount of rain with high accuracy and time resolution, a supervised learning algorithm-based hourly rainfall prediction (HRF) model [8] was developed. The hyper-parameters for the HRF model were hourly precipitable water vapor (PWV) produced by the global navigation system and temperature data (GNSS). The proposed methodology is then trained using a support vector machine. The average root-mean-square error (RMSE) and relative RMSE for the proposed HRF model are 1.36/1.39 mm/h and 1.00/0.67, respectively.

A deep convolutional neural network (CNN) forecasting technique was advised to estimate monthly rainfall for a particular region in eastern Australia [9]. The Australian Community Climate and Earth-System Simulator-Seasonal Prediction System (ACCESS), a forecasting tool made available by the Bureau of Meteorology, was compared to the proposed methodology. With the suggested CNN, better mean absolute and root mean square errors (RMSE) were attained.

The researchers of this work [10] applied rainfall prediction to identify barrier destruction. The El-Nino and IOD features were used in the tests of the first scheme, whereas the rainfall time series pattern was used in the tests of the second system. El-Nino and IOD parameters were used to make predictions, and the MAAPE values of the hidden layer, batch size, and learn rate drop period were 100, 64, and 50, respectively. A higher MAAPE value of 0.5810 was found in the results of the forecasts made utilizing the characteristics of rainfall

3. Experimental Dataset Description and ML Algorithms used

The purpose of this section is to outline the necessary background information regarding the experimental dataset and methodology used in this .

3.1 Dataset Information

The weatherAUS dataset has 23 attributes as shown in Table 1, and contains about 10 years of daily weather observations from various locations in Australia.

The target variable to be predicted is rain tomorrow. Does it mean that it rained the following day, Yes or No? If the amount of rain that day was 1 mm or greater, the column is set to Yes.

- **Number of attributes:** 23
- **Number of tuples:** 145460
- **Number of Independent Columns:** 22
- **Number of Dependent Columns:** 1

Table 1: WeatherAUS Attributes List

Attribute Name	Attribute Name
Date	Location
Min Temp	Max Temp
Evaporation	Sunshine
WindGustDir	WindGustSpeed
WindDir9am	WindDir3pm
WindSpeed9am	WindSpeed3pm
Humidity9am	Humidity3pm
Pressure9am	Pressure3pm
Cloud9am	Cloud3pm
Temp9am	Temp3pm
RainToday	RISK_MM
RainTomorrow	

3.2 ML Algorithms used

In this section, we are addressing the detailed description regarding the configuration of various machine learning algorithms, used in this article for the prediction of rainfall.

3.2.1 Decision Tree (DT)

A supervised machine learning technique is decision trees. It can be used to solve the both classification and regression problems. It is a tree-structured classifier, with each leaf node denoting the classification result and inner nodes denoting the dataset's features. The general format of a decision tree is described in detail in Fig. 1.

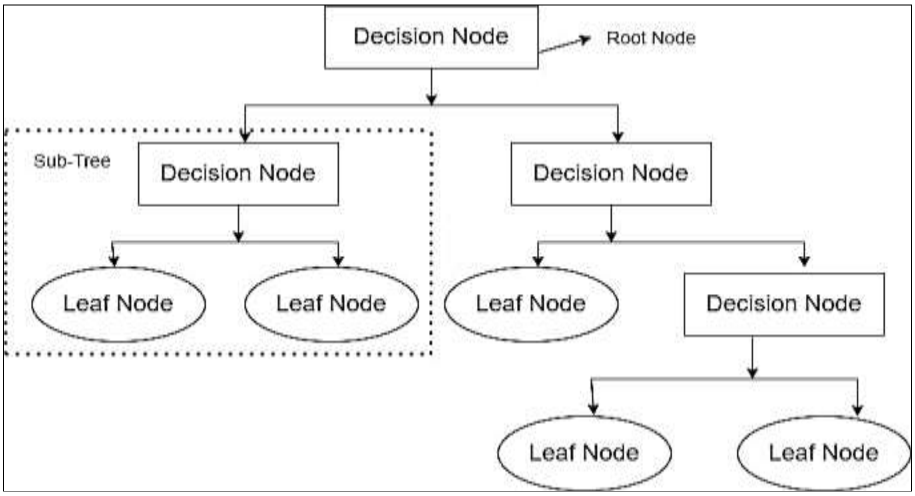

Fig 1: Decision Tree Architecture

3.2.2 Neural Network (MLPClassifier)

A network of perceptrons is called a multi-layer perceptron (MLP). It has an input layer that receives the input signal and features that are taken directly from the audio file as input. Mel Frequency Cepstral Coefficients, Mel Spectrogram Frequency, and Chroma are examples of the extracted features. An output layer that uses these characteristics to create predictions or choices for a given input is known as the hidden layer, as represented in Fig.2.

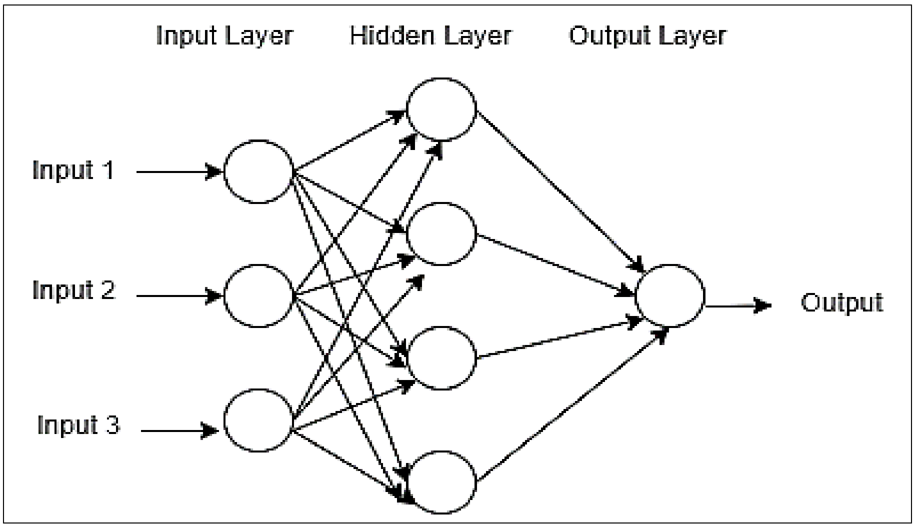

Fig 2: Multi-Layered Perception Classifier Architecture

3.2.3 Random Forest (RF)

Random forests are made up of many trees. As the number of uncorrelated trees rises, so does accuracy. Classifiers using Random Forest can help fill in certain missing values. The classification outcomes are chosen based on the classification outcomes with the highest majority after adding all the classification outcomes of these constructed trees. The random forest is constructed with a huge number of decision trees, as shown in Fig. 3.

The random forest technique includes the following steps

- **Step 1:** In this algorithm, n records are selected randomly from the data with k records.
- **Step 2:** For each record, a unique decision tree is constructed.
- **Step 3:** Every decision tree will generate an output.
- **Step 4:** The outcome is assessed for classification and regression using a majority vote or an average.

3.2.4 Extreme Gradient Boosting (XGBoost)

The term Extreme Gradient Boosting refers to an ensemble-supervised gradient boosting classifier is known by the abbreviation XGBoost. It is a extensible gradient boosting machine-learning method. Each new tree is created and added to the ensembled model until no more advancements are possible and the errors that the prior trees introduced have been corrected. To predict the outcome, the trees are then joined. One tree is created by XGBoost at a time in a forward motion.

3.2.5 LightGBM

A fast and secure gradient boosting framework is LightGBM. LightGBM trains the models using a parallelized tree architecture that can scale up to thousands of CPU cores or across hundreds of GPUs in a single operation according to its leaf-wise tree growth architecture. The model may be used for prediction with high accuracy on huge datasets, and the training procedure is very efficient in terms of memory utilization and speed.

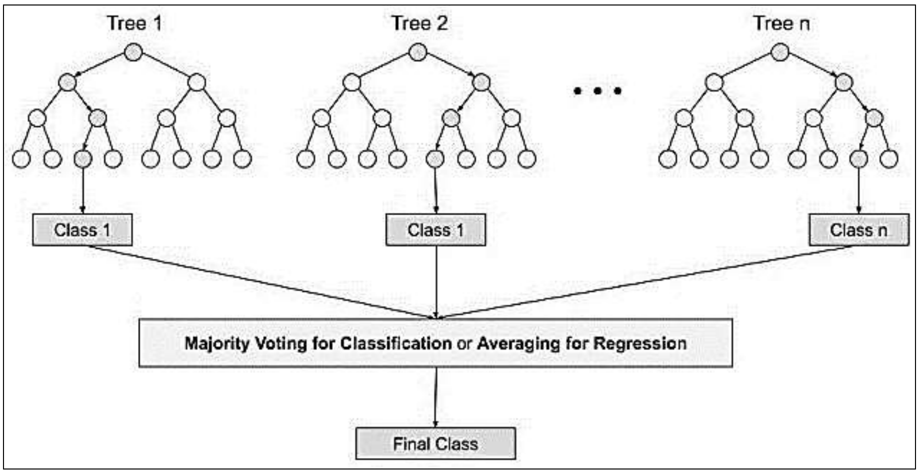

Fig 3: Random Forest Architecture

3.2.6 Categorical Boosting (CatBoost)

Yandex created the open-source boosting library known as categorical boosting. In addition to regression and classification, categorical boosting can be used for ranking, recommendation systems, forecasting and even personal assistants.

The name "CatBoost" comes from the two words "Category" and "Boosting." As mentioned, the library works well with multiple categories of data, including audio, text, and image as well as historical data. The acronym "Boost" corresponds to a gradient boosting machine learning approach because this library is based on a gradient boosting library. CatBoost employs decision trees, which use the same characteristics to create a left and right split for each level of the tree and limit the number of features divided per level to one. This reduces prediction time.

3.2.7 K-Nearest Neighbors (KNN)

As a non-parametric approach, K-Nearest Neighbor makes no assumptions about the underlying data. Since the method saves the training dataset rather than immediately learning from it, so it is also referred to as lazy learner. It performs the operations at the time of classification. KNN categorizes new data into a category that is highly close to the training data and simply saves the data from the training phase. The new data point in Fig. 4 is classified as Category 1 or A.

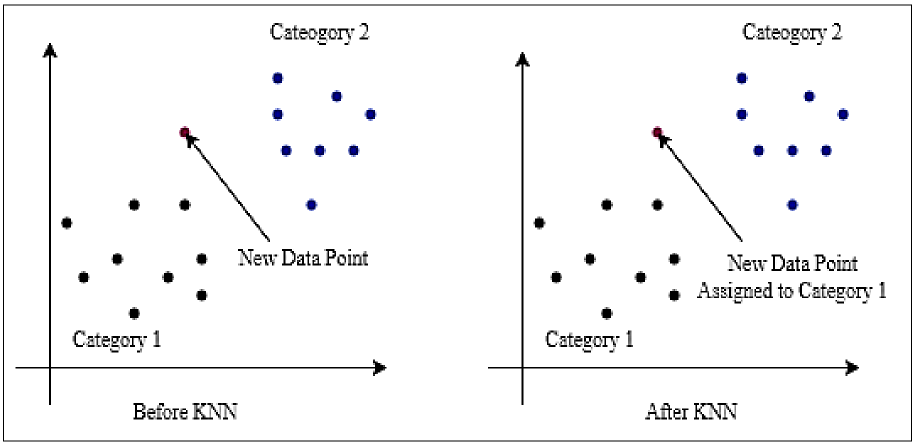

Fig 4: K Nearest Neighbor's Architecture

3.2.8 Logistic regression (LR)

One of the machine learning algorithm that is most frequently used in the supervised learning category is logistic regression. A well-established set of independent variables are used to predict the categorical dependent variable. It forecasts the behavior of a categorical dependent variable. The outcome must therefore be a discrete or categorical value. Instead of precise integers between 0 and 1 the probabilistic values between 0 and 1 are supplied. There are two possible outcomes: True or False, 0 or 1, or Yes or No.

The logistic regression equation can be derived from the linear regression. Below are the mathematical processes needed to obtain equations for logistic regression:

- We are aware that the equation for a straight line is:

$$Y = b_0 + b_1x_1 + b_2x_2 + \cdots + b_nx_n \qquad (1)$$

Let us divide the following equation by (1-y), because of y in a logistic regression model can only be between 0 and 1

$\frac{Y}{1-Y}$ = 0 for y = 0 and ∞ for y=1

However, we require a range between -[infinity] and +[infinity]. If we take the equation's logarithm, it becomes:

$$log\left[\frac{Y}{1-Y}\right] = b_0 + b_1x_1 + b_2x_2 + \cdots + b_nx_n \qquad (2)$$

4. Work Flow/Architecture of the proposed system: Fig. 5 shows the Work Flow design for the proposed system.

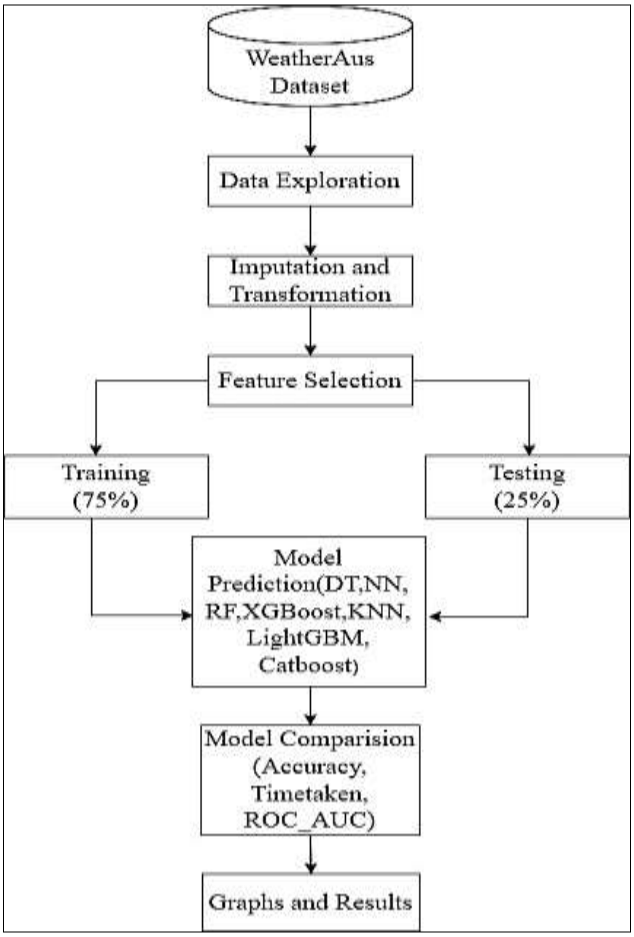

Fig 5: Proposed System Architecture

Step 1: Upload Dataset: Download the WeatherAUS dataset from Kaggle and upload it.

Step 2: Data Exploration
The size of the dataset was examined to see whether size compression is necessary. After that, we determined whether the dataset is balanced or not. To balance an unbalanced data collection, we must either oversample the minority or under sample the majority.

Step 3: Imputation and Transformation
The label encoder was used to convert the categorical columns' values from categorical to numeric after they have been imputed with mode. The multiple imputations by chained equations package (MICE package) were used to impute the missing values once all the columns in the entire data frame have been transformed into numeric columns. The final working dataset will then be created by identifying outliers and removing them using the interquartile range.

Step 4: Feature Selection
For the feature selection phase of our rainfall prediction model, both filter approach and the wrapper methods were utilized. We initially normalize our data before choosing features with the chi-square value filtering method. We transition from Standard Scaler to Min Max Scaler to prevent negative numbers.

Step 5: Training and Testing
To train the rainfall prediction model, we divided the dataset into training (75%) and test (25%) sets.

Step 6: Model Prediction: Several machine-learning approaches were employed to create a prediction model. The eight techniques we employed included Decision Trees, KNN, Logistic Regression, Light GBM, Catboost, XGBoost, and Random Forest Classifiers.

Step 7: Model Comparison: The model that performed the best in terms of precision, recall, F1-Score, ROC AUC, and Total Run Time must now be chosen.

5. Evaluation Metrics
To validate the findings and evaluate the effectiveness of the models, we can use accuracy, precision, recall, F1-score and AUC-ROC curves.

5.1 Accuracy
Accuracy is one of the most straightforward Classification metrics to evaluate and is determined as the ratio of accurate predictions to all other predictions. One way to put it is as in equation (3).

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \quad (3)$$

5.2. Precision
The precision score represents the ratio of correctly anticipated positive observations to all successfully predicted positive observations and is formulated as shown in equation (4).

$$\text{Precision} = \frac{TP}{TP+FP} \quad (4)$$

5.3 Recall
Recall/sensitivity is used to calculate the proportion of positive observations that are correctly predicted to all observations in a true class. It can be calculated by using equation (5)

$$\text{Recall} = \frac{TP}{TP+TN} \quad (5)$$

5.4 F1-Score
The weighted average of the Precision and Recall calculations is known as the F1 score. It can be proven that this score, which accounts for false positive and false negative readings, is more valuable than accuracy. It can be computed using equation (6)

$$F1-Score = \frac{2(Recall*Precision)}{Recall+Precision} \quad (6)$$

5.5 AUC-ROC
It is a popular and significant metric for assessing the effectiveness of the classification model. Let us start by understanding the receiver operating characteristic curve (ROC). This curve is a visual representation of a classification model's performance at different threshold levels. The curve is created between the next two variables:

- True Positive Rate
- False Positive Rate

The recall is a synonym for TPR, or true positive rate, which can be determined as in equation (7):

$$\text{TPR} = \frac{\text{TP}}{\text{TP+FN}} \quad (7)$$

False Positive Rate, or FPR, can be computed as using equation (8)

$$\text{FPR} = \frac{\text{FP}}{\text{TN+FP}} \quad (8)$$

6. Experimental Results

Table 2 shows the accuracy scores for each model.

Table 2: Accuracy Scores for the Models

Model	Accuracy (%)
Decision Tree	89.17
MLPClassifier	88.65
Random Forest	92.72
XGBoost	94.76
K-Nearest Neighbor	92.24
Logistic Regression	78.69
LightGBM	94.00
CatBoost	95.07

Fig.6 shows the accuracy comparison graph between three machine learning models. As seen in the above chart, Decision Tree gives 89% accuracy, Neural Network gives 88% accuracy, Random Forest gives 92% accuracy, XGBoost gives 94% accuracy, K-Nearest Neighbor gives 92% accuracy, Logistic Regression gives 78% accuracy, LightGBM gives 94% accuracy, and Categorical Boosting gives 95% accuracy respectively. When compared to all the models categorical boosting performed significantly better.

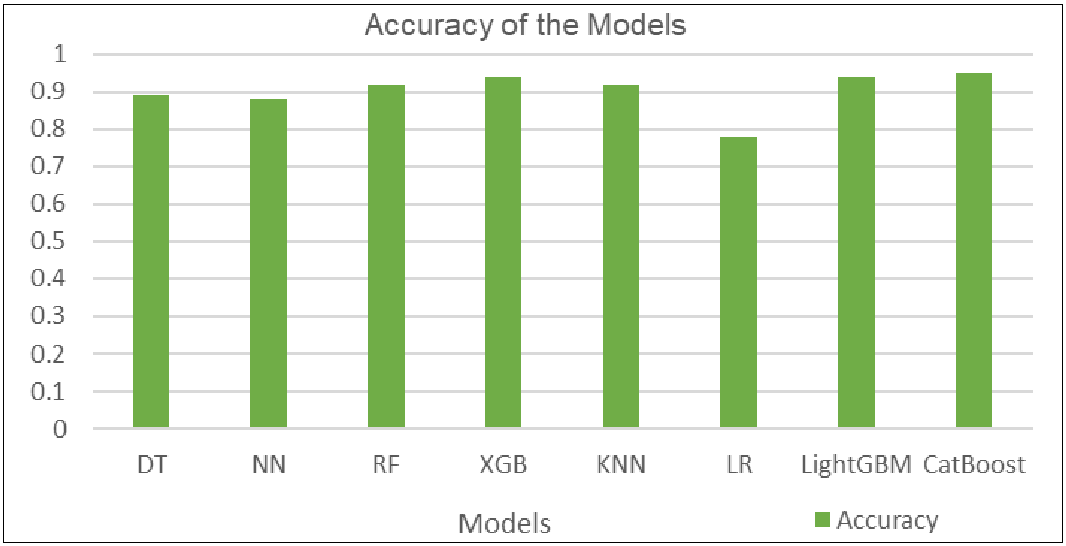

Fig 6: Accuracy of Models

Fig.7. shows the precision comparison graph between eight machine learning models. As seen in the above chart, Decision Tree shows precision scores for No Rain and Rain classes as 92% & 83%, Neural Network gives for No Rain and Rain classes as 90% & 85%, Random Forest gives for No Rain and Rain classes as 94% & 89%, XGBoost gives for No Rain and Rain classes as 96% & 91%, K-Nearest Neighbor gives for No Rain and Rain classes as 97% & 85%, Logistic Regression gives for No Rain and Rain classes as 80% & 74%, LightGBM gives for No Rain and Rain classes as 95% & 92%, and Categorical Boosting gives for No Rain and Rain classes as 98% & 90%, respectively.

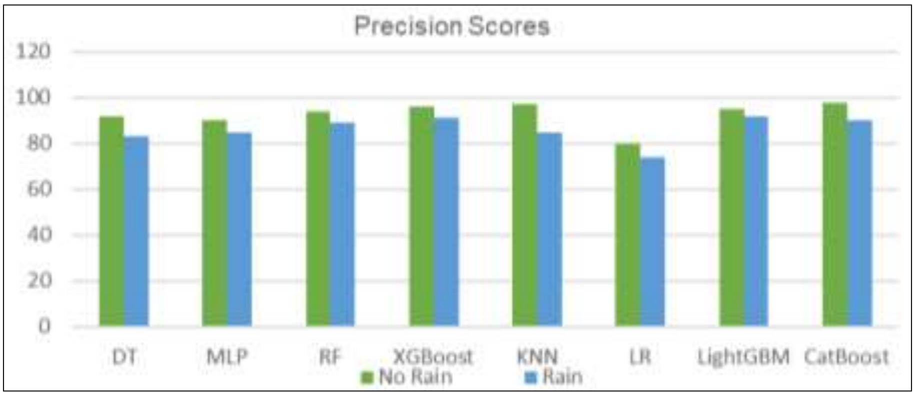

Fig 7: Precision Comparison Graph

Fig.8. shows the recall comparison graph between eight machine learning models. As seen in the above chart, Decision Tree shows recall scores for No Rain and Rain classes as 88% & 89%, Neural Network gives for No Rain and Rain classes as 90% & 85%, Random Forest gives for No Rain and Rain classes as 93% & 91%, XGBoost gives for No Rain and Rain classes as 94% & 94%, K-Nearest Neighbor gives for No Rain and Rain classes as 89% & 96%, Logistic Regression gives for No Rain and Rain classes as 85% & 67%, LightGBM gives for No Rain and Rain classes as 95% & 92%, and Categorical Boosting gives for No Rain and Rain classes as 93% & 97%, respectively.

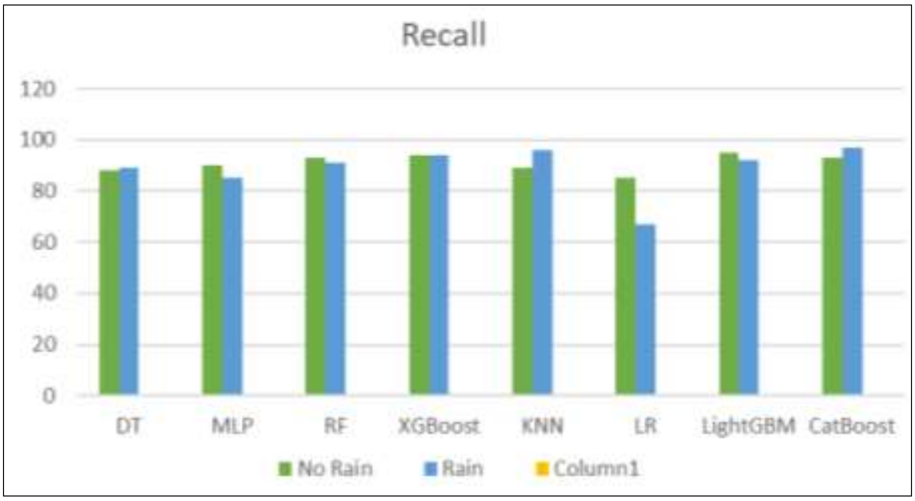

Fig 8: Recall Comparison Graph

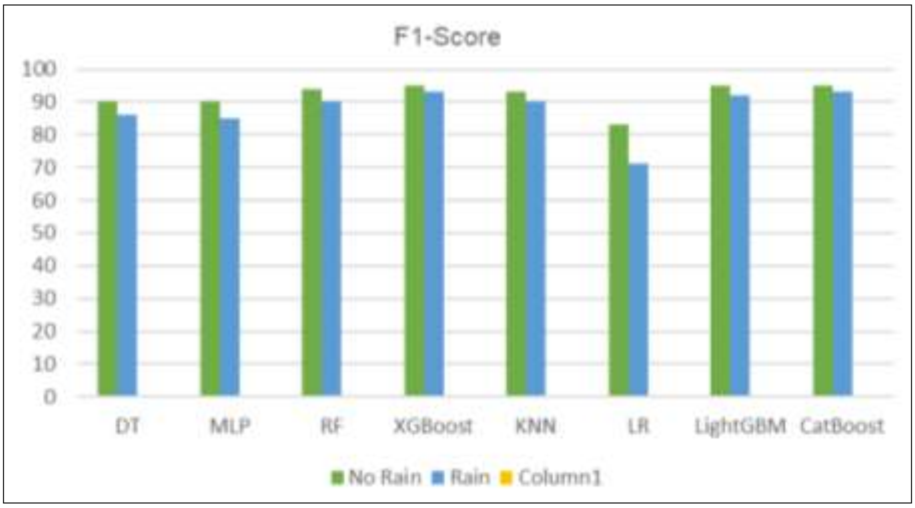

Fig 9: F1-Score Comparison Graph

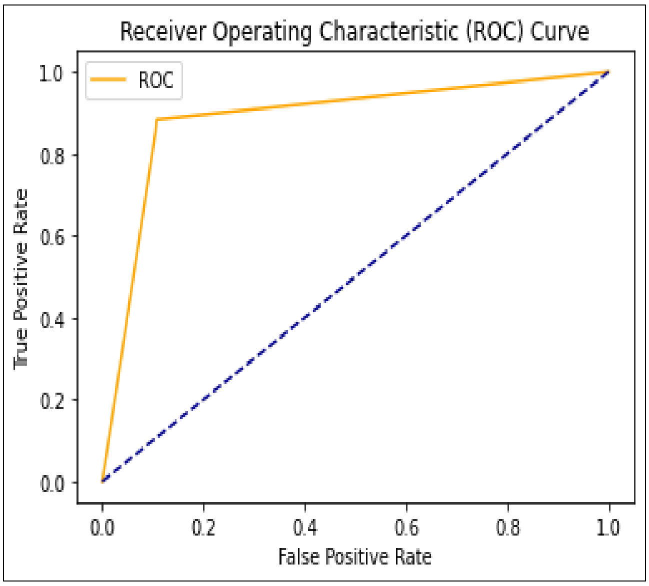

Fig 10: DT ROC Curve

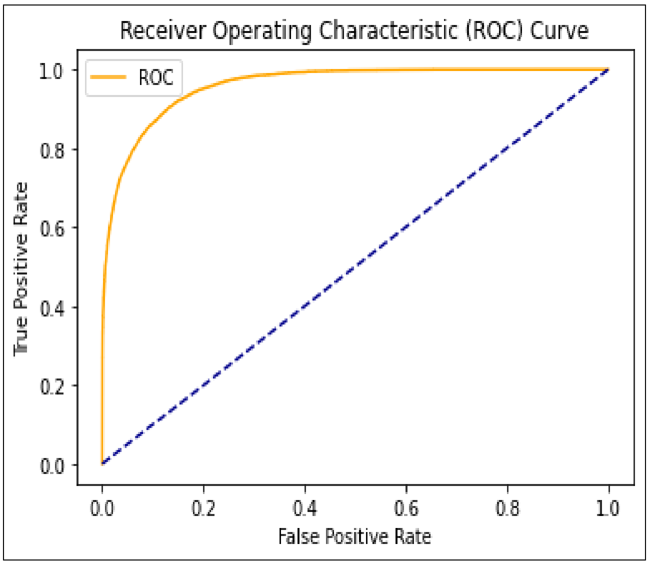

Fig 11: NN ROC Curve

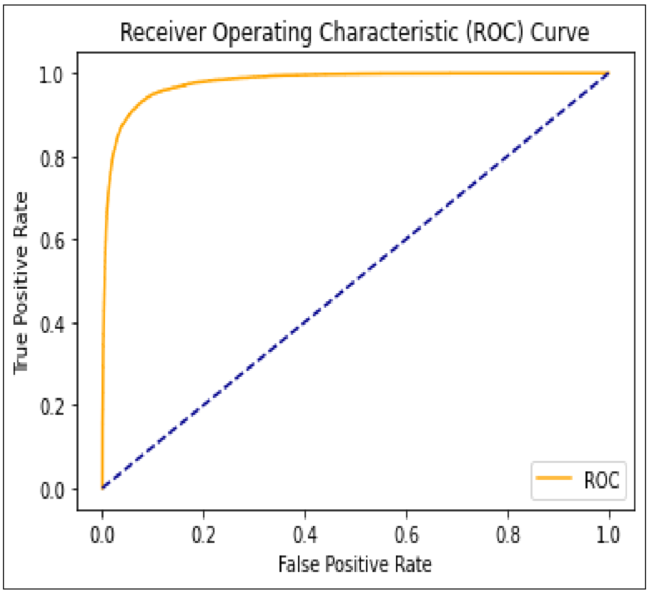

Fig 12: RF ROC Curve

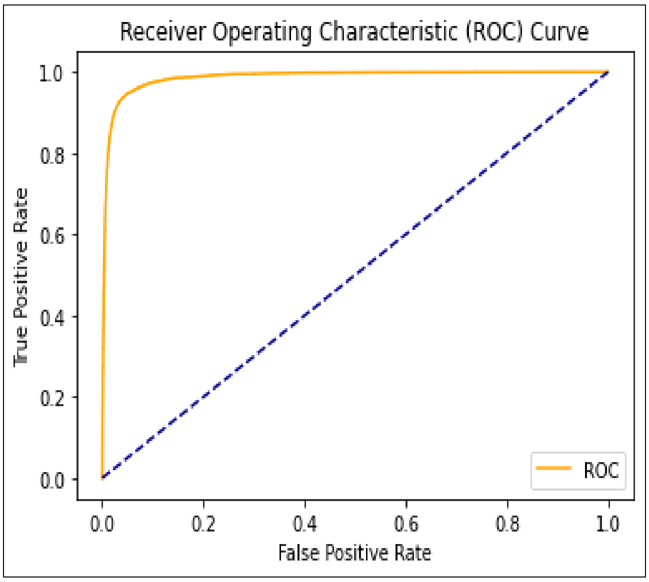

Fig 13: XGBoost ROC Curve

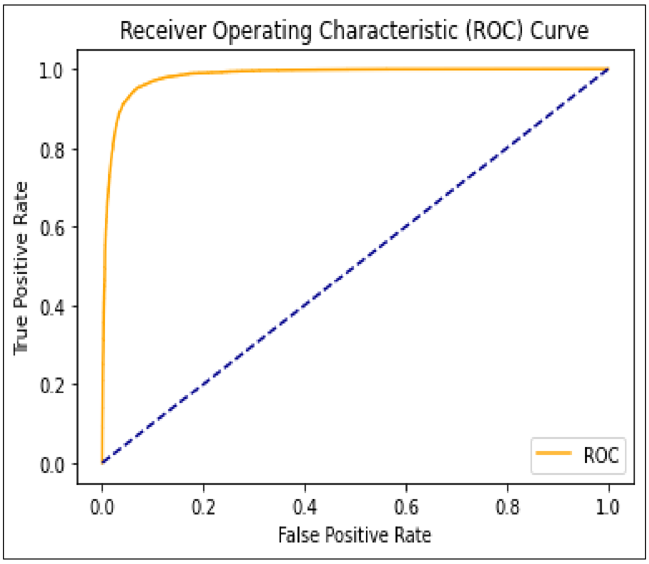

Fig 14: KNN ROC Curve

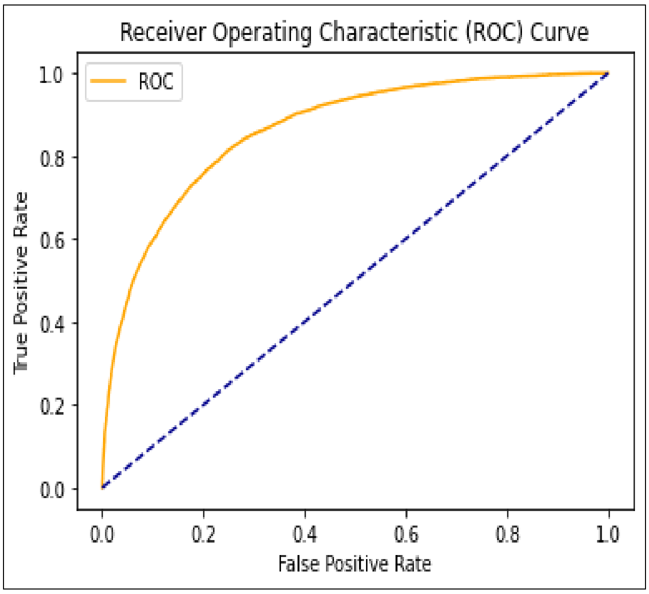

Fig 15: LR ROC Curve

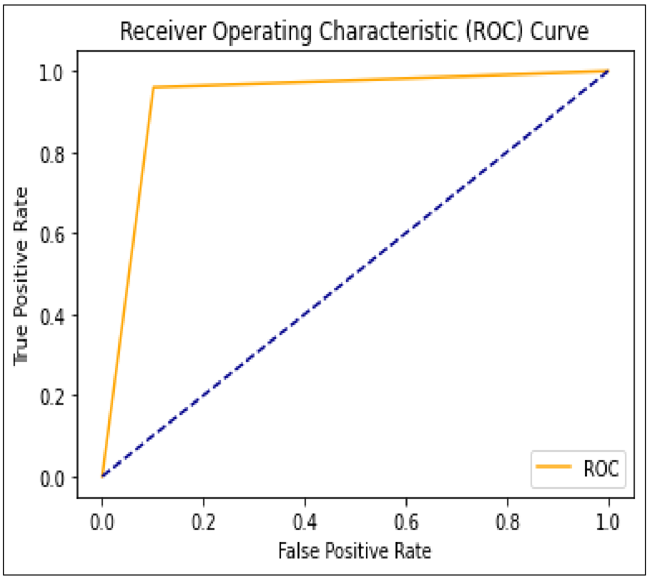

Fig 16: LightGBM ROC Curve

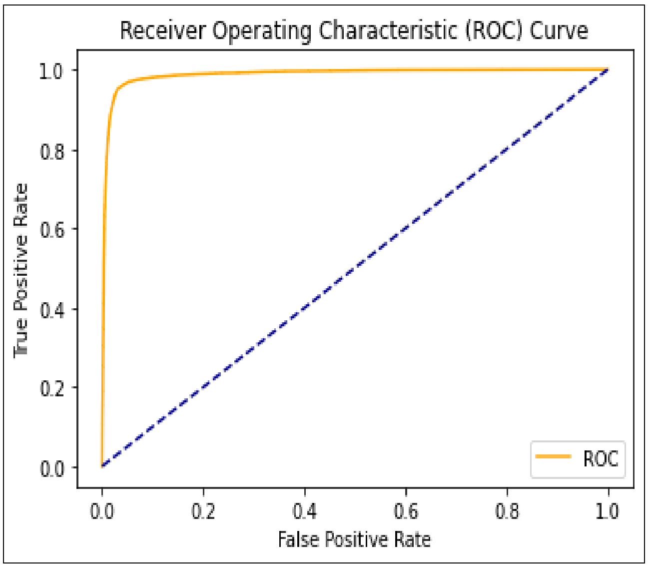

Fig 17: CatBoost ROC Curve

The figures above from Fig.10 to Fig.17, illustrate the ROC curves for various models. These plots display the specificity (FPR) along the x-axis and sensitivity (TPR) along the y-axis for several possible candidate threshold values between 0.0 and 1.0. The diagonally dashed blue line represents random guessing with an AUC value of 0.5. (Random classifier). The ROC values are 0.90 for the decision tree model, 0.87 for the neural network, 0.92 for the random forest, 0.94 for the XGBoost, 0.98 for the K-Nearest Neighbor, 0.76 for the logistic regression, 0.93 for the LightGBM, and 0.95 for the CatBoost, respectively.

7. Conclusion

A thorough examination of the literature has been done in this project to find a good method for predicting rainfall. There was no solid proof that one algorithm was the best strategy for making predictions. The result was the selection of a group of algorithms, including Decision Tree, Neural Network (MLPClassifier), Random Forest, Extreme Gradient Boosting, K

Nearest Neighbors, Logistic Regression, LightGBM, and Categorical Boosting. The WeatherAUS data was used to train the chosen algorithms. To evaluate the accuracies of the models, each algorithm in a machine learning model is trained using a set of records. The trained algorithms were evaluated with the use of accuracy performance indicators. Categorical Boosting, Extreme Gradient Boosting, and LightGBM algorithms outperform the other algorithms in terms of accuracy prediction, scoring 95%, 94%, and 94%, respectively, after the analysis of the results. When compared to the existing system algorithms, the proposed system algorithm called Categorical Boosting performs significantly better at predicting rain.

8. Future Scope

Future research should focus on evaluating the results of employing various weather factors for different stations using a larger data set, and consider the derivatives of characteristics as well as additional features like cloud coverage, etc. for rainfall prediction with longer lead times.

9. References

1. Missah YM, Amekudzi LK, Ussiph N, Frimpong T, Ahene E. Rainfall prediction using machine learning algorithms for the various ecological zones of Ghana. IEEE Access. 2022;10:5069-5082.
2. Manoj O, Ananth JP. MapReduce and optimized deep network for rainfall prediction in agriculture. Comput J. 2020;63(6):900-912.
3. Zhang X, Mohanty SN, Parida AK, Pani SK, Dong B, Cheng X. Annual and non-monsoon rainfall prediction modelling using SVR-MLP: an empirical study from Odisha. IEEE Access. 2020;8:30223-30233.
4. Zhang P, Jia Y, Gao J, Song W, Leung H. Short-term rainfall forecasting using multi-layer perceptron. IEEE Trans Big Data. 2018;6(1):93-106.
5. Kusiak A, Wei X, Verma AP, Roz E. Modeling and prediction of rainfall using radar reflectivity data: a data-mining approach. IEEE Trans Geosci Remote Sens. 2013;51(4):2337-2342.
6. Pucheta J, Rivero CR, Herrera M, Salas C, Sauchelli V. Rainfall forecasting using sub-sampling nonparametric methods. IEEE Lat Am Trans. 2013;11(1):646-650.
7. Khan MI, Maity R. Hybrid deep learning approach for multi-step-ahead daily rainfall prediction using GCM simulations. IEEE Access. 2020;8:52774-52784.
8. Zhao Q, Liu Y, Yao W, Yao Y. Hourly rainfall forecast model using supervised learning algorithm. IEEE Trans Geosci Remote Sens. 2022, 60.
9. Haidar A, Verma B. Monthly rainfall forecasting using one-dimensional deep convolutional neural network. IEEE Access. 2018;6:69053-69063.
10. Haq DZ. Long short-term memory algorithm for rainfall prediction based on El-Nino and IOD data. Procedia Comput Sci. 2021;179:829-837.