International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 10 Issue: 03 | March - 2026 SJIF Rating: 8.659 ISSN: 2582-3930

RainSense – A Machine Learning Based Rainfall Prediction System

J. Durga Prasanna[1],B.Shanmukha Sriram[2] ,CH.Siva Durga Prasad[3] ,S. Subash[4] Mr. Y. Leela Krishna[5], Assistant Professor, Department of CSE (AI & ML)
UG Students, Department of CSE (Artificial Intelligence & Machine Learning)
Kallam Haranadha Reddy Institute of Technology (Autonomous), Guntur, Andhra Pradesh, India
leelakrishna@khitguntur.ac.in

ABSTRACT

Weather forecasting plays a critical role in agriculture, disaster management, transportation, and water resource planning. Among various meteorological factors, rainfall prediction is particularly important as it directly influences crop production, flood control, and environmental management. Traditional rainfall forecasting methods rely on complex meteorological models and manual analysis, which may lead to delayed predictions and limited accessibility for general users. This research proposes RainSense, a machine learning-based rainfall prediction system that integrates data preprocessing, predictive modeling, and a web-based interface to analyze and forecast rainfall occurrence. Data preprocessing techniques are applied to handle missing values, encode categorical variables, and prepare the dataset for model training.

Experimental evaluation shows that weather attributes such as humidity, pressure variation, and prior rainfall indicators significantly influence rainfall occurrence. Among the tested algorithms, the Random Forest model demonstrated superior prediction performance with higher accuracy and reliable classification results. The proposed system provides an accessible and efficient rainfall prediction tool that can assist farmers, planners, and researchers in making informed decisions related to weather-dependent activities.

KEYWORDS : Rainfall Prediction, Machine Learning, Weather Forecasting, Random Forest, Decision Tree, Logistic Regression, Flask Web Application, Exploratory Data Analysis (EDA), Principal Component Analysis (PCA), Predictive Analytics.

I. INTRODUCTION

Rainfall prediction is one of the most important aspects of weather forecasting, as it plays a crucial role in agriculture, water resource management, disaster prevention, and environmental monitoring. Accurate rainfall forecasting helps farmers plan irrigation activities, assists governments in flood control and disaster preparedness, and supports various industries that depend on weather conditions. However, traditional rainfall prediction methods rely on complex meteorological models and manual analysis, which often require extensive computational resources and may not always provide accurate predictions due to the dynamic nature of atmospheric conditions. With the advancement of data science and machine learning techniques, it has become possible to analyze large volumes of historical weather data and identify patterns that influence rainfall occurrence .Machine learning algorithms can automatically learn relationships between meteorological parameters such as temperature, humidity, atmospheric pressure, and wind speed to predict rainfall events. These approaches have shown promising results in improving prediction accuracy compared to conventional statistical methods. This research proposes **RainSense**, a machine learning-based rainfall prediction system designed to analyze weather data and predict whether rainfall will occur on the following day. The system incorporates **Exploratory Data Analysis (EDA)** to understand data patterns and **Principal Component Analysis (PCA)** to reduce dimensionality and improve model efficiency. Several machine learning algorithms, including Logistic Regression, Decision Tree, Random Forest, are used to build predictive models. The trained model is integrated into a Flask-based web application, enabling users to input weather parameters and obtain real-time rainfall predictions through an interactive interface.

II PROBLEM STATEMENT

Accurate rainfall prediction is essential for agriculture, disaster management, and water resource planning, yet traditional forecasting methods often rely on complex meteorological models and manual analysis that may not efficiently capture the nonlinear relationships between atmospheric parameters. These approaches can lead to inaccurate or delayed predictions and are often not easily accessible to farmers and decision-makers who require timely weather insights. Therefore, there is a need for an intelligent and user-friendly system that can analyze historical weather data and provide reliable rainfall predictions.

The proposed RainSense system addresses this challenge by applying machine learning techniques along with data preprocessing methods such as Exploratory Data Analysis (EDA) to improve prediction accuracy. The system analyzes weather parameters including temperature, humidity, wind speed, and atmospheric pressure to identify patterns related to rainfall occurrence. Furthermore, it provides a web-based platform that allows users to easily input weather conditions and obtain real-time rainfall predictions. This approach helps support better planning and decision-making for weather-dependent activities**.**

III. LITERATURE SURVEY

Recent Studies have shown that Researchers have applied various algorithms such as Artificial Neural Networks, Support Vector Machines, Decision Trees, and ensemble learning models to understand the relationship between atmospheric parameters and rainfall occurrence.

Ahmed et al. (2022) proposed a rainfall prediction model using Artificial Neural Networks published in the Journal of Atmospheric and Climate Sciences. Their study used a meteorological dataset containing approximately 100,000 records collected from weather monitoring stations. The model demonstrated the ability to capture nonlinear relationships between weather attributes such as humidity, pressure, and temperature. However, the system required high computational power and lacked a practical deployment platform.

Kumar and Singh (2023) developed a rainfall forecasting approach using Decision Tree and Support Vector Machine algorithms, published in the International Journal of Advanced Computer Science and Applications (IJACSA). Their study utilized approximately 120,000 weather records and focused on improving prediction accuracy through feature selection techniques. Although the model achieved reasonable performance, the work primarily concentrated on algorithm evaluation without providing an interactive application for real-time usage**.**

Zhang et al. (2024) introduced a rainfall prediction system using Random Forest and Gradient Boosting algorithms, published in IEEE Access. Their research analyzed a dataset of approximately 145,000 meteorological records. The results showed that ensemble learning |www.ijirset.com |A Monthly Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710| IJIRSET ©2025 | An ISO 9001:2008 Certified 1113 Journal | methods significantly improve prediction accuracy compared to traditional statistical approaches. However, the system focused mainly on model performance rather than providing visualization and real-time prediction capabilities.

comparative Analysis of Related Work:

Author (Year)	**Dataset Size**	**Methods Used**	**Accuracy**	**Limitations**
Ahmed et al. (2022)	~100K records	Artificial Neural Network	82%	High computational cost
Kumar& Singh (2023)	~120K records	Decision Tree, SVM	85%	No real-time application
Zhang et al. (2024)	~145K records	Random Forest, Gradient Boosting	88%	Focused on model evaluation
Proposed RainSense (2025)	**970K+ records**	**EDA, Logistic Regression, Decision Tree, Random Forest**	**90%**	Web-based rainfall prediction

IV.PROPOSED METHODOLOGY

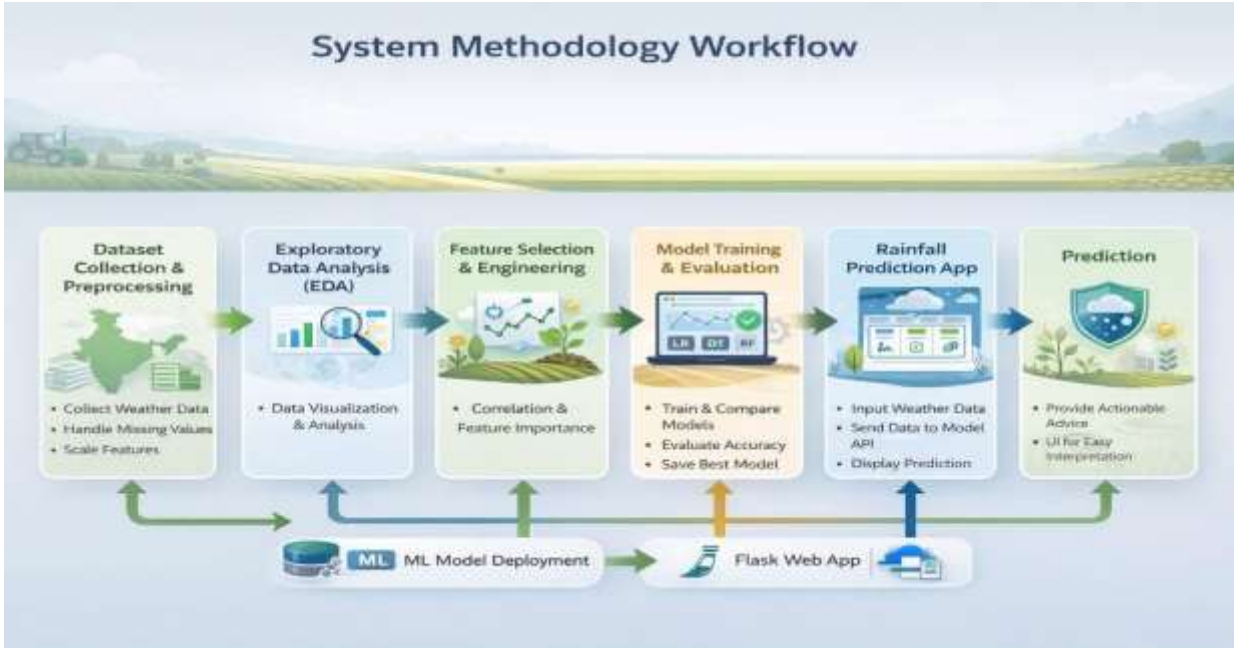

Fig-1:Methodology

Dataset Collection & Data Preprocessing

The dataset used in this project is the **Indian Rainfall and Weather Prediction Dataset**, which contains approximately **970K+ records** of historical weather data.

It includes data from **almost all cities and districts across India**, making it highly comprehensive and suitable for building a robust rainfall prediction model.

Dataset Features

The dataset consists of multiple attributes, including:

- Date of record
- Month and season
- Station name, state, and district
- Temperature (average, minimum, maximum)
- Wind speed
- Air pressure
- Elevation, latitude, longitude
- Rainfall

These features help capture various environmental and climatic conditions influencing rainfall.

Target Variable Creation

Since the dataset did not contain a predefined target variable, a new column named **"Rainfall Tomorrow"** was created using Excel.

Excel Formula Used:

=IF(O2>=2.5,"Yes","No")

Logic:

- Rainfall ≥ 2.5 mm → **Yes (Rain Expected)**
- Rainfall < 2.5 mm → **No (No Rain)**

Threshold Justification

The threshold value of **2.5 mm** was selected based on meteorological standards:

- According to the **India Meteorological Department (IMD)**:
  - 0.1 – 2.4 mm → Very light rain
  - ≥ 2.5 mm → Light rain

This ensures that very small rainfall values (drizzle/noise) are not misclassified as rainfall events.

Data Preprocessing

Before training the machine learning models, the dataset was preprocessed to ensure data quality and consistency.

Handling Missing Values

- Missing values were replaced using **median imputation** for numerical features.

Data Cleaning

- Removed null or invalid records
- Eliminated inconsistencies in data

Data Transformation

- Converted categorical target values:
  - Yes → 1
  - No → 0

Feature Selection

- Selected relevant features for prediction:
  - avg_temp
  - min_temp
  - wind_speed
  - air_pressure
  - rainfall

Data Scaling

- Applied **StandardScaler** to normalize numerical features for better model performance.

Exploratory Data Analysis (EDA)

Exploratory Data Analysis (EDA) is performed to understand the dataset, identify patterns, detect anomalies, and analyze relationships between different features.

In this step, statistical and visual techniques are used to gain insights into the data before applying machine learning models

Data Visualization

- Used **Seaborn and Matplotlib** for visualization

- Plotted:
  - Distribution plots (rainfall, temperature)
  - Scatter plots (feature relationships)

Correlation Analysis

- Generated **heatmap**
- Identified relationships between:
  - Rainfall
  - Temperature
  - Air pressure
  - Wind speed

Helps in selecting important features

Purpose of EDA

- Understand data distribution
- Detect outliers and noise
- Identify important features
- Improve model performance

Feature Selection & Engineering

Feature selection and engineering play a crucial role in improving the performance and efficiency of machine learning models. In this study, both statistical analysis and domain knowledge were utilized to identify the most relevant features influencing rainfall prediction**.**

Feature Selection

Initially, a large number of attributes were available in the dataset, including meteorological and geographical parameters. However, not all features contribute equally to prediction accuracy. Therefore, feature selection was performed to retain only the most significant variables.Correlation analysis was conducted to understand the relationship between independent variables and the target variable (Rainfall Tomorrow). Features showing higher correlation with rainfall were considered important.

Feature Engineering

Feature engineering was applied to transform the dataset into a format suitable for machine learning algorithms.

Target Variable Transformation

The categorical target variable "Rainfall Tomorrow" was converted into numerical format:

- Yes → 1
- No → 0

This transformation enables the model to process the target variable effectively.

Benefits of Feature Selection & Engineering

The application of feature selection and engineering offers several advantages:

- Improves model accuracy by focusing on relevant features
- Reduces overfitting by eliminating noise and redundant data
- Enhances computational efficiency

Model Training

Model training is a crucial step in the machine learning pipeline, where the algorithm learns patterns and relationships from the preprocessed dataset. In this study, the dataset was divided into **training and testing sets** using a standard split ratio of 70:30 to ensure proper evaluation of model performance on unseen data.Before training, **feature scaling** was applied using the **StandardScaler** technique to normalize the input features to perform classification:

Logistic Regression

Logistic Regression is a statistical classification model that estimates the probability of a binary outcome. It is simple, efficient, and works well when there is a linear relationship between input features and the target variable.

Decision Tree

Decision Tree is a non-linear model that splits the dataset into branches based on feature values. It is easy to interpret and can capture complex relationships, but it may suffer from overfitting if not properly controlled.

Random Forest

Random Forest is an ensemble learning method that combines multiple decision trees to improve accuracy and reduce overfitting. It works by averaging the predictions of several trees, making it more robust and reliable for real-world datasets.

Model Performance Comparison Table

Model	Accuracy (%)	Advantages	Limitations
Logistic Regression	89.65%	Simple,fast, interpretable	Assumes linearity
Decision Tree	90.02%	Handles non- linearity, easy to visualize	Prone to overfitting
Random Forest	**90.50%**	High accuracy, reduces overfitting	Slightly complex

Best Model Selection

Based on the evaluation results, the **Random Forest model** achieved the highest accuracy of **90.50%**, making it the best-performing model among the three.

The superior performance of Random Forest is attributed to:

- Its ability to handle complex and non-linear relationships

- Reduction of overfitting through ensemble learning
- Better generalization on unseen data

Therefore, the Random Forest model was selected as the **final model for rainfall prediction**

Web Application Deployment

The trained machine learning model was deployed using a web-based application to provide an interactive and user-friendly interface for rainfall prediction. A lightweight web framework, **Flask**, was used to develop the application due to its simplicity and flexibility in integrating machine learning models. The application allows users to input key weather parameters such as temperature, wind speed, air pressure, and rainfall through a structured form interface.
Once the user submits the input data, it is preprocessed and passed to the trained model for prediction.The model then generates an output indicating whether rainfall is expected on the following day.The prediction results are displayed on the web interface in a clear and intuitive format. The model and preprocessing components were saved using serialization techniques (e.g., joblib) to ensure efficient loading and execution during runtime.

V. RESULTS

The proposed rainfall prediction system was evaluated using multiple machine learning algorithms, including Logistic Regression, Decision Tree, Random Forest. Among these models, the Random Forest classifier achieved highest accuracy of 90.50%,outperforming Logistic Regression (89.65%)

and Decision Tree (90.02%).The improved performance of Random Forest is due to its ensemble learning capability, which effectively captures complex and non-linear relationships between weather parameters and rainfall occurrence. The model demonstrated strong generalization ability and provided consistent predictions on unseen data, confirming its effectiveness for real-world applications.

The developed system offers significant benefits to various stakeholders. For farmers, it enables better planning of agricultural activities such as irrigation, sowing, and harvesting, thereby reducing crop loss and improving productivity. For agricultural planners and policymakers, the system provides data-driven insights that support efficient resource allocation and disaster preparedness. Researchers and meteorologists can utilize the model for further analysis and improvement of weather prediction systems. Additionally, government agencies can use such predictive tools for early warning systems and climate risk management. Overall, the integration of machine learning with a user-friendly application makes the system practical, scalable, and beneficial for multiple stakeholders in the agricultural ecosystem

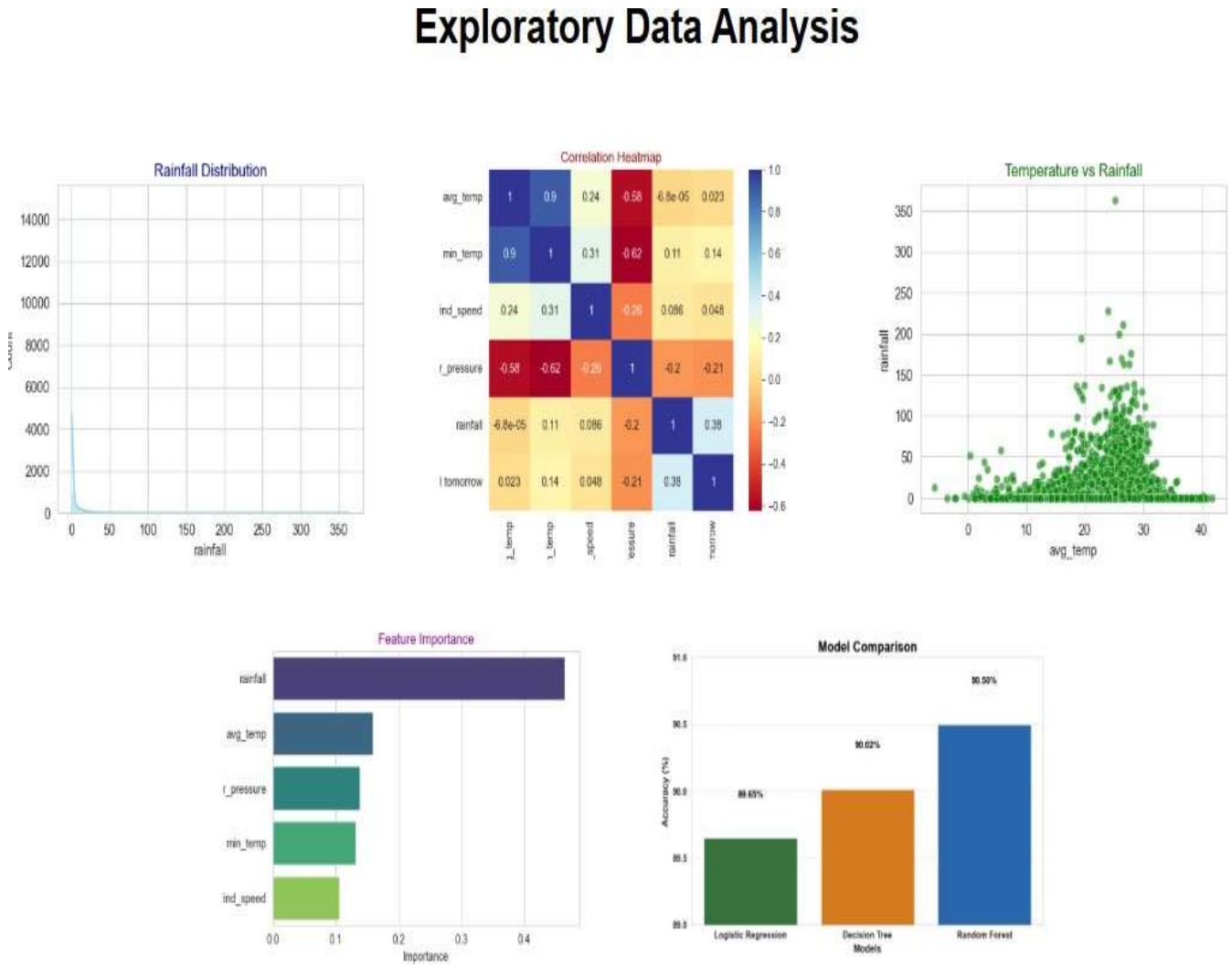

Fig-2:EDA Graphs

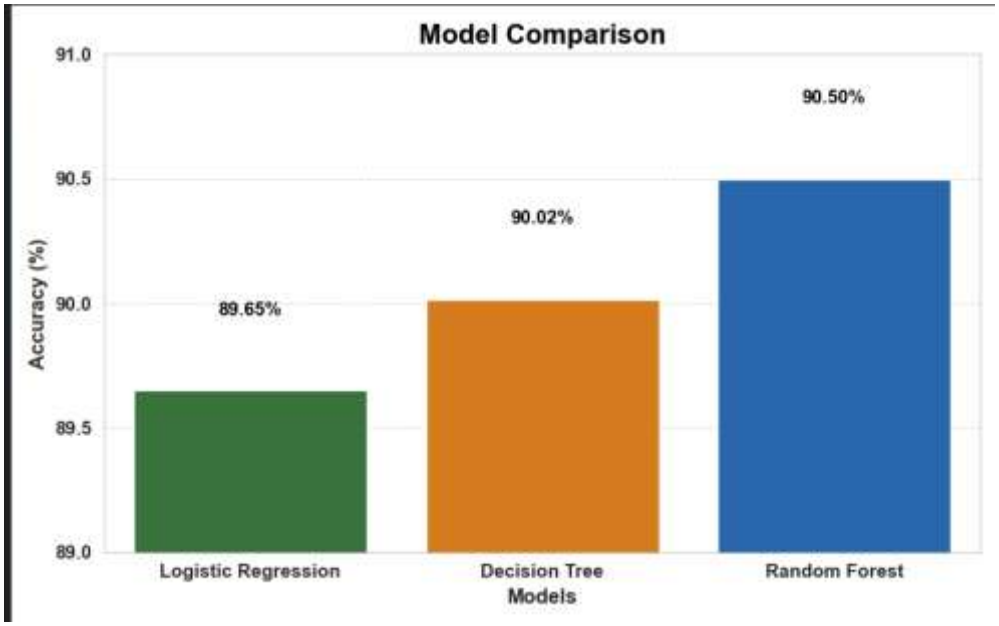

Fig-3:Machine Learning Model Accuracy Comparison Graph

Fig-4:Deployed Web Application Interface With Output

VI. DISCUSSION

The experimental results of the proposed rainfall prediction system demonstrate the effectiveness of machine learning techniques in forecasting rainfall using meteorological data. The results indicate that data-driven models provide more reliable and accurate predictions compared to traditional estimation methods. Interpretation of Predictive Model Performance Among the implemented models, the Random Forest classifier achieved the highest accuracy of approximately 90.50%, outperforming Logistic Regression and Decision Tree. Impact of Weather Parameters The analysis

shows that key meteorological parameters such as temperature, rainfall, and air pressure significantly influence rainfall prediction. Feature importance analysis confirmed that these attributes contribute the most to model performance, validating the feature selection process.

A. Comparison with Traditional Methods

Traditional rainfall prediction methods rely on historical trends and manual interpretation, which may lack accuracy and adaptability. In contrast, the proposed system offers Data-driven prediction ,Automated model selection Improved accuracy, Real-time prediction capability This makes the system more efficient and reliable.

B. Deployment and Scalability

The integration of the trained model into a Flask-based web application enables real-time prediction through a user-friendly interface. The system is scalable and can be extended for cloud deployment or integration with larger weather forecasting systems.

C. Practical Implications

The proposed system provides valuable benefits to stakeholders. Farmers can make better decisions regarding irrigation and crop planning, while policymakers and researchers can use the system for improved agricultural management and climate analysis.

VII. LIMITATIONS

Despite achieving good prediction accuracy, the proposed rainfall prediction system has certain limitations. Firstly, the model relies on historical weather data and does not incorporate real-time meteorological updates, which may affect prediction accuracy under rapidly changing weather conditions. Secondly, the dataset, although large (970K+ records), may contain regional imbalances or missing variations that could impact the generalization of the model across all

geographical locations. Furthermore, the system is implemented as a basic Flask web application, which may have scalability limitations for handling large-scale concurrent users or real-time deployment scenarios. The model performance is primarily evaluated using accuracy, and additional evaluation metrics such as precision, recall, and F1-score were not extensively explored. These limitations indicate areas for further improvement and future enhancement of the system.

VIII. FUTURE SCOPE

The proposed rainfall prediction system can be further enhanced in several ways to improve its accuracy, scalability, and real-world applicability. One major improvement is the integration of **real-time weather data** from APIs and meteorological services, which can make predictions more dynamic and responsive to current atmospheric conditions. Additionally, incorporating more relevant features such as humidity, cloud cover, and satellite-based data can significantly improve model performance.

Advanced machine learning techniques such as **deep learning models (LSTM, ANN)** can be explored to capture temporal dependencies and improve prediction accuracy over time. The system can also be extended to predict **rainfall intensity levels** (light, moderate, heavy) instead of only binary classification, making it more useful for practical applications. From a deployment perspective, the application can be upgraded to a **cloud-based scalable system** with API integration, enabling access for a large number of users. A mobile application can also be developed to provide easy access for farmers in rural areas. Furthermore, integration with **agricultural advisory systems** can help provide recommendations related to irrigation, crop selection, and risk management. These enhancements will make the system more robust, intelligent, and beneficial for the agricultural sector.

IX.CONCLUSION

In this study, a machine learning-based rainfall prediction system was developed using a large- scale Indian weather dataset. The dataset was preprocessed by handling missing values, creating a target variable using a scientifically justified threshold, and selecting relevant features. Exploratory Data Analysis was performed to understand data patterns and relationships, followed by feature selection and engineering to improve model performance**.** Multiple machine learning models, including Logistic Regression, Decision Tree, and Random Forest, were trained and evaluated. Among these, the Random Forest model achieved the highest accuracy of 90.50%, demonstrating its effectiveness in capturing complex relationships between meteorological parameters and rainfall occurrence.

The trained model was successfully deployed using a Flask-based web application, enabling users to input weather parameters and obtain real- time rainfall predictions. The proposed system provides a practical and efficient solution for rainfall prediction, with significant benefits for farmers and other stakeholders. It supports informed decision-making, improves agricultural planning, and reduces risks associated with uncertain weather conditions. Overall, the integration of data preprocessing, machine learning, and web deployment makes the system reliable, scalable, and suitable for real-world applications.

X.ACKNOWLEDGEMENT

The authors express their sincere gratitude to **Mr. Y. Leela Krishna, M.Tech., Assistant Professor,** Department of Computer Science and Engineering (Artificial Intelligence and Machine Learning), Kallam Haranadha Reddy Institute of Technology, for his continuous guidance, constructive feedback, and valuable suggestions throughout the development of this research work. His technical expertise and encouragement played a crucial role in the successful completion of this project.

The authors also extend their heartfelt thanks to the Head of the Department, faculty members, and laboratory staff of the Department of CSE (Artificial Intelligence and Machine Learning) for providing the necessary infrastructure, academic support, and learning environment to carry out this study effectively. Special appreciation is given to all team members for their collaborative efforts, technical contributions, and dedication during dataset preparation, model implementation, visualization development, and deployment through the Flask web application.

Finally, the authors acknowledge the open-source communities and platforms such as Scikit-learn, Python, and Tableau for providing essential tools that enabled the successful implementation of this research.

REFERENCES

[1] L. Breiman, "Random Forests," Machine Learning, vol. 45, no. 1, pp. 5–32, 2001. Available: https://link.springer.com/article/10.1023/A:1010933404324

[2] I. Goodfellow, Y. Bengio, and A. Courville, "Deep Learning."
Available: https://www.deeplearningbook.org/

[3] Scikit-learn Developers, "Scikit-learn: Machine Learning in Python."
Available: https://scikit-learn.org

[4] India Meteorological Department (IMD), "Rainfall and Weather Data."
Available: https://mausam.imd.gov.in

[5] A. K. Sahai, V. Satyan, and V. V. Srinivas,
"Long-range forecasting of Indian summer monsoon rainfall using machine learning techniques," Climate Dynamics, 2013. Available: https://link.springer.com/article/10.1007/s00382-012-1570-0