Quantum and Machine Learning Techniques for Rainfall Prediction

Cheruvu Samivulla[1], Dr. M. Humera Khanam[2], Adoni Ghousia Zeeshan[3], Veluru Monica[4], N. Keerthi[5], Virdala Hema Chandrika[6]

[1,3,4,5,6]*Student, Department of Computer Science and Engineering, Sri Venkateswara University College of Engineering, Tirupati, A.P.*

[2]*Professor, Department of Computer Science and Engineering, Sri Venkateswara University College of Engineering, Tirupati, A.P.*

***Abstract*—Rainfall prediction is a critical challenge in meteorology due to the highly non-linear and complex nature of atmospheric data. This project presents a Comparative Analysis of Classical and Quantum Machine Learning Models to evaluate their predictive accuracy and computational efficiency. While classical models like Support Vector Machines (SVM) and Long Short-Term Memory (LSTM) networks have been standard, this study explores the potential of Variational Quantum Classifiers (VQC). By leveraging quantum entanglement and superposition, the proposed quantum approach aims to identify patterns that classical algorithms may overlook. Our results demonstrate that while classical models remain robust for large datasets, Quantum Machine Learning (QML) shows promising potential for high-dimensional feature mapping in localized rainfall forecasting.**

***Index Terms*—Quantum Machine Learning (QML), Variational Quantum Classifier (VQC), PennyLane, Rainfall Prediction, Meteorological Forecasting, Logistic Regression, Hybrid Quantum-Classical Algorithms, Feature Mapping, Qubit Embedding, NISQ Technology, Comparative Analysis.**

I. INTRODUCTION

Rainfall prediction is a fundamental aspect of meteorological studies, significantly impacting agriculture, water resource management, and disaster preparedness. Traditional forecasting methods often struggle with the chaotic and non-linear nature of atmospheric variables such as humidity, temperature, and pressure. With the recent advancements in Machine Learning (ML), classical algorithms like Support Vector Machines (SVM) and Random Forests have provided more accurate predictive capabilities by learning from historical data patterns. However, as datasets become increasingly high-dimensional, classical computing faces limitations in processing complex feature spaces efficiently. This has led to the exploration of Quantum Machine Learning (QML). By utilizing quantum mechanical properties such as superposition and entanglement, QML models like the Variational Quantum Classifier (VQC) offer a new paradigm for pattern recognition. The primary objective of this project is to conduct a Comparative Analysis between classical ML models and quantum-enhanced approaches. By implementing both systems on rainfall datasets, we evaluate their performance in terms of accuracy, precision, and computational overhead. This study aims to determine if quantum algorithms can provide a superior alternative to classical methods in predicting localized rainfall events, potentially overcoming the "curse of dimensionality" inherent in weather forecasting.

II. RELATED WORK

The advancement of rainfall and flood prediction has been driven by the transition from traditional statistical methods to sophisticated Machine Learning (ML) and Artificial Neural Network (ANN) architectures. Recent studies have demonstrated the efficacy of hybrid models in improving forecasting precision. For instance, pour et al. [1] utilized a hybridized approach merging Random Forest (RF) and Support Vector Machines (SVM), where RF was employed for initial binary classification and SVM for subsequent rainfall emulation. Similarly, the use of ensemble methods, such as the Random Forest ensemble proposed by Vijayan et al. [10], has proven vital in capturing non-linear atmospheric dependencies that simple mathematical abstractions often miss.

Neural network architectures remain a cornerstone of meteorological research. Selvi and Seetha [3] implemented a Feed Forward Neural Network (FFNN) trained with backpropagation, using Mean Square Error (MSE) to quantify predictive accuracy. Further specialized architectures include the Complex Perceptron Neural Network explored by Prabha and Radha [5] for rainfall sequence prediction, and the use of Rectified Linear Units (ReLU) in ANN models to introduce the non-linearity necessary for capturing intricate weather patterns. Regional studies have provided localized insights into monsoon behavior and high-variability climates. Singh et al. [2] integrated entropy-based feature extraction with ANN to address the chaotic nature of India-Singapore Ministerial Roundtable (ISMR) timings. In the North-Western Himalayas, comparative studies between traditional ARIMA models and sequential architectures like Long Short-Term Memory (LSTM) networks have highlighted the superiority of deep learning in mountainous terrains. Additionally, while complex models are often favored, research in the Andaman & Nicobar Islands by Bhaskar et al. [9] found that simple Linear Regression can show superior alignment if feature engineering is robust.

Beyond standalone algorithm, fusion and early warning systems are gaining traction et al. [8] developed a framework integrating multiple supervised models with fuzzy logic to mitigate errors inherent in individual techniques. The practical utility of these models is further evidenced by the work of Athreya et al. [4], who developed an ML-driven "flood card" system to support real-time decision-making and public safety. Finally, benchmarking studies by Kumar et al. [11] across standard classifiers in India identified Random Forest as a highly efficient model for nationwide agricultural planning.

III. PROPOSED SYSTEM: HYBRID QUANTUM CLASSICAL FRAMEWORK

The proposed system addresses the limitations of traditional meteorological models, such as linear decision boundaries and the saturation point of classical accuracy. By moving beyond the binary constraints of standard hardware, this system explores a Hybrid Quantum-Classical (HQC) architecture that utilizes the unique properties of quantum mechanics for complex atmospheric pattern recognition.

1. Baseline Integration
To ensure a rigorous performance benchmark, the system first implements a Logistic Regression model using the Scikit-learn framework. This serves as the classical control group for binary classification (Rain vs. No Rain), establishing the "classical ceiling" for accuracy and computational time against which the quantum model is measured.

2. Quantum-Classical Hybrid Architecture
The core of the proposed solution is a Variational Quantum Classifier (VQC) developed within the PennyLane environment. The system is designed to leverage:

- Angle Encoding: This component acts as a superior representation layer, mapping classical scalars (such as humidity and temperature) into quantum rotations within a high-dimensional Hilbert Space.
- Parameterized Ansatz: The system utilizes Strongly Entangling Layers consisting of tunable gates and CNOT-based entanglement. This allows the model to capture non-linear feature correlations

3. Optimization and Decision Logic
The proposed system utilizes a Hybrid Optimization Loop where:

- Quantum circuits process high-dimensional data and measure expectation values.
- Classical resources calculate the error using a Binary Cross-Entropy cost function and update parameters via the Nesterov Momentum Optimizer.
- This integration allows for improved Decision Boundary Precision, potentially providing a more robust alternative for future complex atmospheric analysis.

IV.DESIGN AND METHODOLOGY

The primary architectural objective of this study is the implementation of a Hybrid Quantum-Classical (HQC) framework to evaluate the efficacy of quantum-enhanced classification against a classical baseline for meteorological forecasting. The methodology is divided into a synchronized dual-path pipeline to facilitate a rigorous comparative analysis.

A. System Overview:

Comparative Analytical Framework

a) Development Environment:

- Anaconda: For environment and dependency management
- Jupyter Notebooks: For model training and simulation

b) Tools & Libraries:

- Classical ML: Scikit-learn (Random Forest, SVM, KNN)
- Quantum ML: PennyLane for quantum circuit simulation
- Data Handling: Pandas and NumPy

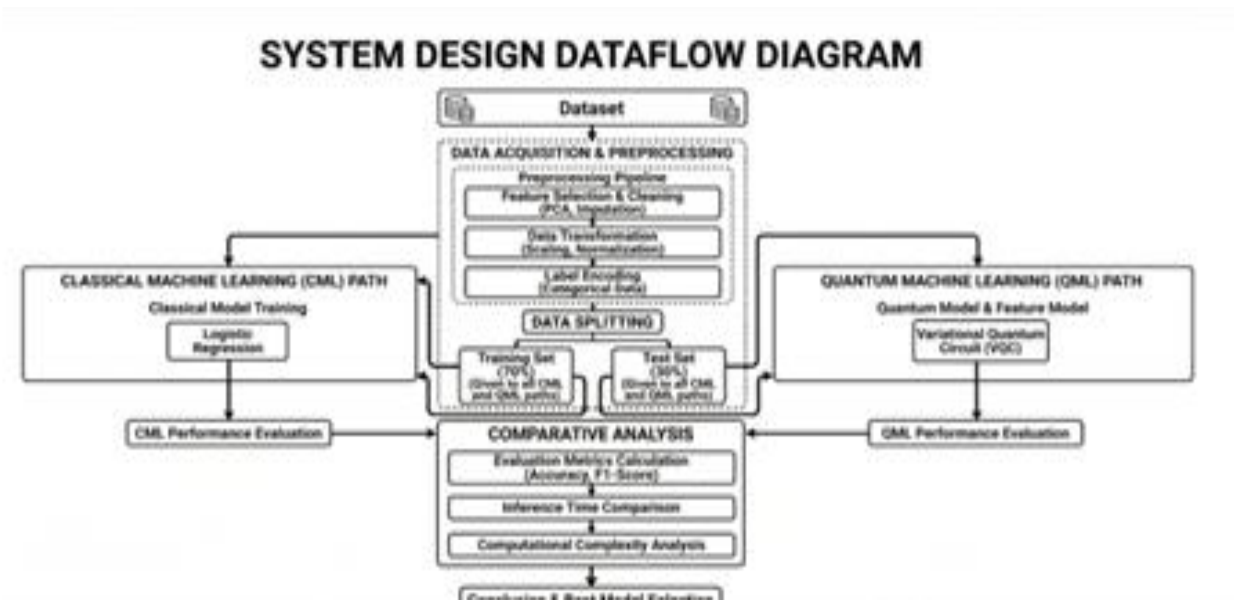

Fig: Methodological Flowchart: Logistic Regression vs. Variational Quantum Circuit (VQC)

1. Data Acquisition and Pre-processing

The research utilizes the weather dataset,to ensure climatological consistency and reduce stochastic noise.

- Feature Selection: Based on thermodynamic relevance, four primary meteorological indicators were selected: Humidity3pm, Pressure3pm, Rainfall (recorded 24-hr prior), and Temp3pm.
- Standardization: To ensure numerical stability within the quantum feature map, input variables were processed using a StandardScaler to achieve a distribution with $\mu = 0$ and $\sigma = 1$
- Dimensionality Management: To accommodate the computational constraints of state-vector simulation on classical hardware, the dataset was truncated to a representative subset of 500 records, partitioned into an 80:20 training-to-testing ratio.
- Target Mapping: Classical binary labels $\{0,1\}$ were transformed into the range $\{-1,1\}$ to align with the expectation values of Pauli-Z quantum measurements.

2. Classical Baseline: Logistic Regression (LR)

A Logistic Regression classifier is employed as the control model. This model establishes the performance ceiling for linear decision boundaries in a classical Hilbert space, providing a benchmark for global accuracy and computational latency.

The empirical model is represented by the following linear equation:

$Z = \beta_0 + \beta_1(x_1) + \beta_2(x_2) + \ldots + \beta_n(x_n)$

where

β_0 : bias term,

β_1, β_2 are the coefficients to each feature,

z: weighted sum,

x_1, x_2 are input features.

$Z = -1.2595 + 1.2682(x_1) - 0.1571(x_2) + 0.3766(x_3) - 0.3227(x_4)$

Where,

X_1: Humidity,

X_2: Pressure,

X_3: Rainfall,

X_4: Temperature.

To derive the final probability of precipitation $\hat{y}$, the weighted sum is passed through the Sigmoid Function:

$\hat{y} = 1 / (1+e^{-z})$

Classical Architectural flow:

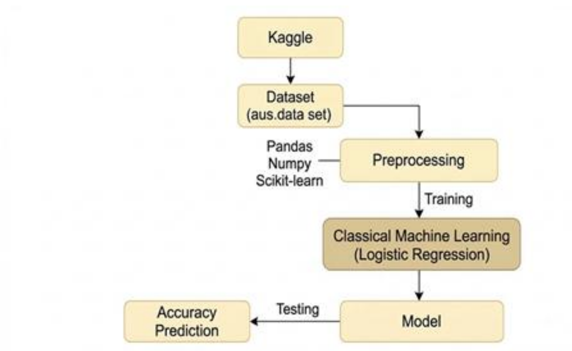

Fig: Classical Architectural Flow

Library Used: Scikit-learn, Pandas, NumPy

Steps:

- Data Input: Load raw meteorological dataset (CSV/Excel)
- Preprocessing: Handle missing values and scale features
- Feature Selection: Identify optimal predictors (Temp, Humidity, etc.)
- Parallel Processing: Send data to Classical and Quantum engines
- Model Comparison: Analyze performance across all algorithms
- Output: Final classification (Rain/No Rain) and Accuracy prediction

Algorithm:

Input: Normalized meteorological feature set $X = \{x_1, x_2, x_3, x_4\}$ representing Humidity, Pressure, Rainfall (24-hr), and Temperature.

Output: Binary classification $y \in \{0, 1\}$ (No Rain / Rain).

1) Initialization: Define the model coefficients β_1, β_2, β_3, β_4 and the intercept β_0.
2) Weighted Summation: Calculate the linear predictor Z by mapping input features to their respective trained coefficients: $Z = \beta_0 + \beta_1(x_1) + \beta_2(x_2) + \beta_3(x_3) + \beta_4(x_4)$
3) Coefficient Mapping: Apply the empirical values derived from the training phase: $Z = -1.2595 + 1.2682(x_1) - 0.1571(x_2) + 0.3766(x_3) - 0.3227(x_4)$
4) Probabilistic Activation: Pass the weighted sum Z through the Sigmoid activation function to squash the output into a range between (0, 1): $y_{prob} = 1/(1 + e^{-Z})$
5) Thresholding and Classification: Apply a decision boundary (typically 0.5) to determine the final class:
   - IF $y_{prob} >= 0.5$, then predict Rain (1).
   - ELSE, predict No Rain (0).
6) Performance Evaluation: Compare the predicted output against the testing subset labels to compute Accuracy, Precision, and Recall.

3. Quantum Variational Classifier (VQC) Architecture

The quantum model is implemented as a Variational Quantum Circuit using the PennyLane framework. The architecture consists of three distinct computational layers:

1) State Preparation: Classical feature vectors χ are mapped into a 2^n dimensional Hilbert space. The system utilizes AngleEmbedding, where each normalized meteorological feature dictates the rotation angle of a specific qubit gate.
2) Variational Ansatz: The circuit utilizes Strongly Entangling Layers. This configuration integrates parameter-dependent rotation gates (R_x, R_y, R_z) and CNOT entanglers to induce quantum correlations between disparate weather variables, such as the non-linear relationship between atmospheric pressure and humidity.

3) Measurement: The network extracts the expectation value of the Pauli-Z operator from the readout qubit, yielding a classical scalar that represents the prediction for Rain Tomorrow.

Quantum Architectural Flow:

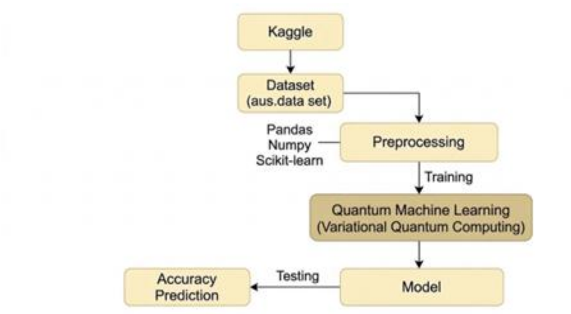

Fig: Quantum Architectural Flow

Library Used: PennyLane

Steps:

- State Preparation: Encode classical data into quantum states (State Loading)
- Ansatz Execution: Apply parameterized quantum gates (Circuit Training)
- Measurement: Collapse quantum states to obtain classical observables
- Classical Optimization: Update gate parameters using a classical optimizer
- Iteration: Repeat loop until the cost function is minimized
- Output: Optimized quantum model for high-precision forecasting

Algorithm:

Objective: Exploration of Hilbert space for high-dimensional, non-linear feature mapping.

1) Quantum Feature Map: Classical scalars are encoded into quantum states. Features (x) are mapped into a high-dimensional Hilbert space by setting the rotation angles of qubits.
2) 2.AnsatzExecution: The parameterized circuit, composed of tunable quantum gates (θ) and entanglement gates (CNOT), processes the data. The entanglement allows the model to identify complex, non-linear correlations.
3) Quantum Measurement: The final quantum state is measured using an observable (measurement operator, $\widehat{M}$), typically in the computational (Z) basis, to retrieve a classical expectation value.
4) Hybrid Cost Evaluation: The quantum output is passed to a classical computer. A Binary Cross-Entropy cost function (Log Loss, similar to the classical path) is used to calculate the error based on the current circuit parameters (θ).
5) Gradient Estimation & Parameter Update: A classical optimizer uses the hybrid cost function to estimate gradients. It determines how the parameters (θ) must change. The parameters (quantum "weights") are updated to minimize the cost.
6) Parameter Iteration Loop: This hybrid loop repeats. Updated parameters are fed back into the quantum circuit (Step 2), and the cost is recalculated until the quantum circuit "learns" the data.
7) Optimal Parameter Testing & Comparison: Once the parameters are optimized, the VQC is tested on the 20% test set. The results are compared directly against the baseline classical model to determine if the quantum architecture provided a performance gain.

4. Hybrid Optimization Loop

The VQC parameters are iteratively updated through a classical feedback loop.

- Cost Function: The system minimizes a Mean Squared Error (MSE) loss between the quantum output and the actual labels.
- Optimizer: Gradient updates are handled by the Nesterov Momentum Optimizer with a learning rate of 0.1.
- Execution Environment: All quantum operations are executed on the default. qubit state-vector simulator, providing a noiseless environment for measuring theoretical quantum utility.

V. RESULT

Evaluation Metrics To quantitatively assess the performance of the proposed Hybrid Quantum-Classical model against the traditional Logistic Regression model, several standard statistical metrics were employed. These metrics are derived from the Confusion Matrix, which tracks True Positives (TP), True Negatives (TN), False Positives (FP), and False Negatives (FN). The metrics utilized in this study are defined as follows:

- Accuracy: This represents the ratio of correctly predicted observations (both rainy and non-rainy days) to the total observations in the Sydney dataset. It serves as a primary indicator of the model's global performance. The Classical model has hit 84.0% while the Quantum model has reached 79.0%.

$$\text{Accuracy} = TP+TN \ / \ TP+TN+FP+FN$$

- Precision: Also known as the positive predictive value, it measures the accuracy of the "Rain" predictions. This metric is critical in weather forecasting to minimize "false alarms" that could lead to unnecessary public alerts. Classical model achieved a high precision of 0.93 for rainy days.

$$\text{R}ecision = TP \ / \ TP+F$$

- Recall (Sensitivity): This measures the model's ability to find all relevant cases within the dataset, specifically how many actual rainy days were correctly identified. High recall is vital for ensuring that actual precipitation events are not missed by the forecasting system.

$$\text{R}ecall = TP \ \ / \ TP+F$$

- F1-Score: Since the weather dataset often has an imbalance between rainy and sunny days, the F1-score provides the harmonic mean of Precision and Recall. It offers a more balanced view of model reliability by penalizing extreme values in either precision or recall, which is essential for the 72:28 class distribution observed in your test set.

$$\text{F1} - Score = 2 * (precision*recall \ / \ precision+recal)$$

- Computational Time: Given the experimental nature of Quantum Machine Learning, the time taken for model training and inference was recorded to evaluate the practical feasibility of the hybrid approach. This metric highlights the current latency gap between classical optimization (0.16s) and quantum circuit simulation (24.97s).

- Mean Squared Error (MSE) Cost: Specific to the Variational Quantum Classifier (VQC), this cost function measures the average squared difference between the quantum circuit's predicted expectation values(y) and the actual target labels(Y).

$$MSE = 1/N * \ \Sigma \, (Y - y)$$

- Optimization Convergence: The evaluation also considers the behavior of the Nesterov Momentum Optimizer. Success is measured by the steady reduction of the cost function across 30 epochs, which in this project dropped from 1.42 to 0.74, indicating effective weight updates within the quantum circuit.

S.NO	MODEL	ACCURACY	PRECISION	RECALL	F1-SCORE	TIME(S)
1	Logistic Regression (classical)	84.0%	0.85	0.84	0.82	0.166891
2	Quantum classifier	79.0%	0.81	0.79	0.77	24.973435

Table: Performance Comparison Metrics

Performance Summary:
As shown in the experimental logs, the metrics were calculated for both models over a test set of 100 samples. The Classical model achieved a weighted average F1-score of 0.82, while the Quantum model demonstrated its learning capability by reducing the training cost from 1.42 to 0.74 over 30 epochs.

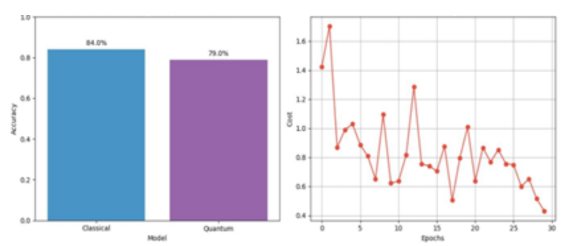

Fig: Comparative Accuracy Analysis and Variational Quantum Circuit (VQC) Training Cost Convergence

VI. CONCLUSION

In our proposed model we made a quantum and machine learning techniques for rainfall prediction with the help of Scikit-learn and PennyLane. We considered a dataset of weather features like Humidity, Pressure, and Temperature and trained using both the Classical Logistic Regression and the Variational Quantum Classifier (VQC) algorithms. Once the training was completed, we performed a detailed comparison to evaluate the performance, accuracy, and computational efficiency of both models. They utilize these different machine learning architectures to check the accuracy of the rainfall prediction and to identify the strengths and limitations of quantum-enhance methods. When the system processes the weather values, the computer compares the results and generates visualization plots to show the differences in accuracy and training cost. The algorithms used in this comparative study gave an accuracy of 84.0% for the Classical model and 79.0% for the Quantum model, concluding that while classical models are currently more efficient for this scale of data, quantum models provide a viable alternative for future complex atmospheric analysis.

VII. FUTURE SCOPE

While this project successfully demonstrates the feasibility of a Hybrid Quantum-Classical (HQC) framework for rainfall prediction, it also identifies several critical avenues for enhancement as quantum hardware and algorithmic research evolve.

A. Expansion of Feature Space and Qubit Scaling
The current model is limited to a 4-qubit architecture mapping to four primary meteorological features.

- High-Dimensional Meteorological Data: Future iterations will move beyond basic variables. By incorporating features such as Wind Gust Speed, Cloud Density, Solar Radiation, and Evaporation rates, the model can develop a more holistic atmospheric profile.
- Dimensionality and Scalability: Transitioning from a 4-qubit system to 16 or 32-qubit architectures will allow for the processing of high-resolution datasets. This avoids the information loss associated with aggressive feature selection, enabling the model to capture subtle correlations in weather patterns.

B. Implementation on Physical Quantum Hardware (NISQ):
This study utilized the default.qubit state-vector simulator to ensure a noiseless environment.

- Deployment on Physical QPUs:A vital next step is to deploy the circuit on Noisy Intermediate-Scale Quantum (NISQ) devices (e.g., IBM Quantum, Rigetti, or IonQ) via PennyLane's hardware-agnostic plugins.
- Quantum Error Mitigation (QEM) : Unlike simulators, physical hardware suffers from decoherence and gate noise. Future research must incorporate QEM techniques, such as Zero-Noise Extrapolation (ZNE) or Probabilistic Error Cancellation (PEC), to maintain prediction accuracy on real-world hardware.

C. Advanced Quantum Architectures and Embeddings:
The current model utilizes a Variational Quantum Classifier (VQC) with Strongly Entangling Layers.

- Quantum Kernel Estimation (QKE): Exploring Quantum Support Vector Machines (QSVM) could uncover non-linear patterns in weather data that are mathematically inaccessible to classical kernels.
- Amplitude Embedding: While Angle Embedding was utilized for its simplicity, future work will explore Amplitude Embedding. This allows for the encoding of 2^n features into n qubits, exponentially increasing data density and allowing the model to handle much larger meteorological datasets.

REFERENCES

[1] S. H. Pour, S. Shahid, E. S. Chung, and X. J. Wang, “Hybridized model merging random forest and support vector machine for rainfall prediction,” Atmospheric Research, vol. 233, p. 104707, 2020.
[2] P. Singh, B. Borah, and M. P. Singh, “Object creation, entropy, and ANN for ISMR forecasting,” International Journal of Information Technology, vol. 10, no. 3, pp. 315–321, 2018.
[3] S. S. Selvi, S. Seetha, et al., “Rainfall prediction using feed forward neural network with backpropagation,” International Journal of Engineering and Advanced Technology, vol. 8, no. 6, 2019.
[4] Y. Athreya, V. B. V., K. Sagar, and S. H. R., “Flood prediction system using machine learning models for early warning,” Journal of Flood Risk Management, vol. 14, no. 2, 2021.

[5] N. D. Prabha and P. Radha, "A complex perceptron neural network for rainfall sequence prediction," in Proc. IEEE Int. Conf. Intelligent Systems, 2020.
[6] "Austin weather dataset study: Rainfall prediction using artificial neural networks and ReLU," Meteorological Data Analytics Report, 2019.
[7] "Comparative study of ML and DL models for North-Western Himalayas rainfall forecasting," Journal of Earth System Science, vol. 130, 2021.
[8] A. Rahman, S. Abbas, M. Gollapalli, et al., "Rainfall prediction system using machine learning fusion with fuzzy logic," Computational Intelligence and Neuroscience, vol. 2022, 2022.
[9] L. Bhaskar, S. M. N., V. S., and T. Anand, "Comparison of regression models for rainfall forecasting in Andaman & Nicobar Islands," MATLAB Applied Research Series, 2020.
[10] R. Vijayan, V. Mareeswari, P. Mohankumar, et al., "Random Forest ensemble for capturing non-linear atmospheric dependencies," International Journal of Computer Science and Network Security, vol. 21, no. 5, 2021.
[11] V. Kumar, V. K. Yadav, and S. Dubey, "Efficiency analysis of ML classification algorithms for rainfall prediction in India," International Journal of Recent Technology and Engineering, vol. 9, no. 2, 2020.