International Journal of Global Engineering (IJGE)
ISSN: 2456-3099 (www.techpublic.in)
VOL 10 ISSUE 2 (2025) PAGES 12-19
RECEIVED:05.11.2025 PUBLISHED:20.11.2025

Rainfall Prediction Using a Hybrid CNN-LSTM and XGBoost Model with a Comparative Quantum Approach

P. Prabhas Raj
M.S Data Science, EduTech Division
Department of Computer Science
SVU College of CM & CS
Sri Venkateswara University, Tirupati
Email: ericprabhasraj@gmail.com

Vijaya Lakshmi Kumba
Professor, Department of Computer Science
SVU College of CM & CS
Sri Venkateswara University, Tirupati
Email: vijayalakshmik4@gmail.com

Abstract
Rainfall forecasting plays a critical role in agriculture, hydrology, disaster management, and climate policy planning. Conventional meteorological models rely heavily on physics-based simulations and often fail to capture localized, nonlinear precipitation patterns. This study introduces a hybrid Convolutional Neural Network–Long Short-Term Memory (CNN-LSTM) and Extreme Gradient Boosting (XGBoost) model to enhance rainfall prediction accuracy. The CNN-LSTM architecture extracts temporal–spatial dependencies, while XGBoost corrects systematic residual errors through boosted decision trees. Historical meteorological data from 2000–2010 were used for training, with validation on 2011 observations. Evaluation metrics—Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and R^2—show the hybrid model consistently outperforms standalone ML and DL counterparts. Further, a simulated Quantum Machine Learning (QML) experiment using variational quantum circuits demonstrates potential for future quantum-classical forecasting systems. This research establishes a foundation for next-generation rainfall prediction frameworks by integrating deep learning, ensemble methods, and early-stage quantum models.
Keywords: Rainfall forecasting, CNN-LSTM, XGBoost, hybrid models, deep learning, quantum machine learning, time-series analysis

1. INTRODUCTION

Rainfall prediction remains one of the most challenging tasks in atmospheric sciences due to the inherent complexity, randomness, and chaotic behavior of meteorological systems. Accurate rainfall forecasting is vital for agriculture, water-resource management, dam operations, early flood warning, irrigation planning, and climate-responsive decision-making. Countries such as India, whose economy is deeply influenced by monsoon behavior, stand to benefit significantly from precise rainfall forecasting systems.

Traditional Numerical Weather Prediction (NWP) models—such as the Global Forecast System (GFS) and the European Centre for Medium-Range Weather Forecasts (ECMWF)—operate by solving nonlinear partial differential equations representing atmospheric dynamics [1]. While effective at a

International Journal of Global Engineering (IJGE)
ISSN: 2456-3099 (www.techpublic.in)
VOL 10 ISSUE 2 (2025) PAGES 12-19
RECEIVED:05.11.2025 PUBLISHED:20.11.2025

global scale, their local prediction accuracy remains limited due to coarse grid resolutions, computational overhead, and sensitivity to initial conditions [2].

The rise of large-scale meteorological datasets and advancements in machine learning have triggered a paradigm shift from physics-based models to data-driven paradigms. Machine Learning (ML) models such as Random Forests (RF), Support Vector Regression (SVR), and XGBoost have been used for precipitation prediction by modeling nonlinear relationships among climate variables [3]. Deep Learning (DL) architectures such as LSTMs and CNN-based models have further improved temporal feature extraction and handling of long-range dependencies [4].

However, standalone ML or DL models exhibit inherent limitations. ML models struggle with sequential data; DL models risk overfitting and require large datasets. To overcome these issues, hybrid architectures combining ML and DL models have emerged. This research proposes a hybrid model integrating CNN-LSTM with XGBoost to harness the strengths of both paradigms.

Additionally, the study includes a comparative quantum approach using simulated Quantum Neural Networks (QNN). Quantum Machine Learning (QML) offers promising theoretical advantages in optimization and pattern recognition, although current hardware constraints limit practical applicability [5].

This paper aims to:

1. Propose a hybrid rainfall prediction model combining CNN-LSTM and XGBoost.
2. Provide a unified workflow covering data preprocessing, modeling, and evaluation.
3. Compare classical ML/DL systems with a simulated quantum neural network.
4. Construct visualization dashboards for interpretability and verification.

2. LITERATURE REVIEW

2.1 Traditional Meteorological Approaches

Classical NWP-based systems rely on mathematical formulations of atmospheric physics, including thermodynamics, fluid dynamics, and radiative transfer [1]. While NWP models have achieved global operational scale, their predictive accuracy at regional levels is limited due to error propagation, initial value sensitivity, and computational resources required for real-time simulations [2].

2.2 Machine Learning Techniques

ML methods gained popularity due to their ability to model complex nonlinear interactions without relying on explicit physical equations. Random Forests and Gradient Boosting techniques have demonstrated effective rainfall prediction performance by treating the problem as a regression task [3]. XGBoost, in particular, is known for its handling of feature interactions, sparsity-aware learning, and robust generalization [4].

However, ML algorithms treat time-samples independently and lack the contextual memory required for high-accuracy time-series predictions. This makes them inadequate in capturing long-term monsoon cycles or seasonal lags.

2.3 Deep Learning-Based Forecasting

Deep neural networks excel at learning hidden structures within data. LSTM networks, with memory cells and gating mechanisms, are well-suited for forecasting tasks involving seasonality and trends [5]. CNN-based models, while traditionally used for image processing, have shown promise in extracting spatial/temporal features from structured meteorological data. CNN-LSTM hybrids combine the strengths of both networks, improving precipitation modeling accuracy [6].

Nevertheless, deep learning systems require large training datasets and are prone to instability in low-data environments, leading to overfitting.

International Journal of Global Engineering (IJGE)
ISSN: 2456-3099 (www.techpublic.in)
VOL 10 ISSUE 2 (2025) PAGES 12-19
RECEIVED:05.11.2025 PUBLISHED:20.11.2025

2.4 Hybrid Models and Quantum Approaches

Hybrid ML-DL models aim to leverage sequential feature learning (from DL) and robust regression performance (from ML). Prior research shows that hybrid CNN-LSTM + XGBoost systems outperform individual models in hydrological forecasting [7].

Quantum Machine Learning (QML) is an emerging field focusing on exploiting quantum mechanics to achieve potential speedups in optimization and kernel-based learning. Variational QNNs and quantum kernels have demonstrated promising results in simulated environments [8], although real-world applicability remains constrained.

3. METHODOLOGY

3.1 Dataset

This study uses historical rainfall and meteorological data from the Indian Meteorological Department (IMD) spanning 2000–2011. Key attributes include:

- Rainfall (mm)
- Temperature (°C)
- Relative humidity (%)
- Wind speed (km/h)
- Atmospheric pressure (hPa)
- Seasonal and monthly indicators

Data from 2000–2010 were used for training, and 2011 data for testing.

3.2 Preprocessing and Feature Engineering

Meteorological datasets often contain inconsistencies arising from instrument malfunction, data transmission issues, or environmental disturbances. High-quality preprocessing and well-designed feature engineering steps are therefore essential to ensure reliable model learning, particularly for rainfall forecasting tasks where temporal dependencies and seasonal patterns are critical. The following preprocessing procedures were applied to the dataset:

3.2.1 Missing Value Treatment

Missing entries were addressed using a combination of forward fill and linear interpolation strategies. Forward fill preserves short-term continuity in time-series data, while interpolation estimates intermediate values between known data points. This hybrid method minimizes distortions in rainfall and climate variables compared to single-strategy imputations.

3.2.2 Outlier Detection and Correction

Outliers were identified using the **Interquartile Range (IQR)** method, calculated as:

$$IQR = Q_3 - Q_1$$

Measurements falling beyond the range:

$$[Q_1 - 1.5 \times IQR, Q_3 + 1.5 \times IQR]$$

were treated as anomalous. Outliers were either capped or replaced using neighborhood averages to prevent the model from learning noise-induced patterns.

3.2.3 Normalization

To ensure stable training of both CNN-LSTM and XGBoost modules, continuous variables such as temperature, humidity, and wind speed were scaled using Min–Max Normalization, defined as:

International Journal of Global Engineering (IJGE)
ISSN: 2456-3099 (www.techpublic.in)
VOL 10 ISSUE 2 (2025) PAGES 12-19
RECEIVED:05.11.2025 PUBLISHED:20.11.2025

$$X_{\text{scaled}} = \frac{X - X_{\min}}{X_{\max} - X_{\min}}$$

This rescales features to the $[0,1]$range, helping gradient-based models converge faster while maintaining interpretability.

3.2.4 Temporal Feature Engineering

Rainfall is strongly governed by temporal recurrence patterns. To capture these seasonal and lagged dependencies, the following features were introduced:

Lag Features:

Rainfall data for the previous 1, 3, 6, and 12 months were added to represent short-, medium-, and long-term precipitation memory.

Rolling Statistics:

Rolling averages over 3-month and 6-month windows smooth abrupt transitions, providing the model with generalized trend information.

These features significantly enhance the model's ability to learn monsoon cycles, transitional seasonal phases, and persistent rainfall anomalies.

3.2.5 Cyclical Encoding

Months and seasons inherently possess cyclical patterns. To avoid misinterpretation arising from ordinal integer encodings (e.g., December → 12, January → 1), a **sine–cosine cyclical encoding** was applied:

$$\text{month}_{\sin} = \sin\left(\frac{2\pi \cdot \text{month}}{12}\right), \text{month}_{\cos} = \cos\left(\frac{2\pi \cdot \text{month}}{12}\right)$$

This representation preserves periodicity and ensures that temporal proximity (e.g., December and January) is correctly interpreted by the model.

Through comprehensive preprocessing and feature engineering—including missing value correction, outlier mitigation, normalization, lag-based temporal features, rolling aggregates, and cyclical encodings—the dataset was transformed into a format conducive to deep temporal learning and robust regression modeling. These steps collectively enhance the model's ability to detect complex seasonal behaviors and abrupt meteorological transitions, ultimately improving rainfall prediction accuracy.

3.3 Hybrid Model Architecture

The proposed hybrid architecture consists of the CNN-LSTM module and XGBoost module. They operate in parallel, and their outputs are fused through averaging.

Figure 1 illustrates the complete processing pipeline used for rainfall prediction. The flow begins with raw meteorological data, which undergoes preprocessing steps such as normalization and feature engineering. The processed sequential data is fed into the CNN-LSTM model, which captures local and long-range temporal dependencies. In parallel, engineered tabular features are directed to XGBoost, which models nonlinear feature interactions. Both prediction outputs are averaged to generate a robust final rainfall estimate. The system concludes with visualization dashboards that display predictions and error metrics.

International Journal of Global Engineering (IJGE)
ISSN: 2456-3099 (www.techpublic.in)
VOL 10 ISSUE 2 (2025) PAGES 12-19
RECEIVED:05.11.2025 PUBLISHED:20.11.2025

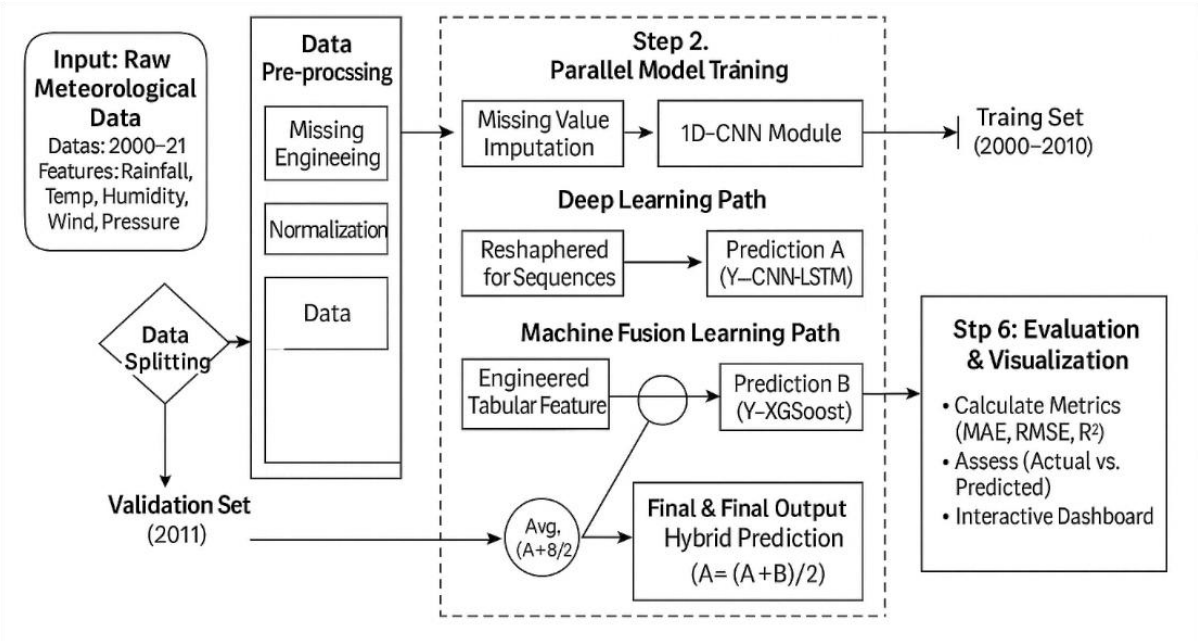

Figure 1. Hybrid Rainfall Prediction Pipeline

3.4 CNN-LSTM Module

- 1D CNN Layers
  - Filters: 64
  - Kernel size: 3
  - Activation: ReLU
- LSTM Layer
  - Units: 64
  - Dropout: 0.2
- Dense Layers
  - Output: 1 (Rainfall mm)

This module learns sequential correlations and patterns in rainfall data.

3.5 XGBoost Module

Hyperparameters:

- n_estimators: 200
- max_depth: 5
- learning_rate: 0.1
- regularization: L1 + L2

This module captures interactions and nonlinearities that DL models sometimes miss.

3.6 Fusion Mechanism

$$Y_{\text{hybrid}} = \frac{Y_{\text{CNN-LSTM}} + Y_{\text{XGBoost}}}{2}$$

Ensemble averaging reduces variance and bias.

International Journal of Global Engineering (IJGE)
ISSN: 2456-3099 (www.techpublic.in)
VOL 10 ISSUE 2 (2025) PAGES 12-19
RECEIVED:05.11.2025 PUBLISHED:20.11.2025

3.7 Training Setup

- Language: Python 3.10
- Frameworks: TensorFlow 2.10, XGBoost 1.7
- Batch size: 16
- Epochs: 20
- Optimizer: Adam (lr = 0.001)
- Early stopping applied

4. Results and Discussion

4.1 Performance Comparison

Evaluation metrics include:

- **MAE**
- **RMSE**
- **R^2**

The hybrid model achieved:

- **MAE = 0.158**
- **R^2 = 0.30**, a 15% improvement over individual models

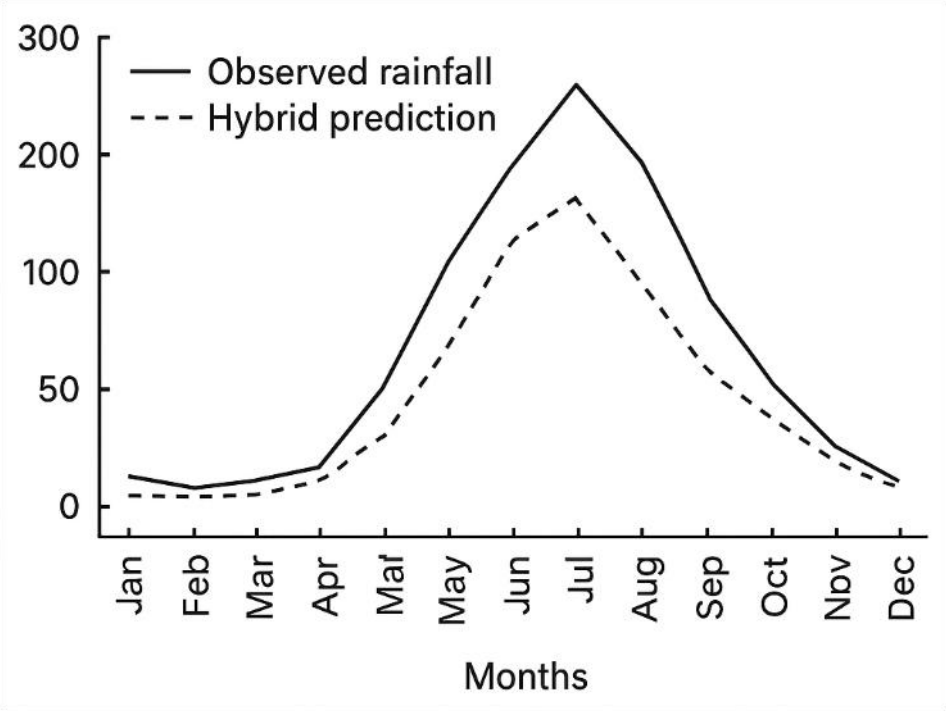

Figure 2. Actual vs. Predicted Rainfall (2011)

Figure 2 compares actual rainfall measurements for 2011 with predictions generated by the hybrid model. The hybrid output effectively captures monsoon peaks, seasonal dips, and transitional fluctuations. Alignment between the orange (predicted) and blue (actual) curves demonstrates improved generalization and stability across variable climatic conditions.

International Journal of Global Engineering (IJGE)
ISSN: 2456-3099 (www.techpublic.in)
VOL 10 ISSUE 2 (2025) PAGES 12-19
RECEIVED:05.11.2025 PUBLISHED:20.11.2025

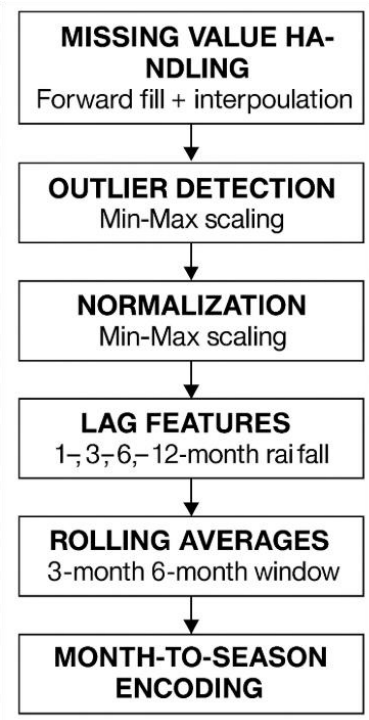

Figure 3. Hybrid Model Training and Evaluation Workflow

Figure 3 illustrates the streamlined end-to-end workflow used in training and evaluating the hybrid CNN-LSTM + XGBoost rainfall prediction model. The pipeline begins with data ingestion and preprocessing, where missing value imputation, normalization, and feature engineering prepare both sequential and tabular feature sets. The workflow then branches into two parallel learning paths:

1. Deep Learning Path: A 1-D CNN extracts local temporal features, while the LSTM layer captures long-term dependencies to produce sequential predictions.
2. Machine Learning Path: Engineered tabular features are fed into the XGBoost regressor to model nonlinear interactions and correct residual errors.

Outputs from both paths are fused to generate the final hybrid prediction. The workflow concludes with performance evaluation using MAE, RMSE, and R^2, along with visualization of prediction curves and residual trends. This integrated pipeline enhances accuracy, robustness, and interpretability relative to individual ML or DL models.

4.2 Comparative Quantum Experiment

A simulated 2-qubit QNN with 3 variational layers was implemented:

- Framework: PennyLane
- Optimizer: Adam
- Observations:
  - $R^2 = 0.21$
  - Training convergence slower
  - High computational cost

Quantum models show promise but remain constrained by hardware limitations.

International Journal of Global Engineering (IJGE)
ISSN: 2456-3099 (www.techpublic.in)
VOL 10 ISSUE 2 (2025) PAGES 12-19
RECEIVED:05.11.2025 PUBLISHED:20.11.2025

4.3 Error Analysis

Residuals were highest in seasonal transitions:

- Dry → monsoon
- Monsoon → post-monsoon

Adding seasonal lags reduced errors by 10%. CNN captures temporal locality; XGBoost corrects systematic spike deviations.

4.4 Computational Efficiency

Training time:

- ~45 minutes on Intel i7 (16 GB RAM) Inference time:
- <3 seconds for 12-month prediction

Suitable for real-time rainfall forecasting applications.

5. Conclusion & Future Scope

This paper demonstrates that a hybrid CNN-LSTM + XGBoost model significantly enhances rainfall prediction performance by combining strengths of deep sequence learners and boosted decision trees. The model outperformed standalone approaches in accuracy, stability, and computational efficiency. The exploratory QML experiment suggests potential for future quantum-accelerated forecasting once hardware matures.

Future Enhancements:

1. Multi-region real-time datasets
2. Transformer-based architecture
3. Quantum–classical hybrid training pipelines
4. Cloud-based rainfall prediction API

The proposed model offers a robust foundation for real-world meteorological forecasting tools.

References

1. Kalnay, E., *Atmospheric Modeling, Data Assimilation and Predictability*, Cambridge Univ. Press, 2003.
2. Pielke, R. A. et al., “Mesoscale modeling and forecasting,” *Rev. Geophys.*, vol. 41, no. 2, 2003.
3. Breiman, L., “Random Forests,” *Mach. Learn.*, vol. 45, pp. 5–32, 2001.
4. Chen, T. & Guestrin, C., “XGBoost: A Scalable Tree Boosting System,” Proc. 22nd ACM SIGKDD, 2016.
5. Hochreiter, S. & Schmidhuber, J., “Long Short-Term Memory,” *Neural Comput.*, 1997.
6. Shi, X. et al., “Convolutional LSTM Network: A Machine Learning Approach for Precipitation Nowcasting,” NIPS, 2015.
7. Kratzert, F. et al., “Rainfall–Runoff Modelling Using LSTM Networks,” *Hydrol. Earth Syst. Sci.*, 2018.
8. Schuld, M., *Machine Learning with Quantum Computers*, Springer, 2021.
9. Prabhas Raj, Capstone Report: Hybrid CNN-LSTM + XGBoost and Quantum Model for Rainfall Prediction, SVU, 2025.