Journal of Innovative Image Processing (ISSN: 2582-4252)
www.irojournals.com/iroiip/

Hybrid Temporal-Spatial Attention Network for Rainfall Prediction in Urban Microclimates

Kumar Chandar S.[1], Sabarmathi G.[2]

[1]Professor, [2]Assistant Professor, School of Business and Management, Christ University, India.

E-mail: [1]kumar.chandar@christuniversity.in, [2]sabarmathi.g@christuniversity.in

Orcid ID: [1]0000-0002-6389-6110, [2]0000-0002-4076-6751

Abstract

Climate change has led to more frequent occurrences of extreme weather conditions and irregular rainfall patterns, making accurate prediction of rainfall amounts essential. In this paper, the performance of a Hybrid Temporal-Spatial Attention Network (HTSAN) model in predicting rainfall for the next day is evaluated using 12 meteorological predictors from the ERA5 reanalysis dataset. The geographical area covered by this research spans a 17 × 17 grid with 0.25° × 0.25° nominal resolution around Bangalore. HTSAN's architecture involved a convolutional feature extractor, recurrent temporal representation, a transformer encoder, and a regression layer. HTSAN's performance was benchmarked against recurrent, convolutional-recurrent, and attention-based approaches using chronological test data. In this, benchmarking, the TFT approach resulted in the best RMSE value of 7.1474, while HTSAN had an RMSE of 7.5911, an MAE of 3.6779, and an R^2 of 0.4259. Extreme-rainfall benchmarking also revealed that HTSAN's critical success index values were lower than those of the comparison approaches at the selected grid-cell rainfall thresholds. Thus, it can be inferred that the current version of HTSAN did not perform better than the selected benchmarks.

Keywords: Rainfall Prediction, ERA5 Reanalysis, Deep Learning, Spatiotemporal Modelling, Transformer Encoder, Extreme-Rainfall Detection.

1. Introduction

Rainfall is the key meteorological factor that directly affects various sectors, including agriculture, the energy industry, water resource planning, and hydropower. In developing countries, crop yield heavily relies on rainfall. For example, about 65% of the total agricultural yield in countries such as India is rainfall-dependent during the monsoon season [1]. Rainfall in catchments is another crucial factor that determines the total global production of hydroelectricity. Adequate rainfall is essential for recharging decreasing groundwater levels. On the other hand, heavy or long-term rainfall causes disasters such as flash floods, drought, urban flooding, and sea-level rise, resulting in the loss of lives and property [2]. This makes the need for dependable and accurate rainfall prediction extremely important. However, accurate rainfall prediction has been one of the most difficult challenges in meteorology due to the complexity and lack of understanding of the mechanisms responsible for rainfall occurrence.

DOI: https://doi.org/10.36548/jiip.2026.4.012
Received: 22.08.2026, received in revised form: 26.09.2026, accepted: 19.10.2026, published: 03.11.2026
© 2026 Inventive Research Organization. This is an open access article under the Creative Commons Attribution-NonCommercial International (CC BY-NC 4.0) License

There exist three major types of approaches that have been used for rainfall forecasting in the literature. First are the physical approaches, where rainfall is forecasted through the simulation of the physical process. But the major disadvantage of these techniques is their high computational overhead and inefficiency. Second are the statistical approaches that focus on analyzing historical data, like ARIMA and multivariate regression. But here, the drawback is that the assumptions of the models are always stationarity, and they have difficulties with non-linearity [4]. Third is the pattern recognition approach, where Machine Learning (ML) models are used to forecast rainfall through feature recognition. Since the atmosphere is very dynamic, ML models are better suited compared to traditional techniques, and the superior performance of these techniques has been proven through multiple research studies [6][7]. Considering the importance of rainfall prediction, there is a need to develop ML based models.

Bangalore suffers from various dangers concerning its infrastructure, water resources, and utility services, owing to changes in rainfall patterns. Uncertain rainfall causes urban flooding, affecting the drainage system, transport, healthcare services, and housing. Water scarcity in Bangalore is a result of high extraction of groundwater and the absence of sustainable surface water resources [8][9]. These problems are exacerbated by urbanization and global warming.

The technology of Artificial Intelligence (AI) has been found to be very helpful in regional gridded precipitation models owing to its unique abilities in forecasting and pattern recognition. AI technology is able to process complicated data to generate a forecast of precipitation for the next day. This kind of forecasting technique assists in creating warning systems that can help in taking preventive measures in flood-prone areas. In view of the issues mentioned above, the integration of AI in urban climate policy is highly significant in Bangalore. The main aim of this research is to create a hybrid model for forecasting rainfall in Bangalore. This research involves evaluating the performance of a new Hybrid Temporal-Spatial Attention Network (HTSAN) developed for forecasting rainfall for the next day. The performance of this method will be evaluated against benchmark methods, rather than assuming that the developed hybrid model inherently increases forecasting accuracy. Daily meteorological data is used to develop and test the model's performance. The scientific contributions of this research are as follows:

- A novel hybrid model is developed to address the complex challenge of rainfall prediction in climate modeling.
- The HTSAN model is proposed for accurate rainfall prediction
- The HTSAN integrates Convolutional Long Short-Term Memory (ConvLSTM) with transformer encoder self-attention mechanism to model the intricate spatiotemporal dynamics of meteorological conditions leading to rainfall events.
- A Spatial tensor generation framework is developed to convert ERA5 meteorological variables into multi-channel spatiotensors
- The reconstructed HTSAN is evaluated on the Bangalore-region ERA5 grid.
- Effectiveness of the model is compared with benchmark models
- A comprehensive ablation study is conducted to analyze the contribution of each architectural component.

2. Literature Review

Rainfall prediction has been widely studied with the use of machine learning, deep learning, and hybrid methods. Previous works have considered nonlinear modeling of rainfall, temporal sequence learning, spatial feature learning, and incorporating meteorological data into the prediction model. The progress made in all these areas is the basis of the proposed HTSAN approach, which focuses on integrating temporal and spatial features for urban rainfall prediction.

Abbot and Marohasy [1] used artificial neural networks for rainfall predictions via long-term time series and optimization in one month. This study demonstrates the significance of past observations and model optimization. Pham et al. [2] applied advanced artificial intelligence models for daily rainfall predictions, whereas Zhou et al. [3] used machine learning models for long-term monthly rainfall predictions through climate and meteorological factors. Animas et al. [5] also used modern machine learning techniques for rainfall time series predictions. Similarly, Kumar et al. [10] used machine learning models for rainfall predictions in urban metropolitan cities. The use of data-driven models in the context of rainfall predictions is evident through the above studies. However, their capability to capture spatial dependencies in urban microclimates needs to be explored.

Recurrent neural networks have also been used in rainfall prediction owing to their effectiveness in dealing with sequences. Kang et al. [4] looked into recurrent neural networks in the prediction of precipitation in Jingdezhen, China. The study by Tanvir and Naiema [16] analyzed LSTM based rainfall prediction while Wijaya et al. [18] compared GRU and LSTM techniques in predicting rainfall in Palembang City. Yang et al. [17] looked at a BiLSTM model trained using the improved whale migration algorithm and variational mode decomposition. These are some of the different ways of analyzing the temporal dependencies and prediction of rainfall time series. Nonetheless, recurrent models that mainly work on sequences will need other mechanisms to handle rainfall data from various urban locations together.

Hybrid deep learning models were considered for enhancing the representation of complex temporal and spatial patterns. Wei and You [7] examined the problem of monthly rainfall prediction with the help of a hybrid neural network model involving discrete wavelet transformation and deep learning techniques. Liu et al. [15] proposed a ConvLSTM framework for rainfall nowcasting using reflectance and wind field data retrieved from radar. The work by Upadhyay et al. [19] focused on 3D convolutional LSTM networks for rainfall nowcasting based on satellite images. Another relevant application to time series is that by Livieris et al. [14], who designed a CNN-LSTM network for forecasting gold price movements. While this application involves a different domain of predictions, the hybrid nature of this methodology still constitutes an example in terms of methods design. Thus, the consideration of hybrid architectures in the context of weather conditions and rainfall distribution evolution motivates the current research.

Furthermore, attention-based and transformer networks provide extra options for modeling dependencies in complex meteorological time series. Piran et al. [20] researched transformers in the context of generative models and transfer learning for precipitation nowcasting. Jun and Kim [21] examined Informers for temperature forecasting using observational and numerical weather prediction data. Despite the difference between temperature forecasting and rainfall prediction, this research may be considered in the context

of wider applicability of attention-based models and meteorological data analysis. Such models give the opportunity to study more thoroughly the mechanisms of finding the most important time moments and features for predicting rainfall.

Predicting rainfall is also crucial in relation to applications in smart cities and urban floods. Rahman et al. [6] presented a system for predicting rainfall using machine learning fusion for smart cities, linking data-driven predictions with urban environmental management. Whereas Kumar et al. [10] emphasized rainfall predictions in metropolitan cities, Ahamed and Asif [8] analyzed flood risk assessments in Bangalore. Meanwhile, Mahadeva et al. [9] described urban floods in Bangalore and their environmental impact. These sources prove the practical value of enhancing rainfall predictions for the urban environment, which would considered localized rainfall and its possible contribution to flood risk management. Moreover, Subramanian et al. [22] researched a complex approach to ambient air pollution prediction in the National Capital Region. While the problem of air pollution prediction is related to a different environmental object, it provides certain context to data-driven environmental predictions in the urban environment.

Meteorological datasets constitute an additional element in the development of forecasting systems. This aspect was elaborated by Hersbach et al. [13], who described the ERA5 hourly dataset, which provides historical meteorological data via the Copernicus Climate Change Service Climate Data Store. Such data can be used in analyzing the temporal changes in weather patterns and, according to the chosen parameters and resolution, can be used in analyzing the spatial patterns pertinent to rain prediction. Preprocessing, selecting the right features, and temporal validation of the input data are crucial.

2.1 Research Gap

The current rainfall prediction models concentrate on either time-based or spatial models. LSTM and BiLSTM models work well when dealing with temporal learning, but they cannot handle complicated spatial relations among weather variables. ConvLSTM works well in modeling localized spatiotemporal models but cannot handle long-distance spatial relations. In addition, some current models use images from radar and satellite data, which are not always present in urban areas where data availability is scarce. Thus, there is a need for a framework that can jointly model temporal and spatial relations using only meteorological variables.

3. Proposed Methodology

3.1 Overview of the Proposed Framework

The research aims to forecast precipitation for the next day over a regional grid around Bangalore, India, using meteorological variables provided by ERA5 reanalysis data. It is a spatiotemporal regression problem where 14 days of meteorological data are used to predict the next day's precipitation field. Input data have 12 predictor variables on 17×17 grids, and the output also has 17×17 grids. The pipeline consists of several stages: data preparation, daily-field formation, relative humidity recovery, training-only normalization, chronological sequence formation, training, and testing.

3.2 Study Area and Dataset

The study uses ERA5 reanalysis data covering the period from 1 January 2016 to 31 December 2025 [23]. The spatial domain extends from 11°N to 15°N latitude and 75°E to

79°E longitude, at a nominal spatial resolution of 0.25° × 0.25°. The resulting grid contains 17 latitude points and 17 longitude points, providing 289 grid cells for each daily meteorological field.

Twelve meteorological variables are used as predictors: 10-metre zonal wind (U10), 10-metre meridional wind (V10), 2-metre dew-point temperature (D2m), 2-metre air temperature (T2m), surface pressure (Sp), total cloud cover (Tcc), boundary-layer height (Blh), convective available potential energy (Cape), total column water vapour (Tcwv), evaporation (E), convective precipitation (Cp), and relative humidity (RH). Relative humidity is reconstructed from air temperature and dew-point temperature. Total precipitation (Tp) is treated as the prediction target rather than as an additional predictor.

These variables represent atmospheric circulation, thermodynamic conditions, moisture availability, cloud characteristics, boundary-layer processes, and precipitation-related processes. Their use provides a multivariable representation of the meteorological conditions associated with rainfall development. The meteorological variables and their respective roles in the prediction framework are summarized in Table 1.

Table 1. Meteorological variables used as predictors and the precipitation target

Variable	Description	Role
U10	10-metre zonal wind	Predictor
V10	10-metre meridional wind	Predictor
D2m	2-metre dew-point temperature	Predictor
T2m	2-metre air temperature	Predictor
Sp	Surface pressure	Predictor
Tcc	Total cloud cover	Predictor
Blh	Boundary-layer height	Predictor
CAPE	Convective available potential energy	Predictor
TCWV	Total-column water vapour	Predictor
E	Evaporation	Predictor
Cp	Convective precipitation	Predictor
RH	Relative humidity reconstructed from T2m and D2m	Predictor
Tp	Total precipitation	Target

3.3 Daily Data Preparation

Hourly ERA5 records were processed into daily meteorological fields for each grid cell. Timestamp consistency, duplicate records, spatial-grid alignment, and missing or non-finite values were checked before sequence construction. Instantaneous variables were aggregated using the operations specified in the preprocessing implementation, while accumulated variables were processed according to their temporal accumulation conventions.

Relative Humidity (RH) was reconstructed from 2-metre air temperature (T2m) and dew-point temperature (D2m). Both temperatures were converted from kelvin to degrees Celsius before applying the Magnus approximation in Equation (1). RH was then estimated using Equation (2), expressed as a percentage, and checked for missing or non-finite values. The resulting daily fields were aligned on the common 17 × 17 grid and used to construct the predictor sequences and next-day precipitation targets.

Relative humidity is derived from T2m and D2m using a temperature-dependent

saturation-vapour-pressure relationship. For temperatures expressed in degrees Celsius, the Magnus approximation can be written as

$$e_s(T) = 6.112\, exp(\frac{17.67T}{T + 243.5}) \tag{1}$$

where e (T) denotes saturation vapour pressure in hPa. Relative humidity is then estimated as

$$RH = 100\, \frac{e_s(T^d)}{e_s(T)} \tag{2}$$

where T^d is dew-point temperature and T is air temperature. Temperatures supplied in kelvin must first be converted to degrees Celsius. The resulting RH values are expressed as percentages and checked for missing or non-finite values.

3.4 Normalization and Sequence Construction

The meteorological predictor variables are normalized using min–max scaling. For a variable x, the normalized value is calculated as follows:

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}} \tag{3}$$

where x_{min} and x_{max} are estimated from the training period only. The same training-derived parameters are applied to the validation and test sets. This procedure prevents the use of validation or test distribution statistics during preprocessing. Values outside the training range may consequently fall below zero or above one in later periods.

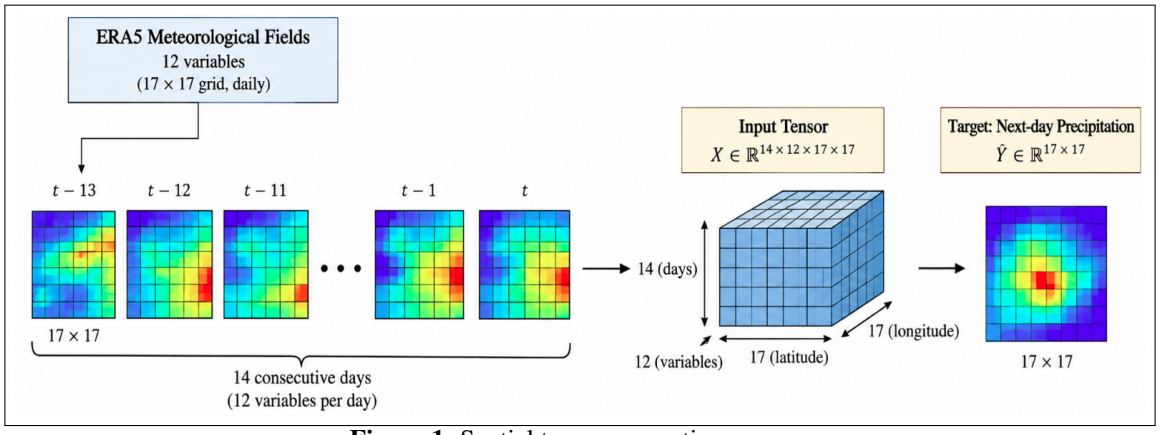


Figure 1. Spatial tensor generation process

The daily meteorological fields are organized into multivariate sequences, preserving their temporal order and spatial grid structure, as illustrated in Figure 1. The verified sequence pipeline uses normalized precipitation values as the prediction target and does not apply a logarithmic target transformation. Therefore, no inverse logarithmic transformation is required when calculating metrics from these particular experiments. Any experiment that uses a different target transformation must be identified and not evaluated separately.

A sliding window technique is applied to create supervised learning samples. The inputs consist of 14 consecutive days of meteorological fields, whereas the target will be the precipitation field of the next day. If the input is on day t, then the target will be the daily precipitation field of t + 1.

The input and target tensors are represented as

$$X_t \in \mathbb{R}^{14 \times 12 \times 17 \times 17}, \quad Y_t \in \mathbb{R}^{17 \times 17} \tag{4}$$

The stored sequence arrays use the sample dimension first. Their verified shapes are N × 14 × 12 × 17 × 17 for the inputs and N × 17 × 17 for the targets

3.5 Chronological Training, Validation, and Test Sets

The data were separated into consecutive time periods to maintain temporal order. Table 2 presents the date ranges and number of sequences in each of the training, validation, and test sets. The training set was used to compute normalization statistics and model weights. The validation set was used for model selection and early stopping when employed, and the test set was used for evaluation purposes. This time-based split substitute the previously described 70%, 15%, 15% train-validation-test random split.

Table 2. Chronological training, validation, and test partitions

Subset	Date range	Number of sequences
Training	1 January 2016–31 December 2022	2,543
Validation	1 January 2023–30 June 2024	533
Testing	1 July 2024–31 December 2025	535

Sequence windows were constructed in chronological order. The split-specific input and target indices must be checked against the saved sequence arrays to confirm that no target is shared across partitions.

3.6 HTSAN Architecture

The overall architecture of the model along with the processing of meteorological sequences to obtain a next-day rainfall map is shown in Figure 2. HTSAN is proposed to integrate the process of spatial feature extraction, representation learning of time, and interactions between features using attention mechanisms.

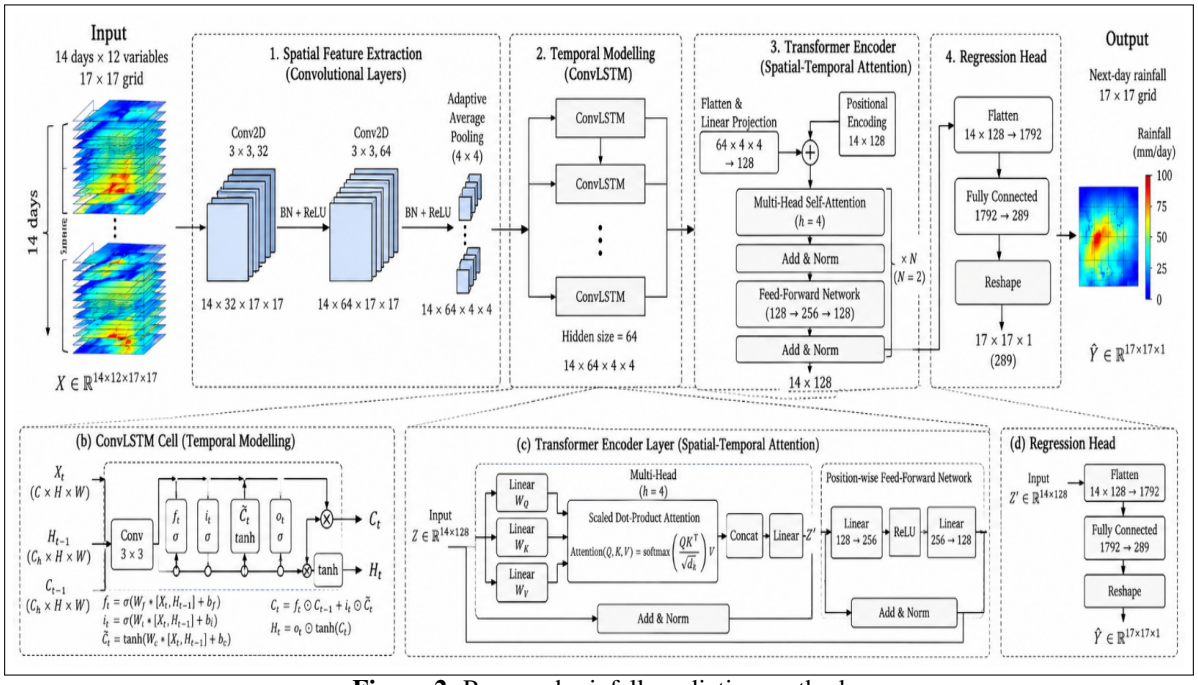


Figure 2. Proposed rainfall prediction method

In the reconstructed implementation used in the latest HTSAN experiment, the network contains two convolutional feature-extraction stages, adaptive average pooling, dropout, an LSTM-based temporal module, a feature projection layer, a Transformer encoder, and a regression head. The reconstructed model has 1,848,897 trainable parameters. This parameter count applies only to that reconstructed implementation and must not be attributed to the original implementation without verification. The principal architectural settings of the reconstructed HTSAN implementation are summarized in Table 3.

Table 3. Configuration of the reconstructed HTSAN implementation

Component	Reconstructed configuration
Input	14 × 12 × 17 × 17
Convolution 1	3 × 3; 32 filters; batch normalization; ReLU
Convolution 2	3 × 3; 64 filters; batch normalization; ReLU
Pooling	Adaptive average pooling to 4 × 4
Dropout	0.2
Temporal module	LSTM; 64 hidden units
Projection	64 to 128 dimensions
Transformer	2 layers; 4 heads; feed-forward dimension 2,048
Regression head	128 → 512 → 289
Output	17 × 17 precipitation field
Parameters	1,848,897 in reconstructed implementation

3.6.1 Convolutional Feature Extraction

The convolutional layers transform the daily meteorological input fields into higher-dimensional feature maps. Batch normalization and ReLU activation are applied in the reconstructed configuration. Adaptive average pooling reduces the spatial dimensions to 4 × 4, and dropout is used with a probability of 0.2. The resulting features are passed to the temporal module.

3.6.2 Temporal Representation

The reconstructed model uses an LSTM with 64 hidden units to encode temporal dependencies in the extracted feature sequence. Its hidden representation is projected into a 128-dimensional feature space before being supplied to the Transformer encoder.

3.6.3 Transformer Encoder

The reconstructed configuration uses a two-layer Transformer encoder with four attention heads and a feed-forward hidden dimension of 2,048. The encoder applies self-attention to the projected feature tokens, allowing the representation at one token to incorporate information from other tokens in the sequence. For query, key, and value matrices Q, K, and V, scaled dot-product attention is defined as

$$Attention(Q,\, K,\, V) \,=\, softmax(\frac{QK^T}{\sqrt{d_k}})V \tag{5}$$

where d is the key dimension. Multi-head attention combines several attention projections to learn complementary relationships between feature tokens.

The manuscript should characterize this module as learning global relationships among the implemented tokens. A stronger claim that it captures global spatial dependencies is justified

only if the token construction and positional information preserve the spatial locations and the attention is applied over those spatial tokens.

3.7 Model Training and Selection

The models are trained on the training data and validated with the help of the validation data. Model selection is done according to the results of validation, and finally, the held-out test set is used for testing the selected model(s). The different training seeds, optimizers, learning rate, batch size, loss function, max epoch and stopping criteria must be specified separately for each model.

The original manuscript specifies AdamW with a learning rate of 10^{-4}, Huber loss, 50 epochs, and a seed of 42. The HTSAN model achieved its lowest recorded validation loss at epoch 8. Best checkpoint epoch number is different from the maximum possible training epoch numbers.

The current verified sequence pipeline does not include the log transform for the rainfall targets. Hence, the training and evaluation procedure description needs to be consistent with this type of target normalization. In addition, training and inference time need to be reported in case they were measured under documented conditions.

3.8 Comparative Evaluation

HTSAN is benchmarked against other methods like recurrent networks such as LSTM, BiLSTM, GRU, and ConvLSTM and attention-based methods such as Transformer, Informer, Autoformer, Time Fusion Network (TFT), and reconstructed ST-GNN for which verified results are available.

3.9 Evaluation Metrics

The accuracy of predictions is assessed based on the following criteria: root mean squared error (RMSE), mean absolute error (MAE), mean bias, and the coefficient of determination. Nash-Sutcliffe efficiency (NSE) can also be given, if calculated. For predictions p and observations y , the metrics are defined as

$$RMSE = \sqrt{(\frac{1}{N}\sum_{i=1}^{N}(p_i - y_i)^2)} \tag{6}$$

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|p_i - y_i| \tag{7}$$

$$Bias = \frac{1}{N}\sum_{i=1}^{N}(p_i - y_i) \tag{8}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{N}(y_i - p_i)^2}{\sum_{i=1}^{N}(y_i - \bar{y})^2} \tag{9}$$

where N is the number of evaluated grid-cell observations and $\bar{y}$ is the mean observed rainfall.

NSE has the same algebraic form as the expression above when computed on the

same observations. Therefore, R^2 and NSE should not be treated as independent evidence of model superiority in this setting. All metrics must be computed consistently across models, with rainfall values converted back to their original units before reporting physical errors in millimetres per day.

3.10 Evaluation of Heavy-Rainfall Predictions

Overall regression metrics can obscure model performance during infrequent high-rainfall events. Therefore, the evaluation additionally examines predictions above thresholds derived from the training data. The verified training-derived grid-cell rainfall thresholds are 11.4689, 18.2261, and 41.4353 mm/day for the 90th, 95th, and 99th percentiles, respectively.

For event evaluation, an observed or predicted grid-cell value exceeding a threshold is classified as an event. Precision, recall, F1-score, and critical success index (CSI) can then be calculated from true positives (TP), false positives (FP), and false negatives (FN). The CSI is defined as

$$CSI = \frac{TP}{TP + FP + FN} \tag{10}$$

These percentile thresholds are empirical thresholds for the study dataset; they are not official flood-warning or public-safety categories. Grid-cell event detection should also be distinguished from daily regional-maximum event detection, because the two approaches evaluate different aspects of rainfall prediction.

4. Results and Discussion

4.1 Overall Predictive Performance

Table 4 presents the main reconstructed comparison. The TFT model showed the minimum RMSE (7.1474) and maximum R2 (0.4910). The LSTM method showed the minimum MAE (3.3736), but the GRU method yielded comparable results. The HTSAN model showed an RMSE of 7.5911, an MAE of 3.6779, and an R2 of 0.4259. These findings contradict the previous assertion that HTSAN is superior to all benchmark methods. The differing rankings of RMSE and MAE prove that model performance varies depending on the error metric used. RMSE penalizes large errors, while MAE estimates the average error. The R2 values indicate that significant variability in precipitation remains unexplained. The presented results reflect the agreement with ERA5-derived targets, not forecasts. The regression performance of the analyzed models is shown in Table 4.

Table 4. Test-set regression performance of the evaluated rainfall prediction models

Model	RMSE	MAE	Bias	R^2
TFT-style	7.1474	3.3962	+0.1314	0.4910
GRU	7.1658	3.4413	+0.1191	0.4884
ST-GNN	7.1978	3.2717	−0.3577	0.4838
LSTM	7.2065	3.3736	−0.1249	0.4826
BiLSTM	7.2263	3.5257	+0.3416	0.4797
ConvLSTM	7.2610	3.6694	+0.3332	0.4747
Informer-style	7.3032	3.4815	−0.3504	0.4686

Autoformer-style	7.3233	3.4736	+0.1436	0.4657
Transformer	7.4246	3.7745	−0.0300	0.4508
HTSAN	7.5911	3.6779	−0.0564	0.4259

4.2 Training Behaviour and Prediction Analysis

Figure 3 shows the learning curves of HTSAN and the competing networks. The figure demonstrates how the optimization process evolves and shows the connection between the training loss and the validation loss. However, the training losses of the networks may differ due to different configurations or even because of different loss scales, so comparing the losses can only be an illustrative approach rather than a direct comparison of test-set accuracies of the models. Figure 4 shows the visualization of the comparison between the observed and predicted precipitation levels. The figures demonstrate how accurate the predictions are and if there is any deviation from the expected one-to-one relationship. The reconstruction of HTSAN predictions reveals that it is difficult to reconstruct the largest precipitation values. The recorded maximum observed value was 214.21, while the maximum prediction was 72.09 in the target scale. Also, there were negative predictions among the saved normalized values, which are physically incorrect in terms of precipitation. This shows the limitations in the representation of high-rainfall values and negative outputs of the model.

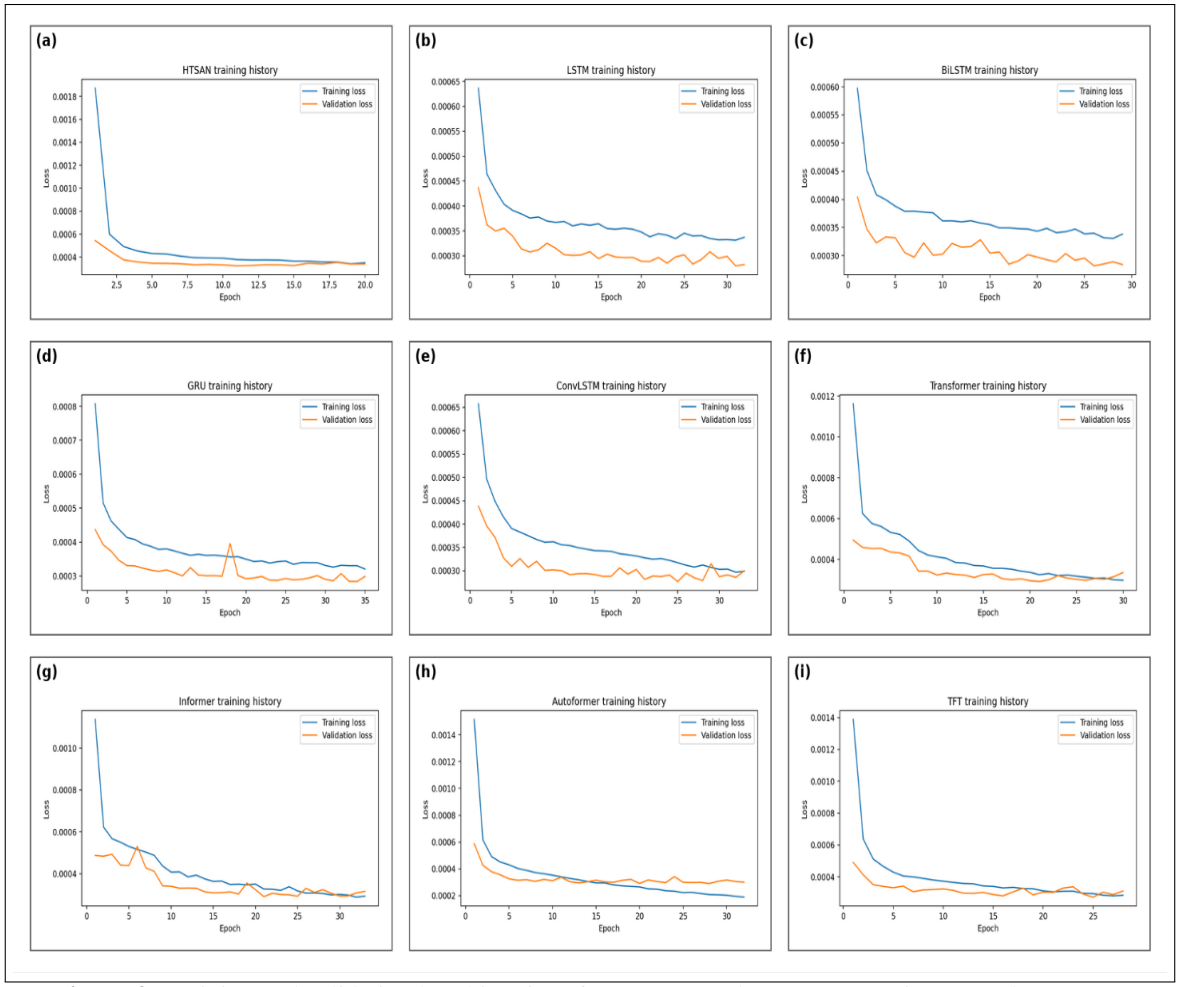


Figure 3. Training and validation loss histories of (a) HTSAN, (b) LSTM, (c) BiLSTM, (d) GRU, (e) ConvLSTM, (f) Transformer, (g) Informer, (h) Autoformer, and (i) TFT

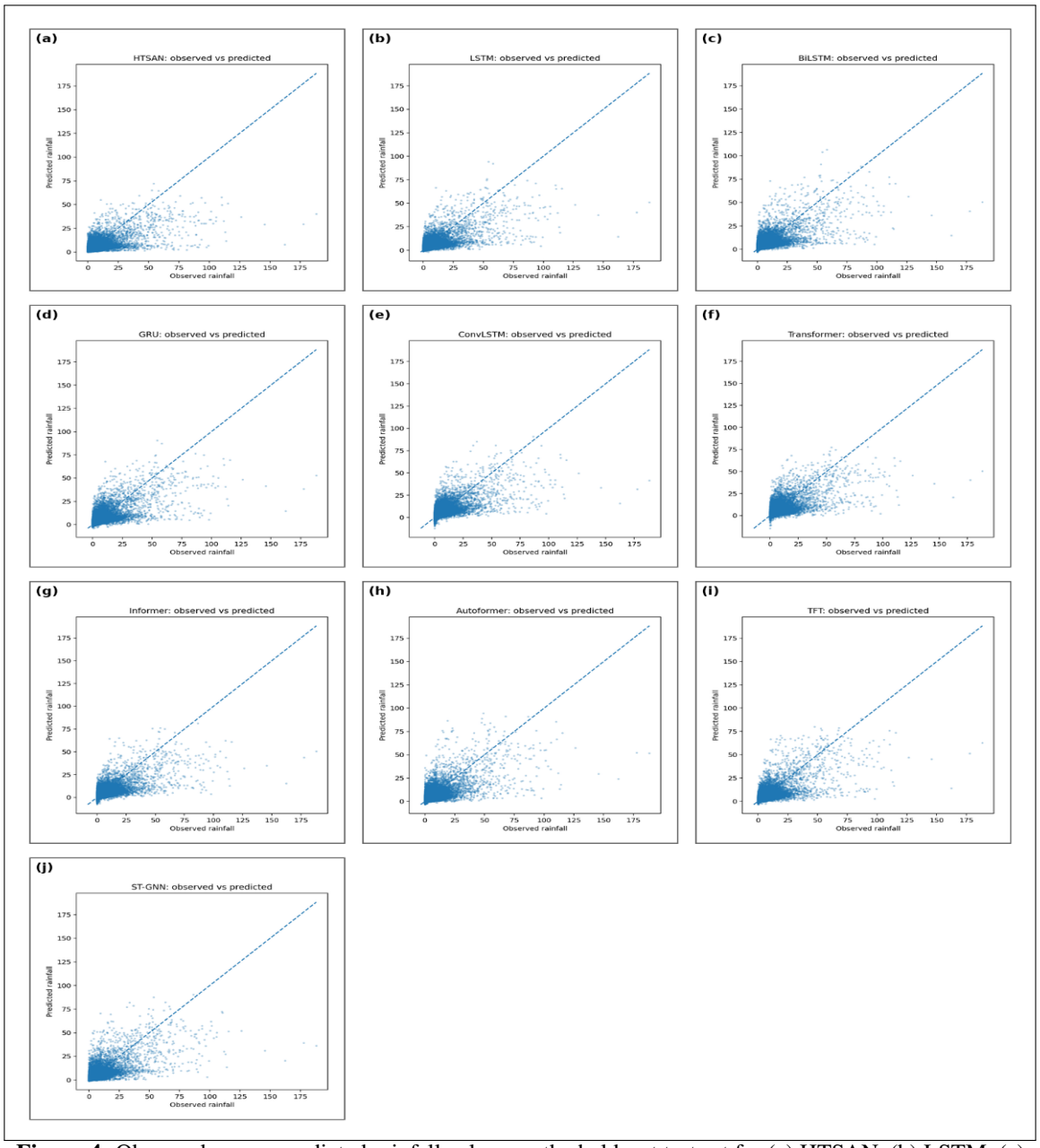


Figure 4. Observed versus predicted rainfall values on the held-out test set for (a) HTSAN, (b) LSTM, (c) BiLSTM, (d) GRU, (e) ConvLSTM, (f) Transformer, (g) Informer, (h) Autoformer, (i) TFT, and (j) ST-GNN

4.3 High-Rainfall Event Detection

The detection of high-rainfall events was determined based on the 90th, 95th, and 99th percentile thresholds, which were obtained from the training dataset. The model's performance in detecting these events was determined by applying the Critical Success Index (CSI). According to Table 5, HTSAN had CSI values of 0.2893, 0.2572, and 0.1477 at the P90, P95, and P99 thresholds, respectively. It is evident that detection performance decreases with increasing rainfall thresholds. At P90, ConvLSTM had the highest CSI of 0.3652. On the other hand, at P99, TFT recorded the highest CSI (0.2912) and was closely followed by Autoformer (0.2906). Additionally, HTSAN did not outperform the other models at any of the three thresholds.

Table 5. Grid-cell critical success index at the training-derived P90, P95, and P99 rainfall thresholds

Model	CSI (P90)	CSI (P95)	CSI (P99)
Autoformer	0.3632	0.3217	0.2906
BiLSTM	0.3417	0.3050	0.2628
ConvLSTM	0.3652	0.3097	0.2474
GRU	0.3400	0.3130	0.2784
HTSAN	0.2893	0.2572	0.1477
Informer	0.3434	0.2916	0.2014
LSTM	0.3242	0.2915	0.2325
ST-GNN	0.3475	0.2946	0.2646
TFT	0.3570	0.3193	0.2912
Transformer	0.3395	0.3073	0.2629

Note: P90, P95 and P99 are the 90th, 95th and 99th percentiles of training-period grid-cell rainfall. CSI is the critical success index.

4.4 Daily Regional-Maximum Rainfall

The daily maximum rainfall for the region was obtained by analyzing the maximum rainfall for each of the 17 × 17 matrices on each testing day. The regression analysis results for RMSE, MAE, and R2 are provided in Table 6. Transformer provided the minimum RMSE (25.6078) and maximum R2 (0.3714), while the GRU gave the minimum MAE (13.6219). The HTSAN showed an RMSE of 28.4596, an MAE of 15.8991, and an R^2 of 0.2236, demonstrating lower performance compared to other benchmark models.

Table 6. Prediction performance for daily regional-maximum precipitation. Regression performance

Model	RMSE (mm/day)	MAE (mm/day)	R^2
Autoformer	26.5470	14.4046	0.3244
BiLSTM	26.4899	14.1054	0.3273
ConvLSTM	26.3928	13.8956	0.3323
GRU	25.6597	13.6219	0.3688
HTSAN	28.4596	15.8991	0.2236
Informer	27.8503	15.1444	0.2565
LSTM	26.9019	14.3770	0.3062
ST-GNN	26.3454	14.2642	0.3346
TFT	26.5796	14.1768	0.3228
Transformer	25.6078	13.8896	0.3714

Table 7. Daily maximum rainfall event detection (CSI)

Model	CSI (P90)	CSI (P95)	CSI (P99)
Autoformer	0.2439	0.1000	0.0000
BiLSTM	0.2208	0.1111	0.0000
ConvLSTM	0.2078	0.0278	0.0000
GRU	0.2692	0.0556	0.0000
HTSAN	0.0933	0.0000	0.0000
Informer	0.1772	0.0286	0.0000
LSTM	0.2179	0.0556	0.0000
ST-GNN	0.2237	0.0571	0.0000

TFT	0.2530	0.0541	0.0000
Transformer	0.2025	0.0000	0.0000

Note: Daily maximum thresholds are derived from the training-period daily spatial maxima. All metrics are calculated from the saved test predictions.

Daily regional-maximum event detection was further evaluated using CSI at the P90, P95, and P99 thresholds, as shown in Table 7. HTSAN achieved a CSI of 0.0933 at P90 and zero at P95 and P99. At P90, GRU recorded the highest CSI (0.2692), while no model successfully detected any P99 event. These findings indicate that daily regional-maximum rainfall events, particularly those associated with extreme rainfall, remain difficult to detect. HTSAN's results do not demonstrate an advantage over the benchmark models for regional-maximum rainfall estimation or event detection.

4.5 Controlled Multi-Seed Ablation

Controlled ablation experiments used a common provisional protocol across seeds 42, 7, and 2026. Table 8 reports the mean and standard deviation of normalized-scale metrics. The variant without the Transformer had the lowest mean RMSE and MAE and the highest mean R^2 among the tested variants. The full model, therefore, did not consistently outperform the ablations under this protocol.

Table 8. Multi-seed ablation results for the full model and component-removal variants

Variant	RMSE normalized (mean ± SD)	MAE normalized (mean ± SD)	R^2 (mean ± SD)
Full model	0.030293 ± 0.000390	0.013651 ± 0.000679	0.462639 ± 0.013896
Without Transformer	0.029430 ± 0.000585	0.013443 ± 0.000168	0.492740 ± 0.020246
Without convolutional recurrent component	0.030126 ± 0.000030	0.014179 ± 0.000337	0.468587 ± 0.001054
Without both components	0.030378 ± 0.000833	0.013621 ± 0.000277	0.459413 ± 0.029765

4.6 Paired Block-Bootstrap Analysis

Paired differences between the controlled ablations and the full model were assessed using 2,000 moving-block bootstrap replicates with seven-day temporal blocks. Differences are calculated as ablation minus full-model error; negative error differences favor the ablation.

Table 9. Paired seven-day block-bootstrap differences between ablation variants and the full model

Comparison	Metric	Difference	95% bootstrap CI	Interpretation
Without Transformer – full	RMSE normalized	−0.000873	[−0.001924, 0.000127]	Includes zero
Without Transformer – full	MAE normalized	−0.000456	[−0.000815, −0.000102]	Favors ablation
Without convolutional recurrent component – full	RMSE normalized	−0.000330	[−0.000950, 0.000263]	Includes zero
Without convolutional recurrent component – full	MAE normalized	+0.000492	[0.000224, 0.000737]	Favors full model
Without both components – full	RMSE normalized	−0.000096	[−0.000895, 0.000765]	Includes zero
Without both components – full	MAE normalized	−0.000169	[−0.000531, 0.000239]	Includes zero

The paired bootstrap estimates and their 95% confidence intervals are reported in Table 9. However, for all three comparisons, RMSE confidence intervals contain zero, which means that no RMSE differences can be statistically established using the bootstrap. The MAE confidence interval is favorable to the variant without the Transformer, while the variant without the convolutional recurrent component has a greater MAE compared to the baseline. Confidence intervals depend on trained models and the testing time period and do not take into consideration training variance.

4.7 Discussion and Limitations

The last reconstructed comparison proves that HTSAN was not superior to the state-of-the-art baselines in terms of regression performance. Conclusions about high-rainfall detection are restricted by the CSI scores calculated based on the final stored predictions. The results of the ablation study additionally prove that the Transformer part did not consistently contribute to performance within the proposed training procedure. Overall, the above-mentioned facts prove that complex architectures do not ensure improved precipitation prediction. The problem with obtaining large precipitation values indicates a need for analysis of the representation of extreme events, constraints on the output, as well as the input temporal and spatial information. It should be mentioned that the target data are obtained based on ERA5 with a 0.25° grid size.

5. Conclusion

In this study, HTSAN was explored for predicting the following day's precipitation in the Bangalore area using the ERA5 meteorological dataset. This framework employs a combination of feature extraction through convolutions, representation learning based on time series, and feature interaction through Transformers to predict precipitation in a 17x17 grid based on 14 days of meteorological data. In the reconstruction comparison test, the TFT model demonstrated the minimum RMSE (7.1474) and the maximum (R2) (0.4910), while LSTM demonstrated the minimum MAE (3.3736). For HTSAN, the RMSE is 7.5911, the MAE is 3.6779, and the (R2) is 0.4259, implying that this architecture did not outperform the best benchmark models. Furthermore, HTSAN'scritical success indices were lower than those of the best benchmark models at all grid-cell rainfall threshold values under consideration. The multi-seed ablation experiments did not show the advantage of the complete architecture either; the variant without the Transformer provided better mean normalized RMSE and MAE. Future work should verify precipitation aggregation, evaluate predictions against independent rain-gauge observations, and investigate physically constrained outputs and improved extreme-rainfall modelling.

References

[1] Abbot, John, and Jennifer Marohasy. "Skilful Rainfall Forecasts from Artificial Neural Networks with Long Duration Series and Single-Month Optimization." Atmospheric Research 197 (2017): 289-299.

[2] Pham, Binh Thai, Lu Minh Le, Tien-Thinh Le, Kien-Trinh Thi Bui, Vuong Minh Le, Hai-Bang Ly, and Indra Prakash. "Development of Advanced Artificial Intelligence Models for Daily Rainfall Prediction." Atmospheric Research 237 (2020): 104845.

[3] Zhou, Zhengzheng, Jie Ren, Xiaogang He, and Shuguang Liu. "A Comparative Study of

Extensive Machine Learning Models for Predicting Long-Term Monthly Rainfall with an Ensemble of Climatic and Meteorological Predictors." Hydrological processes 35, no. 11 (2021): e14424.

[4] Kang, Jinle, Huimin Wang, Feifei Yuan, Zhiqiang Wang, Jing Huang, and Tian Qiu. "Prediction of Precipitation Based on Recurrent Neural Networks in Jingdezhen, Jiangxi Province, China." Atmosphere 11, no. 3 (2020): 246.

[5] Barrera-Animas, Ari Yair, Lukumon O. Oyedele, Muhammad Bilal, Taofeek Dolapo Akinosho, Juan Manuel Davila Delgado, and Lukman Adewale Akanbi. "Rainfall Prediction: A Comparative Analysis of Modern Machine Learning Algorithms for Time-Series Forecasting." Machine Learning with Applications 7 (2022): 100204.

[6] Rahman, Atta-ur, Sagheer Abbas, Mohammed Gollapalli, Rashad Ahmed, Shabib Aftab, Munir Ahmad, Muhammad Adnan Khan, and Amir Mosavi. "Rainfall Prediction System Using Machine Learning Fusion for Smart Cities." Sensors 22, no. 9 (2022): 3504.

[7] Wei, Ming, and Xue-yi You. "Monthly Rainfall Forecasting by a Hybrid Neural Network of Discrete Wavelet Transformation and Deep Learning." Water Resources Management 36, no. 11 (2022): 4003-4018.

[8] Ahamed, S., and M. Asif. "Flood Risk Assessment in Bangalore: Challenges and Solutions." International Journal of Creative Research Thoughts (IJCRT) 6, no. 3 (2018): 515-531.

[9] Mahadeva, M., C. C. Abhijth, V. E. Harshith, N. R. Leela, and N. S. Naik. "A Review on Urban Flooding in Bangalore: A Growing Environmental Crisis." J. Environ. Eng. Stud 10 (2025): 11-19.

[10] Kumar, Vijendra, Naresh Kedam, Kul Vaibhav Sharma, Khaled Mohamed Khedher, and Ayed Eid Alluqmani. "A Comparison of Machine Learning Models for Predicting Rainfall in Urban Metropolitan Cities." Sustainability 15, no. 18 (2023): 13724.

[11] Farman, H., Mastoi, Q., Abbas, Q., Ahmad, S., Alshahrani, A., Jan, S., & Syed, T. A. "A Comparative Study of Deep Learning and Modern Machine Learning Methods for Predicting Australia's Precipitation". International Journal of Advanced Computer Science and Applications, (2025). 16(4).

[12] Kumar, Vijendra, Naresh Kedam, Ozgur Kisi, Saleh Alsulamy, Khaled Mohamed Khedher, and Mohamed Abdelaziz Salem. "A Comparative Study of Machine Learning Models for Daily and Weekly Rainfall Forecasting: Vijendra Kumar." Water Resources Management 39, no. 1 (2025): 271-290.

[13] Hersbach, Hans, Bill Bell, Paul Berrisford, Gionata Biavati, András Horányi, Joaquín Muñoz Sabater, Julien Nicolas et al. "ERA5 Hourly Data on Single Levels from 1940 to Present." Copernicus climate change service (c3s) climate data store (cds) 10, no. 1.24381 (2023): 24381.

[14] Livieris, Ioannis E., Emmanuel Pintelas, and Panagiotis Pintelas. "A CNN–LSTM Model for Gold Price Time-Series Forecasting." Neural computing and applications 32, no. 23 (2020): 17351-17360.

[15] Liu, Wan, Yongqiang Wang, Deyu Zhong, Shuai Xie, and Jijun Xu. "ConvLSTM

Network-Based Rainfall Nowcasting Method with Combined Reflectance and Radar-Retrieved Wind Field as Inputs." Atmosphere 13, no. 3 (2022): 411.

[16] Muhtasim Ahmed Tanvir, Nazratun Naiema. "Rainfall Prediction Using Machine Learning: LSTM." Journal of Computer Science and Information Technology Vol. 2 No. 2 (2025): Journal of Computer Science and Information Technology (2025): 23-33.

[17] Yang, Yueqiao, Shichuang Li, Ting Zhou, Liang Zhao, Xiao Shi, and Boni Du. "Rainfall Forecasting Using a BiLSTM Model Optimized by an Improved Whale Migration Algorithm and Variational Mode Decomposition" Mathematics 13, no. 15: 2025. 2483.

[18] Wijaya, Surta, Tri Basuki Kurniawan, Edi Surya Negara, and Yesi Novaria Kunang. "Rainfall Prediction in Palembang City using the GRU and LSTM Methods." Journal of Data Science 2023.

[19] Upadhyay, Abhay B., Saurin R. Shah, and Rajesh A. Thakker. "Advanced Rainfall Nowcasting Using 3D Convolutional LSTM Networks on Satellite Data." Journal of Computational Mathematics and Data Science (2025): 100125.

[20] Piran, Md Jalil, Xiaoding Wang, Ho Jun Kim, and Hyun Han Kwon. "Precipitation Nowcasting Using Transformer-Based Generative Models and Transfer Learning for Improved Disaster Preparedness." International Journal of Applied Earth Observation and Geoinformation 132 (2024): 103962.

[21] Jun, Jimin, and Hong Kook Kim. "Informer-based Temperature Prediction Using Observed and Numerical Weather Prediction Data." Sensors 23, no. 16 (2023): 7047.

[22] Subramanian, Dhanalakshmi, Poongothai Marimuthu, and Nithyalakshmi Ramadoss. 2026. "A Multifaceted Framework for Predicting Ambient Air Pollution in the National Capital Region". Journal of Trends in Computer Science and Smart Technology 8 (3): 558-78. https://doi.org/10.36548/jtcsst.2026.3.007.

[23] Copernicus Climate Change Service Climate Data Store. "Copernicus Climate Change Service (C3 S)(2017): ERA5: Fifth generation of ECMWF atmospheric reanalyses of the global climate." (2017). https://cds.climate.copernicus.eu