Vol. 8, No. 4, August 2026

ISI Journal of Information Systems and Informatics

Published By Asosiasi Doktor Sistem Informasi Indonesia

A Hybrid ACO-BPNN-XGBoost Model for Monthly Rainfall Time-Series Forecasting

Syaharuddin[1*], Safaruddin[2], Vera Mandailina[3], Abdillah[4], Mahsup[5]

[1,2,3,4,5]Mathematics Department, Faculty of Education and Educational Sciences, University of Muhammadiyah Mataram, Mataram, Indonesia

Received:
February 22, 2026
Revised:
June 3, 2026
Accepted:
August 4, 2026
Published:
August 30, 2026

Corresponding Author:
Author Name*:
Syaharuddin
Email*:
syaharuddin.ntb@gmail.com

DOI:
10.63158/journalisi.v8i4.1693

Abstract. This study aimed to develop and evaluate a hybrid forecasting model integrating Ant Colony Optimization (ACO), Backpropagation Neural Network (BPNN), and XGBoost for monthly rainfall prediction. The proposed hybrid framework combines optimization, neural network, and boosting techniques within a single forecasting model. Monthly rainfall time-series data from 2016 to 2025 in Alas Subdistrict, Sumbawa Regency, West Nusa Tenggara, were used, comprising 120 observations obtained from BPS, BMKG, and NASA POWER. The methodology included data preprocessing, an 80%–20% chronological training–testing split, model development, and performance evaluation using MSE, MAE, RMSE, MAPE, and R^2. The results indicated that all models experienced performance degradation during testing, suggesting overfitting and limited generalization capability. The testing RMSE values for ACO, BPNN, XGBoost, and the hybrid model were 108.41, 109.33, 109.21, and 106.08 mm, respectively. The corresponding testing MAPE values were 2836.5%, 2900.5%, 2073.9%, and 2602.2%, although these values should be interpreted cautiously because rainfall observations occasionally approached zero. While the hybrid model achieved the lowest testing RMSE, the improvement over the best individual model was modest, and all testing R^2 values remained negative, indicating weak generalization capability. Therefore, the findings should be regarded as preliminary, and further validation using larger datasets, exogenous climatic predictors, baseline forecasting methods, and more rigorous evaluation procedures is required.

Keywords: Rainfall Forecasting, Hybrid Model, Ant Colony Optimization, Backpropagation Neural Network, XGBoost

ISI Journal of Information Systems and Informatics
Published By Asosiasi Doktor Sistem Informasi Indonesia

1. INTRODUCTION

Rainfall forecasting is a crucial component of climatological studies and environmental resource management. Changes in rainfall intensity and distribution have a significant impact on the agricultural sector, food security, clean water availability, and sustainable water management. Furthermore, predictive information on rainfall holds strategic value in efforts to mitigate hydrometeorological disasters, such as floods, droughts, and landslides, whose frequency has increased in recent decades [1]. Recent research findings indicate that changes in rainfall patterns due to global climate dynamics also increase uncertainty regarding water availability and agricultural productivity [2]. Furthermore, adaptation strategies that utilize weather forecasts are becoming increasingly important for safeguarding water resources and supporting more effective disaster mitigation planning [3]. Consequently, the development of more accurate and responsive rainfall forecasting methods has become an urgent need for various strategic sectors and data-driven policy-making processes. However, monthly rainfall forecasting using only 120 historical observations remains challenging because machine learning models generally require sufficient data to learn stable temporal patterns. Such a limited dataset may increase the risk of overfitting and reduce model generalization, thereby motivating the development of forecasting approaches that can effectively utilize limited historical information.

Rainfall data exhibit complex characteristics because they are nonlinear, spatio-temporal, and often non-stationary, making it difficult to model them using traditional statistical methods [4]. Climatological research indicates that extreme rainfall variability is strongly influenced by climate dynamics, which cannot be explained solely by changes in average rainfall values [5]. Furthermore, recent time-series-based climate trend analyses reveal fluctuating seasonal patterns and long-term trends, causing conventional predictive models to frequently exhibit bias due to inappropriate stationarity assumptions. In Indonesia, monthly rainfall is strongly influenced by monsoonal seasonality, resulting in recurring wet and dry seasons that typically follow an annual (12-month) cycle. Therefore, rainfall forecasting models should capture both short-term temporal dependence and recurring seasonal patterns. Consequently, a modeling approach capable of capturing these dynamics is required to improve the accuracy and reliability of rainfall predictions.

ISI Journal of Information Systems and Informatics
Published By Asosiasi Doktor Sistem Informasi Indonesia

Over the past decade, machine learning has been increasingly utilized to address rainfall prediction challenges due to its ability to process non-linear and high-dimensional data [6]. Unlike multivariate forecasting studies, this research adopts a univariate time-series approach using only historical rainfall observations as model inputs. Deep learning models have been applied in radar-based precipitation nowcasting, where the U-Net architecture demonstrates superior performance compared to conventional approaches [7]. Additionally, a hybrid deep learning approach combining CNN and XGBoost has proven capable of generating high-precision rainfall predictions, as evidenced by low daily RMSE values in both satellite and field observation data [8]. Meanwhile, metaheuristic algorithms such as ant colony optimization have been used to optimize neural network weights in various environmental applications, indicating the great potential for integration between optimization techniques and machine learning [9].

The Ant Colony Optimization algorithm offers a highly flexible heuristic mechanism for tuning parameters in predictive models and has been used to improve the performance of time-series models as well as hyperparameter optimization processes [10]. Meanwhile, backpropagation-based artificial neural networks (such as BPNN) remain one of the primary methods for modeling complex nonlinear patterns in rainfall data due to their ability to adjust weights and biases through iterative training [11] . However, BPNNs have weaknesses related to the potential to get stuck in local minima as well as dependence on network structure and parameter initialization, thus requiring a more meticulous tuning process [12]. On the other hand, XGBoost, as a gradient boosting algorithm, has demonstrated promising performance in climate forecasting and predicting extreme rainfall events, primarily due to its ability to apply regularization and improve accuracy through an ensemble of decision trees [13]. The integration of ACO, BPNN, and XGBoost into a hybrid model is expected to overcome the limitations of each approach by combining adaptive optimization, non-linear representation capabilities, and boosting stability, thereby providing a complementary forecasting framework that is expected to improve prediction performance by combining the strengths of the three algorithms.

A hybrid approach that combines optimization algorithms with neural networks has proven effective in improving the accuracy of rainfall predictions. For example, a study in Malaysia by [14] used an MLP model optimized with various metaheuristic algorithms such as HGSO, Bat, and PSO, achieving an MAE as low as 0.712 mm and an NSE of up to

0.90. At one test station, the results were significantly better than those of non-hybrid models [15]. Additionally, another study that combined wavelet analysis with macroclimate indicators such as ENSO and sunspot activity in a machine learning model achieved a prediction accuracy of 85.7% using a Bayesian Network [16]. These findings indicate that sunspots are the most influential factor in modulating rainfall in their models. Overall, previous studies suggest that hybrid approaches may improve rainfall prediction performance under certain conditions, although their effectiveness depends on data characteristics and model design [17].

Various previous studies have shown that artificial intelligence methods and hybrid models are effective at modeling nonlinear and fluctuating rainfall data. A study by [18] applied a hybrid neural network combined with Evolutionary Programming and achieved better performance than conventional methods in predicting rainfall time series. Furthermore, [19] developed a Hybrid Wavelet Neural Network (HWNN) that integrates Ant Colony Optimization (ACO) and Particle Swarm Optimization (PSO) for optimizing the network's weights and biases, and then tested it using monthly rainfall data from Ningde City. The results of the study show that the HWNN model produces a lower RMSE value compared to both the Wavelet Neural Network (WNN) and the conventional Artificial Neural Network (ANN). Meanwhile, [20] utilized the XGBoost algorithm to predict monthly rainfall patterns and obtained relatively low MAE values on both training and test data, demonstrating the effectiveness of boosting methods in capturing nonlinear relationships in climate data. In general, these studies used monthly rainfall data with an observation span of approximately 10–15 years and evaluated model performance using metrics such as RMSE, MAE, MAPE, and the coefficient of determination (R^2), thereby demonstrating that artificial intelligence-based approaches have good potential for improving the accuracy of rainfall predictions. Nevertheless, most previous studies focused primarily on predictive accuracy and rarely discussed model generalization under limited observations or compared weighted hybrid models with their individual counterparts.

Nevertheless, most previous studies have focused on the use of single models or hybrid models that combine only two algorithmic approaches; consequently, research on the integration of metaheuristic optimization methods, artificial neural networks, and boosting algorithms within a single predictive framework remains relatively limited.

Moreover, few studies have investigated weighted hybrid strategies that explicitly compare the predictive contribution of individual models and weighted hybrid models for univariate monthly rainfall forecasting in tropical regions using relatively small datasets. Furthermore, previous studies generally emphasized improving prediction accuracy without examining the relative contributions of each model within the constructed hybrid system, and few have been applied to tropical regions where rainfall characteristics are heavily influenced by seasonal variability, such as Alas Subdistrict, Sumbawa Regency. Alas Subdistrict was selected as the study site because it is one of the agricultural areas in Sumbawa Regency that is highly dependent on seasonal water availability to support agricultural activities. Furthermore, this area exhibits significant year-to-year rainfall variability, which has the potential to create uncertainty in crop planning, water resource management, and drought risk mitigation. Therefore, a better understanding of rainfall dynamics in Alas Subdistrict is essential to support more effective decision-making in the agricultural sector and environmental management. Consequently, this study proposes a weighted hybrid ACO-BPNN-XGBoost framework integrating parameter optimization, nonlinear learning, and ensemble learning through performance-based weighting. Historical rainfall observations were represented using lag-1, lag-2, and lag-3 predictors to capture short-term temporal dependence while maintaining a parsimonious forecasting framework appropriate for the relatively small dataset. The rationale for excluding the seasonal lag (t−12) is discussed in the Methodology section. Thus, this study aims to compare the predictive performance of the individual ACO, BPNN, and XGBoost models with the proposed weighted hybrid ACO-BPNN-XGBoost model for monthly rainfall forecasting and to evaluate whether the weighted hybrid framework provides complementary predictive performance under limited historical observations.

2. METHODS

This study employed a quantitative experimental forecasting approach based on monthly rainfall time-series data [21]. The research aimed to evaluate and compare the predictive performance of three individual models, namely Ant Colony Optimization (ACO), Backpropagation Neural Network (BPNN), and Extreme Gradient Boosting (XGBoost), as well as a weighted hybrid ACO-BPNN-XGBoost model. The dataset consisted of 120 monthly observations covering the period from January 2016 to December 2025. Rainfall data were obtained from BMKG and BPS, whereas NASA POWER data were collected only as supporting climatic information for descriptive purposes and were not incorporated as predictor variables in the forecasting models. The forecasting framework consisted of data collection, preprocessing, model development, training and testing procedures, hybrid model construction, and performance evaluation using several statistical metrics. Because the data are time-series observations, a chronological split was applied to preserve the temporal structure of the dataset. The first 96 observations (80%) were used for training, while the remaining 24 observations (20%) were reserved for testing. The methodological procedures employed in this study are described in detail in the following subsections.

A chronological hold-out validation strategy (80% training and 20% testing) was adopted because of the relatively limited number of monthly observations available in this study. This validation strategy preserves the temporal ordering of the observations and prevents future information from being used during model training. Although more rigorous validation procedures, such as rolling-origin evaluation or walk-forward validation, are generally recommended for time-series forecasting, their implementation was not considered appropriate in the present study because the dataset comprised only 120 monthly observations. Therefore, the adopted hold-out strategy was considered a practical compromise for evaluating model performance while maintaining sufficient training data. Future studies should investigate rolling-origin or walk-forward validation using larger datasets to obtain more robust estimates of model generalization performance.

This study employed a univariate time-series forecasting framework in which monthly rainfall served as the only variable used for model development. No exogenous climatic variables, including temperature, humidity, wind speed, solar radiation, ENSO indices, or the Standardized Precipitation Index (SPI), were incorporated as predictor variables during model training. Consequently, all forecasting models were developed and evaluated exclusively using historical monthly rainfall observations under the same univariate forecasting framework to ensure a fair comparison among the ACO, BPNN, XGBoost, and hybrid ACO-BPNN-XGBoost models.

ISI Journal of Information Systems and Informatics
Published By Asosiasi Doktor Sistem Informasi Indonesia

2.1. Research Procedure

The overall research procedure is presented at the beginning of the Methods section to provide a clear overview of the sequential stages undertaken in developing and evaluating the proposed forecasting framework. As illustrated in Figure 1, the research procedure consists of several interconnected stages, beginning with the design and development of the MATLAB GUI-based forecasting system, followed by data collection, data preprocessing, chronological data splitting, individual model development, hybrid model construction, prediction, performance evaluation, model comparison, and interpretation of the results. Each stage is described below to clarify how the proposed forecasting framework was implemented.

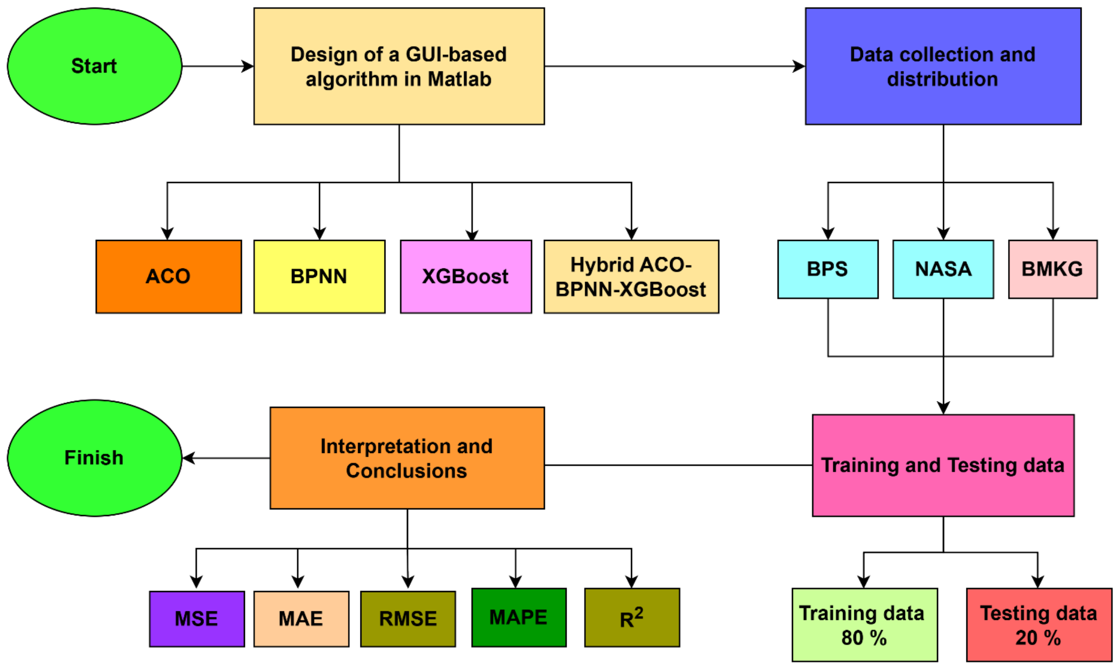

Figure 1. Research Procedure

The first stage involved the design and development of a MATLAB GUI-based forecasting system as the computational platform for implementing the proposed models. The GUI was designed to allow the ACO, BPNN, XGBoost, and hybrid ACO-BPNN-XGBoost models to be executed interactively within a single forecasting framework. This structure enabled the prediction results generated by the different models to be obtained under a consistent experimental setting.

ISI Journal of Information Systems and Informatics
Published By Asosiasi Doktor Sistem Informasi Indonesia

The second stage involved data collection. Monthly rainfall observations covering January 2016 to December 2025 were collected from the Central Bureau of Statistics (BPS) and the Meteorological, Climatological, and Geophysical Agency (BMKG), resulting in 120 monthly observations. NASA POWER data were also collected as supporting climatic information for descriptive purposes; however, these data were not incorporated as predictor variables in the forecasting models. Therefore, the forecasting framework remained univariate and relied exclusively on historical rainfall observations.

The third stage consisted of data preprocessing. The collected rainfall observations were organized chronologically and examined for completeness and consistency. No missing observations were identified, and therefore no imputation procedure was required. No explicit outlier removal was conducted because extreme rainfall values represent genuine climatological events that should be retained in the forecasting dataset. Subsequently, Min-Max normalization was applied to transform rainfall values into the interval [0,1]. The normalization coefficients were estimated using the training data and then applied to the testing data to prevent information leakage.

The fourth stage involved dividing the dataset into training and testing subsets while preserving the chronological order of the observations. The first 96 observations (80%) were used for model training, whereas the remaining 24 observations (20%) were reserved for testing. This chronological hold-out strategy was adopted because the dataset contained a relatively limited number of monthly observations and because preserving temporal ordering prevents future observations from being used during model training.

The fifth stage involved the development of three individual forecasting models: ACO, BPNN, and XGBoost. ACO was employed as an optimization-based forecasting component to search for parameter configurations that minimize forecasting error. BPNN was developed to learn nonlinear relationships between lagged rainfall observations and the target rainfall value through its input, hidden, and output layers. XGBoost was implemented as a gradient-boosting ensemble model in which decision trees were sequentially constructed to reduce prediction errors. Each model generated an independent rainfall prediction based on the same training and testing framework.

ISI Journal of Information Systems and Informatics
Published By Asosiasi Doktor Sistem Informasi Indonesia

The sixth stage involved the construction of the weighted hybrid ACO-BPNN-XGBoost model. The prediction outputs generated by the three individual models were integrated using weighting coefficients determined from the prediction errors obtained during the model training phase. The use of training errors for determining the hybrid weights was adopted because the dataset contained only 120 observations and an additional validation subset would substantially reduce the number of effective training samples. The resulting weighted combination produced the final hybrid rainfall prediction. The seventh stage consisted of denormalizing the predicted rainfall values to return the model outputs to the original rainfall scale. The denormalized predictions were then compared with the corresponding observed rainfall values in the testing dataset.

The eighth stage involved evaluating the forecasting performance of all four models using five statistical metrics: Mean Squared Error (MSE), Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), Mean Absolute Percentage Error (MAPE), and the coefficient of determination (R^2). These metrics were used to assess prediction error, the magnitude of deviations between observed and predicted values, and the goodness of fit of each forecasting model.

Finally, the performance results of the ACO, BPNN, XGBoost, and hybrid ACO-BPNN-XGBoost models were compared to determine the relative forecasting performance of the proposed approaches. The model comparison was based on the obtained evaluation metrics, with lower error values indicating better forecasting performance, while R^2 was used to assess the proportion of variation in the observed rainfall data explained by each model. The final stage involved interpreting the comparative results and drawing conclusions regarding the effectiveness of the proposed hybrid framework.

2.2. Data Preprocessing

Prior to model development, the monthly rainfall dataset underwent several preprocessing procedures to ensure data quality and consistency. First, all rainfall observations obtained from BMKG and BPS were organized chronologically according to the monthly observation period from January 2016 to December 2025. The dataset was then examined for completeness and consistency. No missing observations were identified; therefore, no data imputation procedure was required. Likewise, no explicit outlier removal was performed because extreme rainfall events represent genuine climatological phenomena that should be preserved in time-series forecasting studies. Subsequently, Min-Max normalization was applied to transform the rainfall values into the interval [0,1], thereby improving numerical stability during model training. The normalization coefficients estimated from the training dataset were consistently applied to the testing dataset to prevent information leakage.

2.3. Input Features and Target Variable

The forecasting framework developed in this study employed a univariate time-series approach using historical monthly rainfall observations. The target variable was the monthly rainfall amount at time (t), while the predictor variables consisted of rainfall values from several previous periods. Within the proposed forecasting framework, ACO was employed as an optimization algorithm to search for parameter configurations that minimize the forecasting error during model development. The optimized ACO model was subsequently evaluated individually and its prediction output was further integrated with the BPNN and XGBoost models through a weighted hybrid strategy. Therefore, ACO functioned as an optimization-based forecasting component rather than merely as a standalone optimization algorithm.

Table 1. Input Features and Target Variable

Variable	Description
Rainfall ((t-1))	Monthly rainfall in the previous period
Rainfall ((t-2))	Monthly rainfall two periods earlier
Rainfall ((t-3))	Monthly rainfall three periods earlier
Rainfall ((t))	Target variable to be predicted

Because only historical rainfall observations were utilized, the proposed ACO, BPNN, XGBoost, and hybrid ACO-BPNN-XGBoost models can be categorized as univariate time-series forecasting models. The use of lagged rainfall variables enables the models to learn temporal relationships among consecutive observations and identify recurring rainfall patterns contained in the historical data.

In addition, the Standardized Precipitation Index (SPI) was calculated to provide a descriptive assessment of rainfall anomalies and drought conditions in the study area. However, SPI was used solely as a descriptive indicator and was not incorporated as an input variable or prediction target in the forecasting models. Therefore, the prediction process focused exclusively on estimating monthly rainfall amounts based on historical rainfall records. All predictor variables and the target variable were arranged according to the same monthly observation period spanning from January 2016 to December 2025, thereby ensuring temporal consistency throughout the modeling process.

2.4. ACO (Ant Colony Optimization)

Ant Colony Optimization (ACO) is a metaheuristic algorithm that mimics the behavior of ant colonies in finding the shortest path between the nest and a food source. Virtual ants move through the solution space, leaving a pheromone trail that increases the likelihood that the next ant will choose the optimal path. ACO is highly effective for combinatorial optimization problems, feature selection, and model parameter tuning, particularly in complex and nonlinear systems.

Basic ACO formula:

1) Path selection probability

$$P_{ij}^{k} = \frac{[\tau_{ij}]^{\alpha} \times [\eta_{ij}]^{\beta}}{\sum_{l \in allowed_k} [\tau_{ij}]^{\alpha} \times [\eta_{ij}]^{\beta}} \quad (1)$$

The probability that an ant k moves from i to j is denoted as (P_{ij}^{k}) and is determined by two main factors: the pheromone intensity along the path i→j, denoted as τ_{ij}, and the heuristic value of that path, η_{ij}. These two factors are influenced by the parameters α and β, which respectively govern the extent to which pheromones and heuristic information contribute to the path selection process. Ant movement is only considered on nodes that have not been visited previously, referred to as {allowed k} or the set of nodes still available to the k-th ant. Thus, the ACO mechanism combines pheromone trails and heuristic information to adaptively determine the direction of the solution search.

2) Pheromone update

$$\tau_{ij} \leftarrow (1-\rho)\tau_{ij} + \sum_{k=1}^{m} \Delta\tau_{ij}^{k} \quad (2)$$

The parameter ρ indicates the rate of pheromone evaporation, with values ranging from 0 to 1, which determines how quickly the pheromone on a route fades. Meanwhile, the symbol $\Delta\tau_{ij}^{k}$ represents the amount of pheromone released by the kth ant along the path from node i to node j, serving as a reinforcement of the path already traversed. The total number of ants involved during the optimization process is denoted by mmm, and each

ISI Journal of Information Systems and Informatics
Published By Asosiasi Doktor Sistem Informasi Indonesia

ant contributes to forming the solution while simultaneously updating the pheromone along its travel path.

In the proposed forecasting framework, ACO was employed to optimize the search process for model parameter configurations that minimize the prediction error. Unlike conventional optimization techniques, ACO iteratively explores the solution space using pheromone updating and probabilistic path selection to identify more suitable parameter combinations. The optimized ACO model was evaluated individually and subsequently integrated with the BPNN and XGBoost models within the weighted hybrid forecasting framework.

2.5. BPNN (Backpropagation Neural Network)

The Backpropagation Neural Network (BPNN) is one of the most popular types of artificial neural networks (ANNs) for predictive modeling. A BPNN works by learning the nonlinear relationships between inputs and outputs through layers of neurons (input, hidden, and output) and activation functions. The learning process uses the backpropagation algorithm, which calculates the error in the output and propagates it backward (backpropagates) to update the weights between neurons using the gradient descent method. BPNNs are highly effective for time series modeling, weather forecasting, pattern recognition, and various classification or regression applications.

BPNN algorithm formula:

1) Output neuron

$$y_j = f\left(\sum_{i=1}^{n} w_{ij}\, x_i + b_j\right) \tag{3}$$

The j-th neuron produces an output y_j, which is the result of applying an activation function f—such as sigmoid, ReLU, or tanh—to the combination of input signals. This output calculation relies on the connection weights w_{ij} linking neuron i to neuron j, as well as the contribution of the bias b_j to enhance the network's learning capability. The input signal to neuron j, denoted as x_i, serves as the primary component before the activation function is applied to determine the neuron's output.

2) Error in the output

$$E = \frac{1}{2}\sum_{J}\left(t_j - y_j\right)^2 \tag{4}$$

The symbol t_j is used to denote the expected output of neuron j, while y_j is the actual output produced by the neuron after processing. The difference between the target and the actual output measures the error, which is then used in the training process to adjust the weights and biases so that the neural network's performance can be further optimized.

3) Weight update (Gradient Descent)

$$w_{ij}(t+1) = w_{ij}(t) + \eta\delta_j x_i \tag{5}$$

$$\eta\delta_j = (t_j - y_j)f'^{(y_j)} \tag{6}$$

The parameter η serves as the learning rate, which determines the magnitude of the weight adjustment step in each training iteration. Meanwhile, δ_j represents the error value at neuron j, which is then used to calculate the precise weight correction. The component $f'(y_j)$ is the derivative of the activation function evaluated at the output of neuron y_j, and plays a crucial role in the backpropagation process by measuring the neuron's sensitivity to changes in error.

2.6. XGBoost (Extreme Gradient Boosting)

XGBoost is a gradient-boosting-based ensemble learning algorithm designed to improve prediction accuracy and computational efficiency. This algorithm builds a series of decision trees incrementally, with each new tree attempting to reduce the error (residual) of the previous tree. The XGBoost formula is described in Equations (7) and (8).

$$y^\wedge{}_i = \sum_{k=1}^{K} f_k\,(x_i), \qquad f_k \in F \tag{7}$$

$$L(\emptyset) = \sum_i l(y_i, y^\wedge{}_i) + \sum_k \Omega(f_k) \tag{8}$$

The prediction for the i-th data point is denoted as $y^\wedge{}_i$, which is obtained from a combination of several decision trees, each denoted as f_k. All possible decision tree models form the hypothesis space F, while K denotes the total number of trees used in the modeling process. Prediction quality is evaluated using the loss function $\sum_i l(y_i, y^\wedge{}_i) + \sum_k \Omega(f_k)$, such as Mean Squared Error (MSE) in the case of regression, to measure the difference between the actual and predicted values. Additionally, each tree f_k is subject to regularization $\Omega(f_k)$= $\gamma T + \frac{1}{2}\lambda \sum_J w_j^2$, which serves to control the model's complexity; where T is the number of leaves on the tree, while w_j represents the weight

ISI Journal of Information Systems and Informatics
Published By Asosiasi Doktor Sistem Informasi Indonesia

for each leaf. This mechanism ensures that the model is not only accurate but also remains simple and avoids overfitting.

2.7. Model Configuration and Hyperparameter Settings

The performance of machine learning algorithms is highly influenced by the selection of appropriate hyperparameters. Therefore, each model employed in this study was configured using a predefined set of hyperparameters to achieve stable learning performance while avoiding excessive model complexity. The selected values were determined based on preliminary experiments and commonly adopted settings reported in previous forecasting studies. A summary of the hyperparameter configurations for the ACO, BPNN, and XGBoost models is presented in Table 2.

Table 2. Hyperparameter Configuration

Method	Parameter	Value
ACO	Number of Ants	20
	Maximum Iterations	100
	Alpha (α)	1
	Beta (β)	2
	Evaporation Rate (ρ)	0.5
	Initial Pheromone	1
	Objective Function	Minimum MSE
BPNN	Hidden Neurons	10
	Learning Rate	0.01
	Epochs	1000
	Training Function	trainlm
	Hidden Activation Function	tansig
	Output Activation Function	purelin
	Performance Function	MSE
XGBoost	n_estimators	50
	max_depth	3
	learning_rate	0.1
	Objective Function	reg:squarederror
	Booster	gbtree
	Evaluation Metric	RMSE

The hyperparameters were selected based on preliminary experiments and recommendations from previous studies on rainfall forecasting and machine learning optimization. For the ACO model, 20 ants and 100 iterations were employed to balance exploration and computational efficiency, while $\alpha = 1$ and $\beta = 2$ were used to regulate the influence of pheromone trails and heuristic information. The BPNN model utilized a single hidden layer consisting of 10 neurons with a tanh activation function and was trained using the Levenberg–Marquardt algorithm for 1000 epochs. For the XGBoost model, 50 estimators, a maximum tree depth of 3, and a learning rate of 0.1 were adopted to reduce overfitting while maintaining predictive performance. Before model training, the rainfall dataset underwent preprocessing consisting of data organization, consistency checking, and Min–Max normalization. The normalization parameters were estimated using only the training dataset and subsequently applied to the testing dataset using the same transformation coefficients to prevent information leakage. Missing observations were not found in the collected monthly rainfall records; therefore, no data imputation procedure was required. Likewise, no explicit outlier removal was performed because extreme rainfall events represent natural climatological variability that should be preserved for forecasting purposes. All forecasting models were implemented using MATLAB R2018a (MathWorks Inc., Natick, MA, USA). The Neural Network Toolbox was employed for the BPNN implementation, whereas the Statistics and Machine Learning Toolbox was used for the XGBoost-related procedures. The experiments were conducted on a computer equipped with an Intel Core processor, 16 GB RAM, running Windows 11. A fixed random seed was employed during model initialization to improve the reproducibility of the experimental results.

The weighting coefficients of the hybrid ACO-BPNN-XGBoost model were determined using the prediction errors obtained during the model training phase. This strategy was adopted because the relatively limited dataset (120 monthly observations) did not allow an additional validation subset without substantially reducing the effective training samples. Although this approach enables the integration of the three forecasting models, it may introduce optimistic weight estimation. Therefore, future studies are encouraged to estimate hybrid weights using validation errors, out-of-fold prediction errors, or rolling-origin validation to improve the reliability and generalization capability of the hybrid model.

Journal of Information Systems and Informatics
Published By Asosiasi Doktor Sistem Informasi Indonesia

To improve the reproducibility of the proposed forecasting framework, the overall workflow of the hybrid ACO-BPNN-XGBoost model is summarized in Algorithm 1. The algorithm describes the sequence of data preprocessing, model training, hybrid prediction, and performance evaluation.

Algorithm 1. Hybrid ACO-BPNN-XGBoost Forecasting Framework

```
Input:
    Monthly rainfall dataset
Output:
    Monthly rainfall prediction
Begin
Load rainfall dataset
Preprocess data
Normalize data
Split dataset
      Training = 80%
      Testing = 20%
Train ACO model
Generate prediction ACO
Train BPNN model
Generate prediction BPNN
Train XGBoost model
Generate prediction XGBoost
Calculate hybrid weights
Generate Hybrid Prediction
Denormalize prediction
Calculate
MSE
MAE
RMSE
MAPE
R²
Compare model performance
End
```

2.8. Performance Evaluation Metrics

The forecasting performance of the ACO, BPNN, XGBoost, and hybrid ACO-BPNN-XGBoost models was evaluated using five statistical indicators, namely Mean Squared Error (MSE), Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), Mean Absolute Percentage Error (MAPE), and the coefficient of determination (R^2). These metrics were selected because they provide complementary information regarding prediction accuracy, error magnitude, and model goodness-of-fit. The evaluation metrics used to assess the performance of the forecasting models are mathematically defined as shown in Equation 9.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{9}$$

ISI Journal of Information Systems and Informatics
Published By Asosiasi Doktor Sistem Informasi Indonesia

Where y_i denotes the actual rainfall value, $\hat{y}_i$ denotes the predicted rainfall value, and nnn represents the total number of observations. A lower MSE value indicates better forecasting accuracy because it reflects a smaller average squared prediction error.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{10}$$

Where y_i is the actual value, $\hat{y}_i$ is the predicted value, and nnn is the number of observations. Mean Absolute Error (MAE) quantifies the average magnitude of prediction errors by calculating the absolute difference between observed and predicted values. Unlike MSE, MAE does not disproportionately penalize large errors, making it easier to interpret as the average deviation of predictions from actual observations. Therefore, smaller MAE values indicate more accurate forecasting performance.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{11}$$

Where y_i denotes the observed value, $\hat{y}_i$ denotes the predicted value, and nnn is the total number of observations. Root Mean Squared Error (RMSE) is derived from the square root of MSE and provides an error measure in the same unit as the original rainfall data, making it easier to interpret in practical applications. RMSE is widely used in forecasting studies because it emphasizes larger prediction errors while still reflecting the overall predictive capability of the model. Lower RMSE values indicate better agreement between observed and predicted rainfall values.

$$MAPE = \frac{100}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \tag{12}$$

where y_i is the actual value, $\hat{y}_i$ is the predicted value, and nnn is the number of observations. Mean Absolute Percentage Error (MAPE) expresses forecasting errors as percentages of the observed values, allowing prediction performance to be evaluated independently of measurement units. This metric facilitates comparison across different datasets and forecasting models because the results are represented as percentages.

However, because rainfall values occasionally approach zero, MAPE should be interpreted cautiously because percentage errors may become unstable under such conditions.

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{13}$$

where y_i is the actual value, $\hat{y}_i$ is the predicted value, $\bar{y}$ is the mean of the observed values, and n represents the total number of observations. The coefficient of determination (R^2) measures the proportion of variation in the observed data that can be explained by the forecasting model. Values approaching 1 indicate that the model successfully captures most of the variability in the rainfall data, whereas values close to 0 indicate limited explanatory capability. Furthermore, negative R^2values indicate that the forecasting model performs worse than a simple predictor based on the mean of the observed data, suggesting poor generalization performance.

Subsequently, the actual rainfall data used as the basis for the forecasting process are presented. These data play an important role in this study because they represent the real conditions of rainfall variability in the study area and serve as the primary reference for evaluating the predictive performance of the proposed models. The dataset consists of monthly rainfall observations covering the period from 2016 to 2025, obtained from the Central Bureau of Statistics (BPS) and the Meteorological, Climatological, and Geophysical Agency (BMKG). An initial visualization of the dataset is provided to illustrate the temporal characteristics and fluctuation patterns of rainfall throughout the observation period. Understanding these patterns is essential for developing and assessing the ACO, BPNN, XGBoost, and hybrid ACO-BPNN-XGBoost models. The visualization of the actual rainfall data is presented in Figure 2.

Figure 2 illustrates the dynamics of the rainfall time series for the observation period from 2016 to 2025, showing fairly significant fluctuations over time. At the beginning of the period (2016–2017), rainfall values tended to be relatively high, with several extreme peaks reaching over 300 mm, though these were interspersed with decreases at certain points in time. Subsequently, from 2018 through approximately 2021, a fairly consistent decline was observed, with rainfall values dominated by low levels approaching zero, indicating a dry phase or a prolonged decrease in rainfall intensity. Entering the period from 2022 to 2025, the pattern again shows a significant increase with high variability, including the emergence of several rainfall spikes that once again reach extreme values approaching 400 mm. Overall, this data pattern indicates non-stationarity in the time series, with a regime shift from relatively wet conditions at the beginning, to dry conditions in the middle period, and a subsequent increase in the final period, reflecting complex climate variations or atmospheric dynamics in the study area.

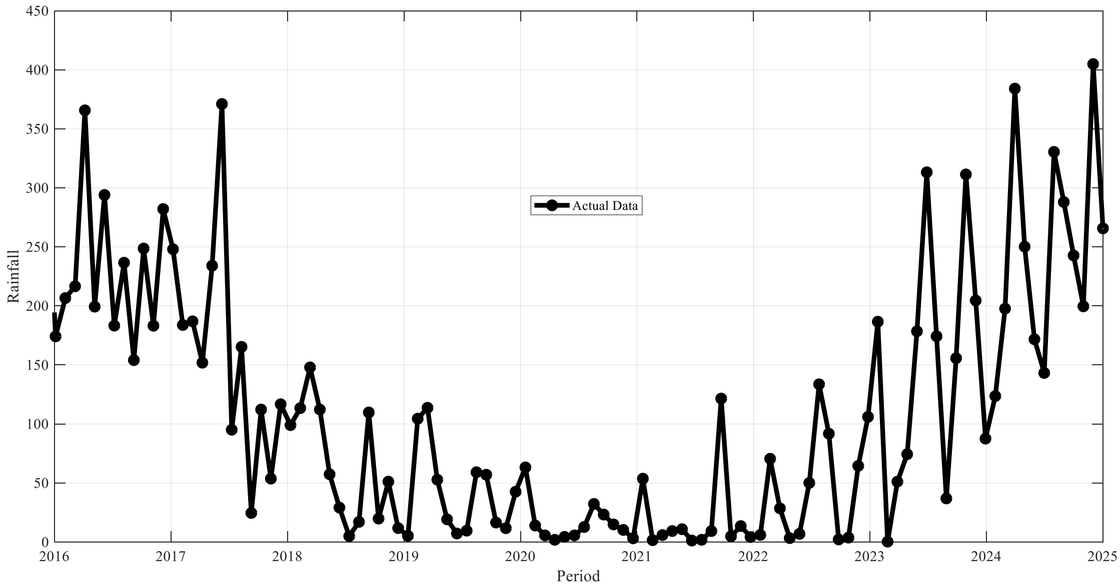

Figure 2. Actual data

This study focused on comparing three artificial intelligence models and their weighted hybrid counterpart under a consistent experimental setting. Conventional statistical forecasting methods, such as seasonal naïve forecasting, moving average, ARIMA, or SARIMA, were not included as baseline models. Therefore, the reported performance comparisons should be interpreted as comparisons among machine learning approaches rather than against traditional forecasting techniques. Future studies are encouraged to incorporate baseline forecasting models to provide a more comprehensive performance evaluation.

3. RESULTS AND DISCUSSION

3.1. Results

The performance evaluation of the predictive models used in this study was conducted by comparing the performance of three main algorithms: ACO, BPNN, XGBoost, and the

ISI Journal of Information Systems and Informatics
Published By Asosiasi Doktor Sistem Informasi Indonesia

Hybrid ACO-BPNN-XGBoost model. Each algorithm was tested using rainfall data, each of which had different patterns and levels of volatility. The testing was conducted by evaluating five key metrics: MSE, MAE, RMSE, MAPE, and R^2. These five indicators were used to describe the level of prediction error, where lower values indicate more accurate and stable algorithm performance.

The data used in this study consist of monthly rainfall time-series observations obtained from the Central Statistics Agency (BPS), the Meteorological, Climatological, and Geophysical Agency (BMKG), and the NASA POWER Data Access Viewer for Alas Subdistrict, Sumbawa Regency, West Nusa Tenggara Province, located at coordinates 8.5187° South Latitude and 116.9934° East Longitude. In addition, the Standardized Precipitation Index (SPI) was utilized as a supporting indicator to describe rainfall variability and drought conditions. The dataset covers a ten-year period from January 2016 to December 2025, resulting in a total of 120 monthly observations. Prior to modeling, the data underwent preprocessing procedures including data organization, cleaning, normalization, and transformation into a suitable format for time-series forecasting. Because the dataset represents chronological observations, a time-series hold-out strategy was applied to preserve temporal dependencies. The training dataset consisted of 96 monthly observations (80%), while the testing dataset contained 24 monthly observations (20%). This data partitioning was used to objectively evaluate the predictive performance and generalization capability of the proposed forecasting models. The prediction results for rainfall data using the ACO method are presented in detail in Table 3.

Table 3. ACO Method: Training Results (80%) and Testing Results (20%)

No	Results	MSE	MAE	RMSE	MAPE	R^2
1	Training (80%)	2876.3	42.971	53.631	299.03	0.77942
2	Testing (20%)	11752	90.885	108.41	2836.5	-0.065727

Table 3 presents the forecasting performance of the ACO model on both training and testing datasets. The model achieved an MSE of 2876.3, MAE of 42.971 mm, RMSE of 53.631 mm, MAPE of 299.03%, and R^2 of 0.77942 on the training data, indicating that the model

ISI Journal of Information Systems and Informatics
Published By Asosiasi Doktor Sistem Informasi Indonesia

was able to capture a substantial proportion of the rainfall variability during the training phase. However, when evaluated on the testing dataset, the prediction errors increased considerably, with an MSE of 11752, MAE of 90.885 mm, RMSE of 108.41 mm, and MAPE of 2836.5%, while the R^2 value decreased to −0.065727. Although the training performance appears satisfactory, the negative testing R^2 value indicates that the forecasting model explained less variability than a simple predictor based on the mean of the observed rainfall values. This result suggests limited generalization capability and provides evidence that the model was unable to maintain its predictive performance when evaluated using unseen observations. Furthermore, the large discrepancy between training and testing performance indicates the presence of overfitting. The extremely large MAPE value should be interpreted cautiously because monthly rainfall observations occasionally approach zero, causing percentage-based errors to become unstable and disproportionately large. Furthermore, the visualization of the actual, training, and test data for the ACO method is presented in Figure 3.

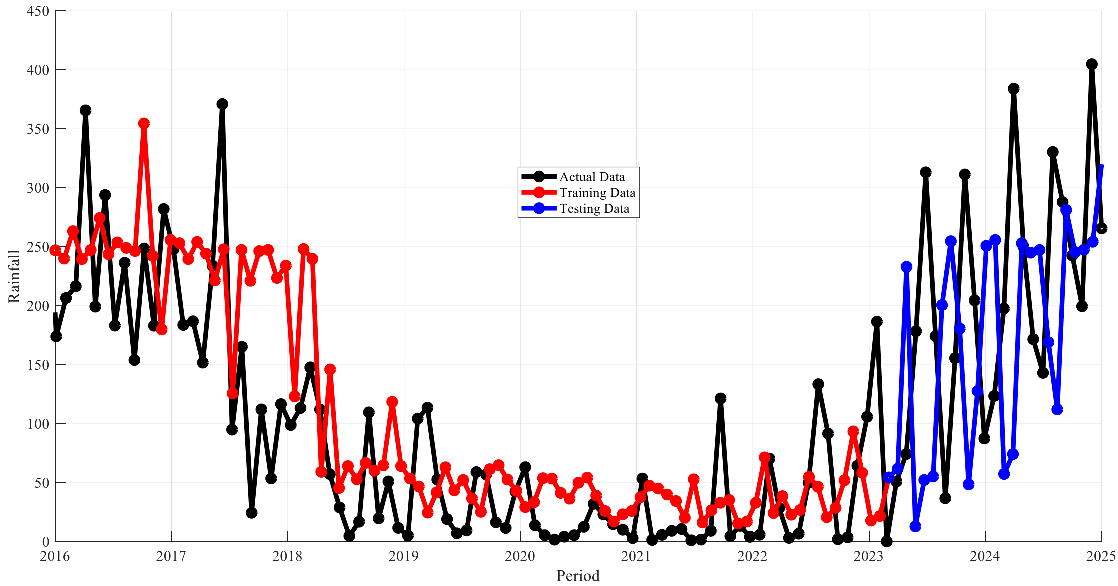

Figure 3. Actual data, training, and testing of the ACO method

Figure shows a comparison between actual data (black line), training data (red line), and testing data (blue line) for the period from 2016 to 2025. In general, the actual data shows a fairly high degree of fluctuation at the beginning of the period, then declines until around 2019–2021, before rising again and showing greater variation during the 2022–2025 period. During the training phase, the model tends to follow the general pattern of the actual data, although there are still some deviations at extreme points. Meanwhile, during the testing phase, the model is still able to capture the upward trend in the final period, but there is a fairly clear discrepancy between the prediction results and the actual data. This indicates that while the model is capable of representing the main patterns of the data, there are still limitations in capturing fluctuations and extreme variations. Overall, the model demonstrates fairly good performance in following data trends but is not yet optimal in terms of prediction accuracy on the test data. A quantitative evaluation was then conducted using MSE, MAE, RMSE, MAPE, and R^2 values using the BPNN method, as presented in Table 4.

Table 4. BPNN Method: Training Results (80%) and Testing Results (20%)

No	Results	MSE	MAE	RMSE	MAPE	R^2
1	Training (80%)	3400.1	44.828	58.311	248.27	0.77942
2	Testing (20%)	11954	93.444	109.33	2900.5	-0.083939

Table 4 shows the results of the model performance evaluation on the training (80%) and test (20%) datasets. It can be seen that during the training phase, the model achieved an MSE of 3,400.1, an MAE of 44.828, an RMSE of 58.311, a MAPE of 248.27, and a coefficient of determination (R^2) of 0.77942. An R^2 value close to 1 indicates that the model is quite capable of explaining the variation in the training data with a relatively good level of accuracy. However, on the test data, there was a significant decline in performance, as evidenced by an increase in MSE to 11,954, MAE to 93.444, RMSE to 109.33, and a very high MAPE of 2,900.5, accompanied by a negative R^2 value of -0.083939. This indicates that the model is unable to generalize well to new data; in fact, its performance is worse than that of a simple mean-based model. The negative testing R^2 value indicates that the forecasting model explained less variability than a simple predictor based on the mean of the observed rainfall values. This finding demonstrates limited generalization capability when the model was evaluated using unseen data. Moreover, the considerable difference between the training and testing performance provides further evidence of overfitting, indicating that the model adapted too closely to the training data while failing to capture consistent rainfall patterns in the testing dataset. Furthermore, the actual data, training, and testing visualizations for the BPNN method are presented in Figure 4.

ISI Journal of Information Systems and Informatics
Published By Asosiasi Doktor Sistem Informasi Indonesia

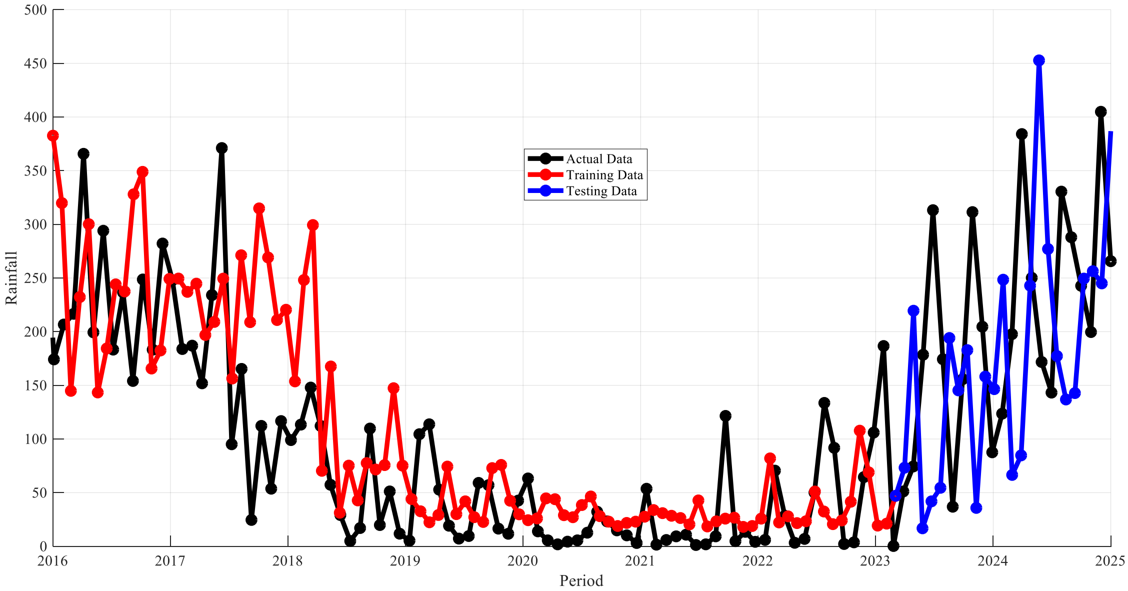

Figure 4. Actual data, training, and testing of the BPNN method

Figure 4 shows a comparison between actual data, training data, and test data in a time series covering the period from 2016 to 2025. Overall, the actual data is characterized by relatively high fluctuations at the beginning of the period, followed by a gradual decline until around 2019–2021, and then a resurgence with considerable variation toward the end of the observation period. In the training data, the model shows a tendency to follow the main pattern of the actual data in the early to mid-phase, although deviations are still found in certain extreme values. As for the test data, the model is still able to represent the upward trend in the final period, but there are significant differences in values between the prediction results and the actual data, particularly at points with high fluctuations. This indicates that the model has successfully learned the general characteristics of the historical data but is not yet fully effective at capturing extreme variations, so the model's generalization ability still needs improvement. A quantitative evaluation was then conducted using MSE, MAE, RMSE, MAPE, and R^2 metrics with the XGBoost method, as presented in Table 5.

Table 5. XGBoost Method: Training Results (80%) and Testing Results (20%)

No	Results	MSE	MAE	RMSE	MAPE	R^2
1	Training (80%)	553.05	8.0328	23.517	33.742	0.95759
2	Testing (20%)	11927	86.16	109.21	2073.9	-0.081496

ISI Journal of Information Systems and Informatics
Published By Asosiasi Doktor Sistem Informasi Indonesia

Table 5 presents the results of the model performance evaluation on the training (80%) and testing (20%) datasets. It was found that during the training phase, the model produced an MSE of 553.05, an MAE of 8.0328, RMSE of 23.517, MAPE of 33.742, and a coefficient of determination (R^2) of 0.95759. An R^2 value close to 1 indicates that the model has excellent ability to explain the variation in the training data, allowing historical data patterns to be learned with sufficient accuracy. However, during the testing phase, there was a very significant drop in performance, marked by an increase in MSE to 11,927, MAE to 86.16, RMSE to 109.21, and a very high MAPE of 2,073.9, accompanied by a negative R^2 value of -0.081496. This indicates that the model is unable to generalize to new data and actually provides worse predictions than a simple mean-based approach. Although XGBoost achieved the highest training R^2 among the individual models, the negative testing R^2 value indicates that the forecasting model explained less variability than a simple predictor based on the mean of the observed rainfall values. This result demonstrates that the model did not generalize adequately to unseen observations. Furthermore, the substantial difference between the training and testing performance indicates overfitting despite the excellent fit achieved during the training phase. Furthermore, the actual data, training, and testing visualizations for the XGBoost method are presented in Figure 5.

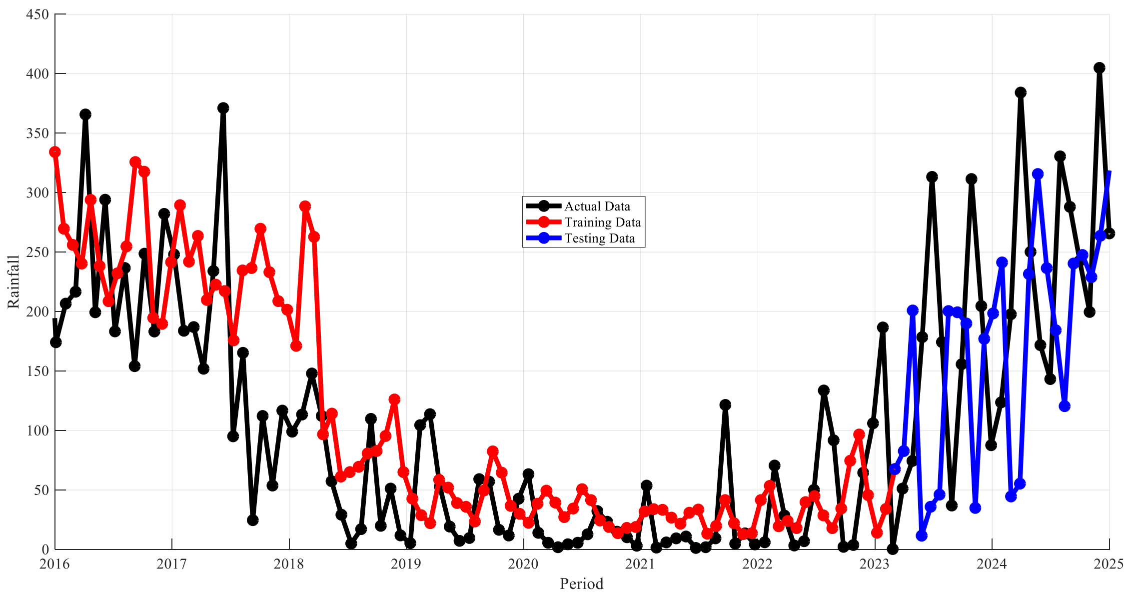

Figure 5. Actual data, training, and testing of the XGBoost method

ISI Journal of Information Systems and Informatics
Published By Asosiasi Doktor Sistem Informasi Indonesia

Figure 5 shows a comparison between actual data, training data, and testing data for the time series covering the period from 2016 to 2025. In general, the actual data (black line) shows a fairly high fluctuating pattern at the beginning of the period, then experiences a gradual decline until around 2019–2021, before rising again with significant variation in the final period. The training data (red line) in this figure appears to closely follow the dynamics of the actual data in the early to mid-phase, although differences are still visible at several extreme points, indicating inaccuracies in the estimation of peak and trough values. Meanwhile, the testing data (blue line) shows that the model is able to capture the upward trend in the final period, but there is still a fairly clear deviation between the prediction results and the actual data, especially during value spikes. Overall, the model demonstrates a fairly good ability to learn historical data patterns, but it still has limitations in representing extreme fluctuations, indicating that the model's generalization ability to new data is not yet fully optimal. A quantitative evaluation was then conducted using MSE, MAE, RMSE, MAPE, and R^2 values via the Hybrid method, as presented in Table 6.

Table 6. Hybrid Method: Training Results (80%) and Testing Results (20%)

No	Results	MSE	MAE	RMSE	MAPE	R^2
1	Training (80%)	1589.6	30.534	39.869	191.77	0.8781
2	Testing (20%)	11253	86.94	106.08	2602.2	-0.020456

Table 6 presents the results of the model performance evaluation on the training (80%) and testing (20%) datasets. It was found that during the training phase, the model produced an MSE of 1,589.6, an MAE of 30.534, RMSE of 39.869, MAPE of 191.77, and a coefficient of determination (R^2) of 0.8781. An R^2 value close to 1 indicates that the model has a fairly good ability to represent patterns in the training data, so that most of the data variation can be explained by the model. However, during the testing phase, there was a significant drop in performance, marked by an increase in MSE to 11,253, MAE to 86.94, RMSE to 106.08, and a very high MAPE of 2,602.2, accompanied by a negative R^2 value of -0.020456. This indicates that the model is unable to generalize well to new data, resulting in predictions on the test data that are less accurate and even worse than those from the baseline approach. Although the hybrid model achieved the lowest testing RMSE

ISI Journal of Information Systems and Informatics
Published By Asosiasi Doktor Sistem Informasi Indonesia

among all evaluated models, the negative testing R^2 value indicates that it still explained less variability than a simple predictor based on the mean of the observed rainfall values. Therefore, the improvement in RMSE should be regarded as modest rather than conclusive. Furthermore, the difference between the training and testing performance indicates that the hybrid model also experienced overfitting and was unable to generalize sufficiently to unseen rainfall observations. Furthermore, the actual data, training, and testing results of the Hybrid method are visualized in Figure 6.

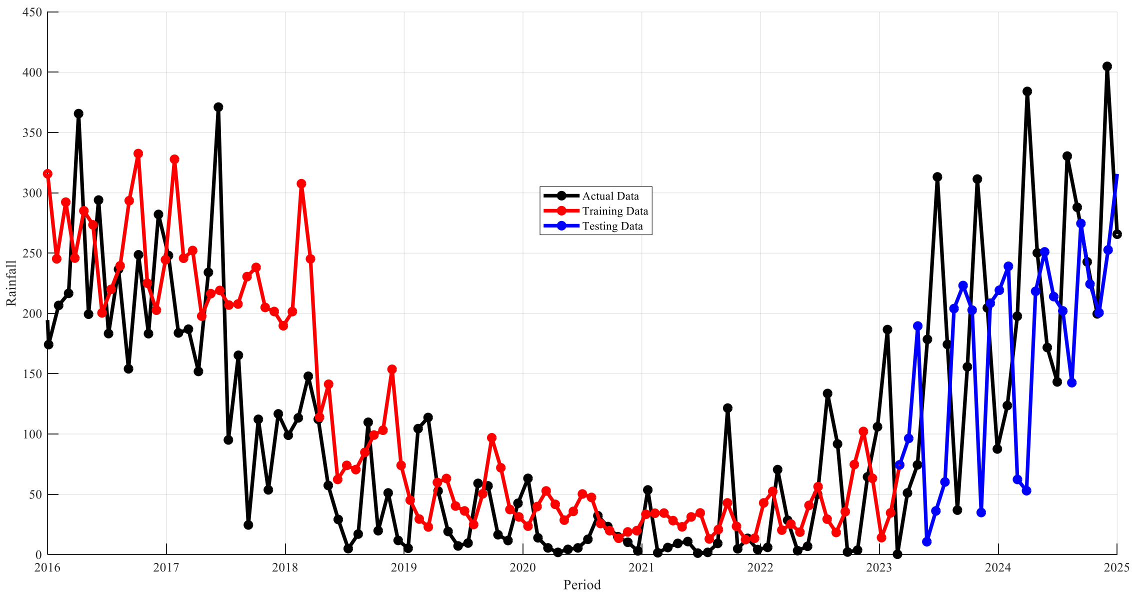

Figure 6. Actual data, training, and testing of the Hybrid method

Figure 6 shows a comparison between actual data, training data, and testing data for the time series covering the period from 2016 to 2025. In general, the actual data (black line) shows a fairly high fluctuation pattern at the beginning of the period, then gradually declines until around 2019–2021, before rising again with greater variation in the final observation period. During the training phase, the model (red line) appears capable of following the general pattern of the actual data from the early to mid-phase, although deviations still occur at some extreme points, indicating inaccuracies in capturing the data's peaks and troughs. Meanwhile, during the testing phase, the model (blue line) demonstrated the ability to capture the upward trend direction in the final period; however, there remains a fairly clear discrepancy between the predicted values and the actual data, particularly during sharp spikes in values. This indicates that the model has successfully learned the general characteristics of the historical data but remains less

ISI Journal of Information Systems and Informatics
Published By Asosiasi Doktor Sistem Informasi Indonesia

responsive to extreme variations and sudden changes in the data. Overall, the model demonstrates fairly good trend-representation capabilities; however, the accuracy rate during the testing phase still needs to be improved to achieve more stable prediction results and better generalization. To provide a comprehensive assessment of model performance, the testing results of the ACO, BPNN, XGBoost, and hybrid ACO-BPNN-XGBoost models were compared using the RMSE and coefficient of determination (R^2) metrics. This comparison allows the evaluation of the effectiveness of the proposed hybrid approach relative to the individual forecasting models. The comparative results are presented in Table 7.

Table 7. Summary of Training and Testing Performance of All Forecasting Models

Model	Dataset	MSE	MAE	RMSE	MAPE (%)	R^2
ACO	Training	2876.3	42.971	53.631	299.03	0.7794
ACO	Testing	11752	90.885	108.41	2836.5	-0.0657
BPNN	Training	3400.1	44.828	58.311	248.27	0.7794
BPNN	Testing	11954	93.444	109.33	2900.5	-0.0839
XGBoost	Training	553.05	8.0328	23.517	33.742	0.9576
XGBoost	Testing	11927	86.160	109.21	2073.9	-0.0815
Hybrid	Training	1589.6	30.534	39.869	191.77	0.8781
Hybrid	Testing	11253	86.940	106.08	2602.2	-0.0205

Table 7 summarizes the overall forecasting performance of all evaluated models on both the training and testing datasets. Among the individual models, XGBoost achieved the best performance during training, with the lowest RMSE and the highest R^2 value, indicating its superior ability to fit the historical rainfall observations. The proposed hybrid ACO-BPNN-XGBoost model achieved the lowest testing RMSE (106.08 mm), representing only a modest improvement compared with ACO (108.41 mm), BPNN (109.33 mm), and XGBoost (109.21 mm). However, all models produced negative testing R^2 values, indicating that none of the evaluated forecasting approaches generalized sufficiently to outperform a simple mean-based predictor on previously unseen observations. Therefore, the improvement achieved by the hybrid model should be interpreted as incremental rather than conclusive. A comparison between the historical rainfall data and the projected values is presented in Figure 7.

ISI Journal of Information Systems and Informatics
Published By Asosiasi Doktor Sistem Informasi Indonesia

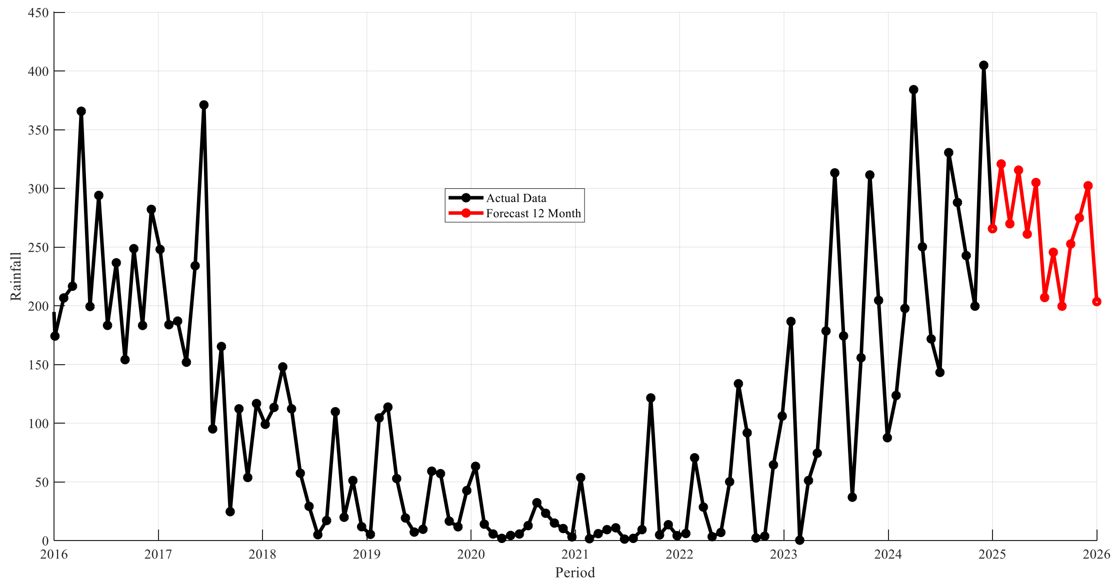

Figure 7. Actual data and 12-month forecast

Figure 7 shows a time series of rainfall consisting of observational data and 12-month-ahead forecast results, revealing significant variations throughout the observation period. In the actual data, it is evident that during the initial period there were high fluctuations with several extreme peaks reaching approximately 350–370 mm, whereas during the 2018–2021 period there was a fairly sharp decline with dominant values in the low range, namely around 0–50 mm to less than 100 mm. Subsequently, during the 2022–2024 period, there was a resurgence marked by greater fluctuations, with some maximum values approaching 300–400 mm. Meanwhile, the 12-month forecast results show a relatively more stable pattern compared to historical data, with values ranging from approximately 200–320 mm and no extreme spikes as seen in previous periods. Overall, these results indicate that the model is capable of capturing the general patterns of rainfall dynamics, though with a tendency to dampen extreme variations observed in the actual data.

After interpreting the patterns of actual rainfall data and the forecast results, the next step is to develop mathematical formulations for each method used in the hybrid model. The development of these mathematical models aims to systematically explain the working mechanisms of each prediction method before the entire model is integrated into the final hybrid system. In this study, the hybrid approach is built based on three main methods, namely Ant Colony Optimization (ACO), Backpropagation Neural Network

ISI Journal of Information Systems and Informatics
Published By Asosiasi Doktor Sistem Informasi Indonesia

(BPNN), and XGBoost, where each model has a different mathematical structure according to the characteristics of its algorithm. The implementation of this study is represented through a neural network approach with a tanh activation function. In general, the ACO model equation is formulated as shown in Equations 14 to 18.

$$A(t) = W_{A2} \cdot tansig(W_{A1}x(t-1) + b_{A1}) + b_{A2} \tag{14}$$

$$W_{A1} = \begin{bmatrix} -14.184799 \\ -14.167624 \\ 15.298546 \\ 16.437953 \\ 14.695910 \\ 13.627550 \\ -14.796746 \\ 14.985190 \\ -14.707649 \\ -13.616150 \end{bmatrix} \tag{15}$$

$$b_{A1} = \begin{bmatrix} 13.787281 \\ 10.791332 \\ -4.728961 \\ -3.531730 \\ 14.695910 \\ 13.627550 \\ -14.796746 \\ 14.985190 \\ -14.707649 \\ -13.616150 \end{bmatrix} \tag{16}$$

$$W_{A2} = \begin{bmatrix} -0.611232 \\ 0.070218 \\ 0.374846 \\ 1.742563 \\ -2.555512 \\ 1.304008 \\ -0.963455 \\ -0.720950 \\ -0.131907 \\ -0.758029 \end{bmatrix} \tag{17}$$

$$b_{A2} = [-0.494713] \tag{18}$$

A(t) is the prediction result of the ACO model at time t, x(t-1) is the rainfall data from the previous period, W_{A1} and W_{A2} are the network weights, while b_{A1} and b_{A2} are the bias parameters. This model works by optimizing network parameters using the principle of best-solution search based on ant colony behavior, thereby improving predictive

ISI Journal of Information Systems and Informatics
Published By Asosiasi Doktor Sistem Informasi Indonesia

capability. The BPNN also uses an artificial neural network structure with a tanh nonlinear activation function. The general equation for the BPNN is formulated as shown in Equations 19 to 23.

$$B(t) = W_{B2} \cdot \text{tansig}(W_{B1} x(t-1) + b_{B1}) + b_{B2} \tag{19}$$

$$W_{B1} = \begin{bmatrix} -14.204988 \\ 15.082085 \\ -14.291744 \\ 14.110935 \\ 13.441325 \\ -14.071124 \\ -14.269289 \\ -13.557652 \\ 14.080479 \\ 13.797424 \end{bmatrix} \tag{20}$$

$$b_{B1} = \begin{bmatrix} 13.757113 \\ -9.400793 \\ 7.531424 \\ -5.056934 \\ -3.911434 \\ -0.777347 \\ -4.546627 \\ -8.704505 \\ 10.761109 \\ 14.176961 \end{bmatrix} \tag{21}$$

$$W_{B2} = \begin{bmatrix} -0.346788 \\ -0.163082 \\ -0.508554 \\ -0.648033 \\ 0.319144 \\ -0.128398 \\ -0.278184 \\ -0.092353 \\ 0.031583 \\ 0.128021 \end{bmatrix} \tag{22}$$

$$b_{B2} = [-0.074756] \tag{23}$$

$B(t)$ is the BPNN prediction output, W_{B1} and W_{B2} are the network weights, and b_{B1} and b_{B2} are the model biases. The BPNN method performs the learning process through a mechanism of forward propagation and backpropagation, so that the network weights can be optimally updated. The decision tree-based ensemble boosting method builds

ISI Journal of Information Systems and Informatics
Published By Asosiasi Doktor Sistem Informasi Indonesia

predictions incrementally through the accumulation of weak learners. Mathematically, this model is formulated as shown in Equations 24.

$$C(t) = \sum_{m=1}^{50} \gamma_m h_m\big(x(t-1)\big) \tag{24}$$

C(t) is the XGBoost prediction, γ_m is the contribution weight of the m-th weak learner, and $h_m\big(x(t-1)\big)$ is the prediction function of each tree in the boosting iteration. In this study, 50 estimators were used to build a more robust prediction model. After each individual prediction model, namely Ant Colony Optimization (ACO), Backpropagation Neural Network (BPNN), and XGBoost, was mathematically formulated, the next step was to develop a mechanism for combining these three models into an integrated hybrid system. This process was carried out through a weight estimation procedure based on the predictive performance of each model using the Mean Squared Error (MSE) criterion. Consequently, models with lower prediction errors received larger contributions in the final ensemble prediction. The weighting mechanism was designed to objectively assign the contribution of each model according to its forecasting capability. Although training MSE was employed in this study due to data availability constraints, the use of validation or out-of-fold prediction errors is generally recommended because it provides a more reliable estimate of model generalization performance and reduces the risk of overfitting. The weight estimation method is formulated as shown in Equations 25.

$$w_i = \frac{(1/MSE_i)}{\sum(1/MSE)} \tag{25}$$

Where w_i denotes the weight assigned to the i-th model, while MSE_i represents the Mean Squared Error produced by the corresponding model. The inverse-MSE weighting strategy ensures that models with higher predictive accuracy receive proportionally larger contributions in the ensemble. As a result, the hybrid framework can objectively integrate the strengths of multiple forecasting models according to their respective performance levels. It should be noted that the weighting scheme reflects the relative predictive capability of each constituent model, thereby allowing the hybrid framework to exploit the advantages of ACO, BPNN, and XGBoost simultaneously. Once the optimal weights are obtained, the outputs of the three models are combined through a weighted

ISI Journal of Information Systems and Informatics
Published By Asosiasi Doktor Sistem Informasi Indonesia

linear combination to generate the final hybrid prediction. In general, the hybrid forecasting model can be expressed as shown in Equations 26.

$$H(t) = w_1 A(t) + w_2 B(t) + w_3 C(t) \quad (26)$$

Where $H(t)$ is the final hybrid prediction, $A(t)$ is the output of the ACO model, $B(t)$ is the output of the BPNN model, and C(t) is the output of the XGBoost model. The coefficients w_1, w_2, and w_3 represent the contribution weights of each constituent model and satisfy the constraint $w_1 + w_2 + w_3 = 1$. Based on the results of the numerical estimation, the final hybrid equation is obtained as shown in Equations 27.

$$H(t) = 0.141911A(t) + 0.120046B(t) + 0.738042C(t) \quad (27)$$

The numerical results show that the XGBoost model makes the largest contribution to the final prediction with a weight of 0.738042, followed by ACO at 0.141911 and BPNN at 0.120046. This indicates that XGBoost delivers the most dominant performance in the hybrid system, while ACO and BPNN continue to serve as supporting components to enhance the overall stability and accuracy of the predictions. Consequently, the resulting hybrid model is able to optimize the strengths of each method to produce more accurate, robust, and adaptive rainfall predictions.

After the hybrid mathematical model was formulated and the final equations were derived, the next step was to implement the model to generate rainfall predictions for the coming period. The resulting predictions are then presented in the form of a forecast table to provide a quantitative overview of rainfall values for each forecast month along with their category classifications. The presentation of this table aims to facilitate the systematic interpretation of the forecast results, so that it not only displays the numerical values of rainfall but also groups rainfall conditions based on specific relevant categories. To facilitate the interpretation of the forecast outputs, the predicted monthly rainfall values were classified into rainfall categories. Rainfall values below 200 mm/month were categorized as light rainfall, values between 200 and 300 mm/month were categorized as moderate rainfall, and values exceeding 300 mm/month were categorized as heavy rainfall. These categories are intended solely for descriptive interpretation and do not represent official operational classifications issued by meteorological agencies. Thus, the forecast table and rainfall categories serve as the final

ISI Journal of Information Systems and Informatics
Published By Asosiasi Doktor Sistem Informasi Indonesia

representation of the hybrid model's implementation in supporting a more structured, informative, and practical analysis of future rainfall conditions, which will be presented in Table 8.

Table 8. Prediction Data and Categories

Month	Forecast	Category
January 2026	320.78	heavy
February 2026	269.82	moderate
March 2026	315.54	heavy
April 2026	261.01	moderate
May 2026	304.98	heavy
Juni 2026	207.01	moderate
July 2026	245.56	moderate
August 2026	199.61	light
September 2026	252.61	moderate
October 2026	274.92	moderate
November 2026	302.26	heavy
December 2026	203.49	moderate

Based on the prediction results in Table 8, the projected monthly rainfall for one year is presented along with the classification of rainfall intensity into specific categories. Overall, the predicted rainfall values range from approximately 199.61 to 320.78 mm, indicating variations in intensity from low to high. Several months—namely the 1st, 3rd, 5th, and 11th months—show relatively high rainfall values, placing them in the heavy rain category, while most other months fall into the moderate rain category with values ranging from approximately 203.49 to 274.92 mm. Only one month, the 8th month, showed the lowest value and was classified as light rainfall with a value of approximately 199.61 mm. In general, this pattern indicates that rainfall distribution during the forecast period is fluctuating but tends to be dominated by the moderate intensity category, with several spikes reflecting significant increases during specific periods. Therefore, these results can serve as a strong scientific basis for developing more targeted environmental adaptation policies, agricultural management, and climate risk reduction strategies.

3.2. Discussion

Research on artificial intelligence-based rainfall prediction continues to advance as the need for more accurate and adaptive prediction systems grows. [22] notes that the Backpropagation Neural Network (BPNN) method is quite effective at learning historical rainfall patterns; however, the model's performance during the testing phase still declines due to limited generalization capabilities regarding changes in weather patterns. [23] also states that the neural network approach is more effective at capturing nonlinear relationships in rainfall data compared to conventional statistical methods. Additionally, [24] explains that integrating optimization algorithms with artificial intelligence methods can improve the stability of predictions for rainfall data with high levels of fluctuation. These findings align with the results of [25], where the ACO and BPNN methods demonstrated good performance during the training phase but still experienced a decline in performance when tested using new data with different patterns.

The results consistently indicate that all forecasting models achieved substantially better performance during the training phase than during the testing phase, suggesting the presence of overfitting. This phenomenon is likely attributable to the relatively limited dataset comprising only 120 monthly rainfall observations. Under such conditions, complex machine learning models tend to learn patterns that are highly specific to the training data while exhibiting limited capability to generalize to unseen observations. Moreover, the highly nonlinear and nonstationary characteristics of tropical rainfall further increase the difficulty of developing robust forecasting models. These characteristics also explain why the predicted rainfall series appeared smoother than the historical observations, as the models primarily captured the dominant temporal trend while failing to reproduce abrupt rainfall spikes and extreme fluctuations.

Advances in ensemble learning and boosting methods have also made a significant contribution to improving the accuracy of rainfall predictions. [26] states that the XGBoost algorithm is highly effective at modeling nonlinear patterns, resulting in a high level of prediction accuracy. Research [27] also shows that XGBoost-based approaches are capable of producing more stable predictions compared to several conventional machine learning methods. Furthermore, [28] reveals that XGBoost outperforms Random Forest in predicting rainfall intensity in regions with complex rainfall characteristics. The results of the study [29] further reinforce these findings by showing that hyperparameter

ISI Journal of Information Systems and Informatics
Published By Asosiasi Doktor Sistem Informasi Indonesia

optimization in XGBoost can enhance the model's ability to minimize prediction error. The results of the present study demonstrate a similar tendency, where the XGBoost model achieved the highest training R^2 value among the individual models. This finding suggests that XGBoost was effective in capturing nonlinear relationships within the rainfall data and contributed to the overall performance of the proposed hybrid model. The weighting mechanism adopted in the hybrid model assigned the largest contribution to XGBoost because this model achieved the lowest training prediction error among the three constituent models. Consequently, the inverse-error weighting strategy naturally produced a larger weight for XGBoost than for ACO and BPNN. Nevertheless, the larger contribution of XGBoost should not be interpreted as evidence of superior generalization capability because the testing results remained unsatisfactory, as reflected by the negative R^2 values. Although ACO and BPNN contributed smaller weights to the hybrid model, they still provided complementary predictive information through optimization and nonlinear pattern learning, respectively. However, under the present experimental conditions, their incremental contribution to the final prediction remained relatively limited.

The findings of the present study are consistent with the development of modern machine learning approaches, which indicate that ensemble algorithms such as XGBoost are capable of providing more stable predictive performance than single models when dealing with nonlinear data structures [30]. Moreover, the Ant Colony Optimization approach proposed by [31] has demonstrated considerable effectiveness in improving solution quality within complex parameter search spaces. The slight improvement achieved by the proposed hybrid model suggests that combining optimization and ensemble learning techniques can contribute to reducing forecasting errors. Nevertheless, as emphasized by [32], the evaluation of time-series models should employ validation schemes that preserve temporal ordering to obtain more reliable estimates of generalization performance. Furthermore, [33] highlighted that the selection of appropriate forecasting accuracy measures plays a crucial role in the interpretation and comparison of model performance. Therefore, the performance improvements observed in this study should be interpreted cautiously, considering both the evaluation methodology and the limitations of the available dataset. Another important aspect concerns the interpretation of the MAPE values reported in this study. The extremely large MAPE values obtained during testing should be interpreted cautiously because several monthly rainfall observations were close to zero, particularly during the dry season. Under these conditions, percentage-based error measures become numerically unstable and may substantially exaggerate forecasting errors. Therefore, RMSE and R^2 provide more reliable indicators for comparing the forecasting performance of rainfall prediction models in the present study.

Despite achieving the lowest RMSE among the evaluated forecasting models, the proposed hybrid ACO-BPNN-XGBoost model produced only a modest improvement over the best individual models. Specifically, the testing RMSE decreased only slightly from 108.41 mm (ACO) and 109.21 mm (XGBoost) to 106.08 mm for the hybrid model. Although this reduction demonstrates that combining multiple predictive models can reduce forecasting error, the practical significance of the improvement remains limited. Furthermore, all forecasting models produced negative testing R^2 values, indicating that their predictive performance remained inferior to that of a simple mean-based predictor when evaluated using unseen observations. Consequently, the proposed hybrid framework should be regarded as a preliminary forecasting approach rather than a definitive predictive solution. Several limitations should also be acknowledged. The study utilized rainfall observations from only one geographical region and a relatively small dataset consisting of 120 monthly observations. In addition, no exogenous climatic predictors were incorporated into the forecasting models, and comparisons with baseline forecasting approaches were not performed. These limitations may have contributed to the limited generalization capability observed in the testing phase. Therefore, future studies should employ larger multi-region datasets, incorporate additional climatic predictors, compare the proposed framework with baseline forecasting models, and adopt more rigorous validation procedures such as rolling-origin or walk-forward validation before practical implementation can be recommended.

4. CONCLUSION

Based on the performance evaluation results of the ACO, BPNN, XGBoost, and hybrid ACO-BPNN-XGBoost models, all models demonstrated satisfactory performance during the training phase but experienced a decline during the testing phase, as indicated by increased prediction errors and negative R^2 values, suggesting limited generalization capability and the presence of overfitting. Among the individual models, XGBoost exhibited relatively better performance, while the proposed hybrid ACO-BPNN-XGBoost model achieved the lowest RMSE value of 106.08 mm. However, the improvement was only modest compared with the best individual model, with the testing RMSE decreasing from 108.41 mm (ACO) and 109.21 mm (XGBoost) to 106.08 mm for the hybrid model. Although the hybrid model achieved the lowest testing RMSE among the evaluated models, the negative testing R^2 values indicate weak generalization capability, demonstrating that the proposed forecasting models were unable to outperform a simple mean-based predictor when evaluated using previously unseen observations. Therefore, the findings of this study should be regarded as preliminary rather than conclusive, and the proposed hybrid framework should not yet be considered ready for practical implementation.

Furthermore, the present study did not compare the proposed hybrid model with simple baseline forecasting approaches, such as seasonal naïve forecasting, which should be outperformed before practical applicability can be claimed. In addition, the relatively small monthly dataset and the evaluation conducted in a single study area represent important limitations of this research. Future studies should employ larger datasets, incorporate exogenous climatic predictors, such as temperature, humidity, or large-scale climate indices, adopt more rigorous validation procedures including walk-forward or rolling-origin validation, compare the proposed approach with baseline forecasting models, and perform external validation using data from multiple geographical regions to improve the robustness and generalization capability of the forecasting framework.

REFERENCES

[1] A. M. Elshewey, A. A. Alhussan, and D. S. Khafaga, "An enhanced adaptive dynamic metaheuristic optimization algorithm for rainfall prediction depends on long short-term memory," *PLoS One*, vol. 20, no. 6, p. e0317554, 2025, doi: 10.1371/journal.pone.0317554.

[2] M. N. Tahroudi, "Comprehensive global assessment of precipitation trend and pattern variability considering their distribution dynamics," *Sci. Rep.*, vol. 15, no. 1, pp. 1–19, 2025, doi: 10.1038/s41598-025-06050-5.

[3] R. J. Hyndman and G. Athanasopoulos, *Forecasting: Principles and Practice*, 3rd ed. Melbourne, Australia: OTexts, 2021. [Online]. Available: https://otexts.com/fpp3/

[4] M. Arief, F. Istiyanto, and P. Palloan, "Monthly Rainfall Prediction Using the Backpropagation Neural Network (BPNN) Algorithm in Maros Regency," *Sci. J. Informatics*, vol. 10, no. 1, pp. 13–24, 2025, doi: 10.15294/sji.v10i1.37982.

[5] E. M. Fischer and R. Knutti, "Observed heavy precipitation increase confirms theory and early models," *Nat. Clim. Chang.*, vol. 6, no. 11, pp. 986–991, 2016, doi: 10.1038/nclimate3110.

[6] G. E. P. Box, G. M. Jenkins, G. C. Reinsel, and G. M. Ljung, *Time Series Analysis: Forecasting and Control*, 5th ed. Hoboken, NJ, USA: John Wiley & Sons, 2015.

[7] M. Ehsani, A. Zarei, H. Gupta, K. Barnard, and A. Behrangi, "NowCasting-Nets: Representation Learning to Mitigate Latency Gap of Satellite Precipitation Products Using Convolutional and Recurrent Neural Networks," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, no. 4706021, pp. 1–21, 2021, doi: 10.1109/TGRS.2022.3158888.

[8] B. Oldfield and S. Kandanaarachchi, "SoftwareX An Item Response Theory-based R module for Algorithm Portfolio Analysis," *SoftwareX*, vol. 31, no. 1, p. 102239, 2025, doi: 10.1016/j.softx.2025.102239.

[9] J. E. Nash and J. V Sutcliffe, "River flow forecasting through conceptual models Part I—A discussion of principles," *J. Hydrol.*, vol. 10, no. 3, pp. 282–290, 1970, doi: 10.1016/0022-1694(70)90255-6.

[10] I. Aljarah, H. Faris, and S. Mirjalili, "Optimizing connection weights in neural networks using the whale optimization algorithm," *Soft Comput.*, vol. 22, no. 1, pp. 1–15, 2018, doi: 10.1007/s00500-016-2442-1.

[11] S. Renuga Devi, P. Arulmozhivarman, C. Venkatesh, and P. Agarwal, "Performance comparison of artificial neural network models for daily rainfall prediction," *Int. J. Autom. Comput.*, vol. 13, no. 5, pp. 417–427, 2016, doi: 10.1007/s11633-016-0986-2.

[12] M. A. F. I. Aslim, Jasruddin, P. Palloan, Helmi, M. Arsyad, and H. Triwibowo, "Monthly Rainfall Prediction Using the Backpropagation Neural Network (BPNN) Algorithm in Maros Regency," *Sci. J. Informatics*, vol. 10, no. 1, pp. 1–10, 2023, doi: 10.15294/sji.v10i1.37982.

[13] H. V Gupta, H. Kling, K. K. Yilmaz, and G. F. Martinez, "Decomposition of the mean squared error and NSE performance criteria: Implications for improving hydrological modelling," *J. Hydrol.*, vol. 377, no. 1–2, pp. 80–91, 2009, doi: 10.1016/j.jhydrol.2009.08.003.

Journal of Information Systems and Informatics
Published By Asosiasi Doktor Sistem Informasi Indonesia

[14] S. S. Sammen *et al.*, "Rainfall modeling using two different neural networks improved by metaheuristic algorithms," *Environ. Sci. Eur.*, vol. 35, no. 1, pp. 1–25, 2023, doi: 10.1186/s12302-023-00818-0.

[15] S. O. Oyedepo *et al.*, "A critical review on enhancement and sustainability of energy systems : perspectives on thermo-economic and thermo-environmental analysis," vol. 12, no. 1417453, pp. 1–29, 2025, doi: 10.3389/fenrg.2024.1417453.

[16] D. N. Moriasi, J. G. Arnold, M. W. Van Liew, R. L. Bingner, R. D. Harmel, and T. L. Veith, "Model evaluation guidelines for systematic quantification of accuracy in watershed simulations," *Trans. ASABE*, vol. 50, no. 3, pp. 885–900, 2007, doi: 10.13031/2013.23153.

[17] S. Madadgar *et al.*, "A hybrid statistical-dynamical framework for meteorological drought prediction: Application to the southwestern United States," *Water Resour. Res.*, vol. 52, no. 7, pp. 5095–5110, 2016, doi: 10.1002/2015WR018547.

[18] H. Chen, V. Chandrasekar, H. Tan, and R. Cifelli, "Rainfall Estimation From Ground Radar and TRMM Precipitation Radar Using Hybrid Deep Neural Networks," *Geophys. Res. Lett.*, vol. 46, no. 17–18, pp. 10669–10678, 2019, doi: 10.1029/2019GL084771.

[19] S. Zhang, T. Chang, and D. Lin, "A Preliminary Study on a Hybrid Wavelet Neural Network Model for Forecasting Monthly Rainfall," vol. 14, no. 5, pp. 1747–1757, 2018, doi: 10.29333/ejmste/85119.

[20] V. Kumar, N. Kedam, and O. Kisi, "A comparative study of machine learning models for daily and weekly rainfall forecasting," *Water Resour. Manag.*, vol. 39, no. 1, pp. 271–290, 2025, doi: 10.1007/s11269-024-03969-8.

[21] J. F. Torres, D. Hadjout, A. Sebaa, F. Martínez-Álvarez, and A. Troncoso, "Deep Learning for Time Series Forecasting: A Survey," *Big Data*, vol. 9, no. 1, pp. 3–34, 2021, doi: 10.1089/big.2020.0159.

[22] Y. Wang, J. Liu, R. Li, X. Suo, and E. Lu, "Precipitation forecast of the Wujiang River Basin based on artificial bee colony algorithm and backpropagation neural network," *Alexandria Eng. J.*, vol. 59, no. 3, pp. 1473–1483, 2020, doi: 10.1016/j.aej.2020.04.035.

[23] M. Usman, S. Khan, K. Mohammad, and A. Hussain, "Results in Engineering Comparative analysis of different rainfall prediction models : A case study of Aligarh City, India," *Results Eng.*, vol. 22, no. 01, p. 102093, 2024, doi: 10.1016/j.rineng.2024.102093.

ISI Journal of Information Systems and Informatics
Published By Asosiasi Doktor Sistem Informasi Indonesia

[24] S. Dotse, I. Larbi, A. Limantol, and L. De Silva, “A review of the application of hybrid machine learning models to improve rainfall prediction,” *Model. Earth Syst. Environ.*, vol. 10, no. 1, pp. 19–44, 2023, doi: 10.1007/s40808-023-01835-x.

[25] A. D. Gorgij, O. Kisi, and S. Heddam, “Metaheuristic-optimized neuro-fuzzy models for meteorological drought prediction,” *Environ. Earth Sci.*, vol. 85, no. 192, pp. 1–24, 2026, doi: 10.1007/s12665-026-12910-8.

[26] S. A. Sahaar and J. D. Niemann, “Estimating rootzone soil moisture by fusing multiple remote sensing products with machine learning,” *Remote Sens.*, vol. 16, no. 19, p. 3699, 2024, doi: 10.3390/rs16193699.

[27] C. Bentéjac, A. Csörgő, and G. Martínez-Muñoz, “A comparative analysis of gradient boosting algorithms,” *Artif. Intell. Rev.*, vol. 54, no. 3, pp. 1937–1967, 2021, doi: 10.1007/s10462-020-09896-5.

[28] H. Li, Y. Zhang, H. Lei, and X. Hao, “Machine learning-based bias correction of precipitation measurements at high altitude,” *Remote Sens.*, vol. 15, no. 8, p. 2180, 2023, doi: 10.3390/rs15082180.

[29] K. Budholiya, S. Shrivastava, and V. Sharma, “An optimized XGBoost based diagnostic system for effective prediction of heart disease,” *J. King Saud Univ. - Comput. Inf. Sci.*, vol. 34, no. 7, pp. 4514–4523, 2020, doi: 10.1016/j.jksuci.2020.10.013.

[30] S. Kolassa, “Why the Mean Absolute Percentage Error Should Not Be Used for Forecast Accuracy Measurement,” *Foresight Int. J. Appl. Forecast.*, no. 61, pp. 24–33, 2020.

[31] M. Dorigo and L. M. Gambardella, “Ant Colony System: A Cooperative Learning Approach to the Traveling Salesman Problem,” *IEEE Trans. Evol. Comput.*, vol. 1, no. 1, pp. 53–66, 1997, doi: 10.1109/4235.585892.

[32] C. Bergmeir, R. J. Hyndman, and B. Koo, “A Note on the Validity of Cross-Validation for Evaluating Autoregressive Time Series Prediction,” *Comput. Stat. Data Anal.*, vol. 120, pp. 70–83, 2018, doi: 10.1016/j.csda.2017.11.003.

[33] R. J. Hyndman and A. B. Koehler, “Another Look at Measures of Forecast Accuracy,” *Int. J. Forecast.*, vol. 22, no. 4, pp. 679–688, 2006, doi: 10.1016/j.ijforecast.2006.03.001.