Evaluating Single and Hybrid Feature Selection for Rainfall Prediction Using XGBoost

1*Bambang Widoyono, 2Muhammad Fahmy Nadhif, 3Ridha Adjie Eryadi
1,2Faculty of Information Technology and Data Science, Universitas Sebelas Maret
3Department of Informatics, Digital Technology University
Email: 1bambangwidoyono@staff.uns.ac.id, 2fahmynadhif@staff.uns.ac.id, 3ridhaadjie@digitechuniversity.ac.id

Article Info

Article history:
Received Feb 05th, 2026
Revised Feb 25th, 2026
Accepted Mar 16th, 2026

Keyword:
Feature Selection
Hybrid Feature Selection
Machine Learning
Rainfall prediction
XGBoost

ABSTRACT

Rainfall prediction is challenging due to the complex and nonlinear nature of meteorological data. Previous studies using XGBoost with feature selection have demonstrated superior performance compared to other models, but evaluations have focused solely on error metrics (RSME, SME, MAE). Recent research suggests that predictive models should be evaluated for generalization, stability, interpretability, and computational efficiency to ensure their reliability. To close this gap, this study uses 8,750 hourly records obtained from Open-Meteo with 81 engineered features to evaluate XGBoost under three scenarios: without feature selection, single feature selection (MI, Boruta, SHAP, mRMR, ReliefF), and hybrid feature selection. Our findings demonstrate that accuracy is not always increased by feature selection. It does, however, increase interpretability, decrease overfitting, and improve computational efficiency. SHAP provides the most reliable performance among single methods, achieving lower RMSE (0.72632) and improved stability. Hybrid feature selection produces the most balanced performance gap = 0.01325, and stable variance = 0.03315 while reducing feature complexity to 35 variables. This study demonstrates the value of multidimensional evaluation beyond error metrics. In practical terms, this study suggests a feature selection method for rainfall prediction systems that are effective, reliable, and simple to understand.

Corresponding Author:
Bambang Widoyono,
Faculty of Information Technology and Data Science,
Universitas Sebelas Maret
Jl. Ir. Sutami No.36, Jebres, Kec. Jebres, Kota Surakarta, Jawa Tengah 57126.
Email: bambangwidoyono@staff.uns.ac.id

DOI: http://dx.doi.org/10.24014/ijaidm.v9i1.39110

1. INTRODUCTION

Rainfall prediction is so important for flood disaster mitigation and water resource management [1]. Several studies have shown that reliable rainfall prediction is important component in hydrological risk management and sustainable development planning [2], [3]. Conventional statistical approaches have limitations in modeling complex nonlinear relationships between variables for rainfall prediction [4], [5]. Therefore, machine learning approaches are used in rainfall prediction due to their ability to capture nonlinear patterns, handle high-dimensional data, and improve accuracy compared to traditional statistical methods [6], [7].

Various machine learning models have been applied to rainfall prediction, including artificial neural networks, support vector regression, random forests, and ensemble models [6], [7]. Previous studies have shown that ensemble models generally provide more stable performance than single models due to their ability to reduce bias [2], [8]. XGBoost is an ensemble model that produces lower prediction errors than other machine learning models in high-resolution rainfall prediction [9], [10]. Although XGBoost has good performance, machine learning models are highly dependent on the quality and relevance of the input features used [11]. A large number of features without proper selection can increase model complexity and cause overfitting, thereby reducing the model's generalization to previously unobserved data [12]. Therefore, feature selection is an important way to reduce data dimensionality, improve model stability, and retain only informative features [11], [13].

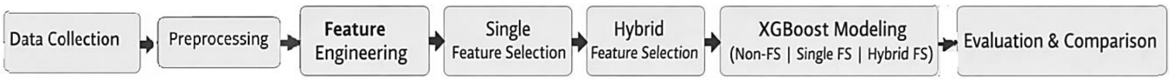
Several previous studies have applied single feature selection methods to improve the performance of rainfall prediction models and related environmental variables. Mutual Information approach is widely used to identify nonlinear relationships between variables and has been reported to improve accuracy on meteorological time series data [12]. Boruta method is also frequently applied in hydrological and environmental studies due to its ability to comprehensively identify relevant features [14]. Furthermore, SHAP-based methods are widely utilized because they not only function as feature selection techniques but also provide quantitative explanations of the contribution of each feature to the prediction model [13]. However, most previous studies have only evaluated a single feature selection method without systematic comparison with other feature selection methods or with conditions without feature selection [14]. Consequently, claims of improved model generalization and stability have not been supported by consistent comparative evaluations across feature selection methods. Previous studies reported RMSE improvements of approximately 38% using single feature selection yet rarely evaluated stability and generalization simultaneously [13]. Recent research has shown that a hybrid feature selection approach, which combines several feature selection methods, has the potential to produce a more robust and stable feature subset than a single approach [11], [15]. According to some studies, this approach is considered more suitable for multivariate and complex meteorological data.

Based on these gaps and problems, this study was conducted with the aim of comparing the performance of XGBoost model in three conditions: without feature selection, single feature selection, and hybrid feature selection for predicting rainfall, with the main focus on performance evaluation through error metrics, generalization gap, stability, and interpretability. Novelty of our research lies in authors' comparative analysis of various feature selection strategies with a broader range of evaluation metrics than previous research. Unlike previous research that emphasizes error metrics alone, this study integrates five evaluation dimensions within a unified comparative framework. Thus, it contributes to the literature on the role of feature selection in improving generalization and interpretability of predictive models. From a practical perspective, this research serves as a reference for practitioners in choosing the most effective feature selection strategies from a broader, multidimensional perspective.

Despite the growing number of studies applying feature selection to rainfall prediction, comparative evidence across multiple evaluation dimensions remains limited. Most prior work emphasizes accuracy improvement, while aspects such as stability, interpretability, and generalization consistency are rarely examined simultaneously. This gap motivates the present study.

2. RESEARCH METHOD

This study uses a quantitative experimental method approach, in the form of machine learning, to analyze the effect of various feature selections on rainfall prediction performance. Experimental approach was chosen because model performance evaluation can be controlled and can also be reused across various feature selection scenarios, so that differences in results can be directly attributed to the method used [1], [2], [3]. These research steps include meteorological data collection, data preprocessing, feature engineering, implementation of single and hybrid feature selection, predictive modeling using XGBoost, and model performance evaluation using error, generalization, stability, and interpretability metrics. Our evaluation approach aligns with recommendations in modern hydrometeorological prediction literature. Accuracy alone is insufficient to represent model quality for complex and noisy environmental data; therefore, model performance must also be evaluated in terms of generalization gaps, stability, and interpretability [4], [6], [16]. Figure 1 illustrates the research steps we conducted:



Data Collection → Preprocessing → Feature Engineering → Single Feature Selection → Hybrid Feature Selection → XGBoost Modeling (Non-FS | Single FS | Hybrid FS) → Evaluation & Comparison

Figure 1. Research Flow

2.1 Data Collection

This study uses 8750 raws data from Open-Meteo in 2025. Open-Meteo is open access, so we chosed, this provides relevant meteorological variables for hourly rainfall prediction and supports the reproducibility of data-driven hydrometeorological research. Open-Meteo reanalysis data and global numerical weather models has been widely recommended in previous literature as a stable and consistent data source for the development of machine learning models, especially for rainfall prediction with high temporal resolution [17], [18]. Using open data is also in line with modern research practices that emphasize transparency and repeatability of experiments [19]. We used meteorological variables including air temperature, relative humidity, wind speed, surface pressure, cloud cover, and solar radiation. The choice of these variables is consistent with previous studies showing that the combination of thermodynamic variables and atmospheric dynamics plays an important role in machine learning-based rainfall modeling [20], [21].

2.2 Preprocessing

Data preprocessing is performed to ensure data quality and consistency before use in feature selection and XGBoost modeling. This stage includes converting data to a numeric format, handling missing values, and adjusting time series data [22] . Conversion to a numeric format is necessary because regression-based machine learning algorithms can only process numeric data [22]. Invalid values are converted to NaN and then removed after all derived features are generated. This is done to avoid bias due to incorrect imputation of high-resolution data [22], [23]. Previous research has shown that data preprocessing errors significantly impact machine learning performance in predicting rainfall [24], [25].

2.3 Feature Engineering

After feature construction, the next step involved identifying which predictors meaningfully contributed to rainfall estimation. Feature engineering is needed to enrich the representation of meteorological data to capture temporal dependencies and atmospheric variability that affect rainfall[26]. In practical terms, this stage focused on constructing lag-based predictors alongside rolling statistical summaries. Lag features were generated from several previous time steps so that short- to medium-range temporal dependencies could be explicitly captured in the learning process, an approach that has repeatedly shown benefits in rainfall modeling studies. This approach is used because it has been proven effective in improving performance of machine learning models in rainfall prediction [26], [27]. Meanwhile, rolling statistical features such as minimum, average, and maximum values within a certain time range are used to capture local fluctuations and trends. Combination of lag features and rolling statistical features has been widely used in previous studies and has been shown to improve the model's ability to represent complex rainfall dynamics, so it is necessary to apply it in this study [28], [29].

2.4 Feature Selection

The next step is featuring selection. The goal of this step is not only limited to dimensionality reduction but also eliminating redundant information and improving model generalization and interpretability. Relevant feature selection helps control model complexity, supports stable learning behavior, and reduces the risk of overfitting, as reported in previous studies [7], [8], [9], [10]. Feature selection methods used in this study include Mutual Information (MI), Boruta, SHAP, mRMR, ReliefF, and hybrid feature selection

The use of Mutual Information feature selection aims to measure the statistical dependence between each feature and the target variable without assuming linearity [7], [10]. Features with high MI scores contain greater information about the prediction target and therefore can create a high chance of contributing to improving model performance [7], [10], [30]. The second method, Boruta, is a wrapper-based technique built on Random Forest that aims to retain all relevant predictors rather than just a minimal subset [13], [14]. This method evaluates feature importance by comparing the original variable with randomized shadow features using statistical testing. A predictor is considered relevant when its importance score significantly exceeds the score of its corresponding shadow feature [8], [12], [13]. SHAP is derived from cooperative game theory and is used to estimate the marginal contribution of each feature to model output [31], [32]. By averaging contributions across all possible feature combinations, SHAP provides consistent and interpretable importance values that reflect each predictor’s influence on the final prediction [31], [32]. mRMR method aims to select features with maximum relevance to the target and minimum redundancy between features [33], with the objective function maximizing the difference between the average relevance of features to the target and the average redundancy between features. This approach ensures that the selected features have a high information contribution to the target while minimizing overlapping information between features. ReliefF evaluates features based on their ability to discriminate between adjacent samples by updating feature weights based on the difference in values between nearest neighbors of the same and different classes [13], [6]. In ReliefF function measures the difference in feature values between two samples. Features with higher values are considered to have higher discriminatory power. The mathematical formulations of each standard feature selection their original definitions and are therefore not re-derived in this study.

Hybrid feature selection is designed to combine the advantages of several single feature selection methods to obtain a more robust feature subset that is independent of the bias of a single method [34], [35],

[36]. Mathematically, suppose there is a set of feature selection methods $M = \{m_1, m_2, \dots, m_k\}$, where each method produces a score or ranking $r_{ij}$for feature-*i*. Hybrid score of feature-*i* is formulated as equation 1.

$$H_i = \frac{1}{k} \sum_{j=1}^{k} r_{ij} \tag{1}$$

H_i represents the aggregate hybrid score of feature-*i*, r_{ij} is the score or ranking of feature-*i* from method-*j*, and k is the number of feature selection methods [34]. The feature with the best H_i value is selected as the final feature subset. In this study, a hybrid feature was constructed by aggregating feature selection results from Mutual Information, XGBoost-based feature importance, SHAP, Boruta, mRMR, and ReliefF. This approach ensures that the selected features have relevance support from more than one selection perspective, thereby increasing the stability and generalization of the model.

2.5 XGBoost Modeling

Machine learning allows models to learn complex patterns from data to make predictions without explicit rules, and excels at modeling nonlinear relationships in meteorological data [4], [6]. For model prediction in this study, we decided to use XGBoost based on its performance and comparison with ensemble models [9], [10]. XGBoost is a decision tree-based gradient boosting algorithm that optimizes the objective function [15], [21], [37]. Referring to Chen and Guestrin (2016), the formula is as equation 2.

$$\mathcal{L} = \sum_{i=1}^{n} \ell(y_i, \hat{y}_i) + \sum_{k=1}^{K} \Omega(f_k) \tag{2}$$

where $\ell(y_i, \hat{y}_i)$ is the loss function between actual values and predicted values, f_k is decision tree-*k*, and $\Omega(f_k)$ is a regularization function that controls model complexity. The regularization function $\Omega(f_k)$ is formulated as equation 3.

$$\Omega(f_k) = \gamma T_k + \frac{1}{2} \lambda \sum w_{kj}^2 \tag{3}$$

where T_k represents the number of leaves in tree-*k* and w_{kj} the leaf weight, while γ and λ are regularization parameters [15], [37]. The dataset was evaluated using 5-fold cross-validation. XGBoost parameters were configured as learning_rate=0.1, max_depth=6, n_estimators=100, subsample=0.8, and colsample_bytree=0.8.

2.6 Evaluation Model

In the model performance evaluation stage in this study, we not only measured it with prediction error metrics using MSE, RMSE, and MAE, but also generalization ability, performance stability, and model interpretability. This multifaceted evaluation approach is recommended in previous studies on machine learning for environmental and hydrometeorological data, because with this approach, the model can be improved in stability, generalization, and ease of interpretation [38], [39]

Mean Absolute Error (MAE) is used to measure the average absolute error between the predicted value and the actual value. MAE provides a direct indication of the magnitude of the prediction error in the same units as the target variable, making it easy to interpret physically in the context of rainfall [40], [41], [42]. The MAE formula is defined as equation 4.

$$MAE = \frac{1}{n} \sum_{i=1}^{n} | y_i - \hat{y}_i | \tag{4}$$

In MAE formula, n represents the number of observation samples, y_i is the actual rainfall value at the observation-*i*, and $\hat{y}_i$ is the model's predicted value for the same observation. A smaller MAE value indicates that, on average, the model's prediction is closer to the actual value.

The Mean Squared Error (MSE) measures the average squared difference between the predicted and actual values, thus imposing a greater penalty on extreme prediction errors [39], [40]. The MSE is formulated as equation 5.

$$MSE = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \tag{5}$$

In MSE formula above, n is the number of samples, y_i represents the actual value, and $\hat{y}_i$ is the model's predicted value. Because the errors are squared, MSE is sensitive to outliers and is particularly useful for evaluating models on rainfall data with asymmetric distributions and extreme events.

Root Mean Squared Error (RMSE) is the square root of MSE and is used to scale the errors back to the original units of the target variable, making it easier to interpret than MSE [41], [42]. RMSE is defined as equation 6.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{6}$$

In this RMSE equation, n represents the number of samples, y_i is the actual value, and $\hat{y}_i$ is the predicted value. RMSE is widely used as a primary metric in hydrometeorological prediction because it represents the overall magnitude of prediction error while being sensitive to large errors.

Generalization Gap evaluation is used to determine the model's generalization ability, in this study the XGBoost model, the extent to which the model's performance on the test data is consistent with its performance on the training data. This difference in performance is expressed as the generalization gap, which is used as an indicator of overfitting or underfitting [43], [44], [45]. In this study, the generalization gap was calculated using the RMSE metric as equation 7.

$$Gap_{RMSE} = |RMSE_{test} - RMSE_{train} | \tag{7}$$

In this formula, $RMSE_{test}$ is the RMSE value on the test data, while $RMSE_{train}$ is the RMSE value on the training data. A small generalization gap value indicates that the model has good generalization ability and does not experience significant performance differences between the training and test data.

Model stability evaluation represents the consistency of model performance across variations in the training and validation data. In the context of cross-validation, stability is often measured using the variance of the inter-fold performance metric values [46], [47]. In this study, stability is calculated using the RMSE variance as equation 8.

$$Stability = \frac{1}{K-1}\sum_{k=1}^{K}(RMS\,E_k - R\bar{M}SE)^2 \tag{8}$$

where K represents the number of cross-validation folds, $RMSE_k$ is the RMSE value for fold-*k*, and $R\bar{M}SE$ is the average RMSE value across all folds. Model stability was defined as the variance of RMSE across folds, where lower variance indicates more consistent learning behavior. A smaller stability value indicates a more consistent model performance and is less sensitive to changes in data partitioning.

Evaluating model interpretability in the context of feature selection relates to the extent to which each feature's contribution to the prediction can be understood. Models with fewer features are generally easier to interpret, especially in scientific applications such as meteorology, where understanding the role of atmospheric variables has high practical and physical value [7][48]. In this study, interpretability was measured quantitatively using the inverse ratio of the number of features used by the model as equation 9.

$$Interpretability = \frac{1}{|F|} \tag{9}$$

In interpretability equation, |F| represents the number of features used by the model after the feature selection process. A larger interpretability value indicates that the model uses fewer features, making the contribution of each feature to the prediction easier to analyze and explain.

3 RESULTS AND ANALYSIS

3.1 XGBoost Model without Feature Selection

Table 1. Model Performance without Feature Selection

Fold	Runtime	MAE		MSE		RMSE		R2		RMSE			Feature
		Train	Test	Train	Test	Train	Test	Train	Test	Generalization	Stability	Interpretability	
1	1.91779	0.31439	0.31853	0.57995	0.41579	0.76154	0.64482	0.26675	0.10197	0.11673	0.03348	0.01235	81
2	1.91779	0.26733	0.26750	0.46443	0.62819	0.68149	0.79258	0.23858	0.14213	0.11109	0.03348	0.01235	81
3	1.91779	0.26909	0.17685	0.50604	0.27403	0.71136	0.52348	0.23807	0.25023	0.18788	0.03348	0.01235	81
4	1.91779	0.24023	0.28736	0.45131	1.02133	0.67180	1.01061	0.23784	0.17307	0.33881	0.03348	0.01235	81
5	1.91779	0.25051	0.28764	0.53551	0.47970	0.73179	0.69260	0.25748	0.17623	0.03918	0.03348	0.01235	81

Based on the test results without feature selection on five folds in Table 1, a consistent runtime value of 1.91779 seconds was obtained with a fixed number of features of 81, indicating a relatively stable computational load. Prediction performance shows that the MAE of the test data ranges from 0.17685 to 0.31853, while the MSE of the test is in the range of 0.27403–1.02133 and produces a test RMSE of

0.52348–1.01061, indicating a still quite high variation in prediction errors between folds. The R² value of the test is also relatively low, between 0.10197 and 0.25023, so the model's ability to explain the variance of the target data is still limited. From the perspective of model quality, stability and interpretability remain constant at 0.03348 and 0.01235, respectively, while the generalization metric shows quite wide fluctuations (0.03918–0.33881), these results confirm that the use of all features without selection does not have generalization ability.

3.2 Mutual Information (MI)

Table 2. Model Performance with MI

Fold	Runtime	MAE		MSE		RMSE		R2		RMSE			Feature
		Train	Test	Train	Test	Train	Test	Train	Test	Generalization	Stability	Interpretability	
1	1.06102	0.32319	0.29283	0.63636	0.42377	0.79772	0.65097	0.19543	0.08474	0.14675	0.03362	0.05000	20
2	1.06102	0.27524	0.28095	0.50988	0.66243	0.71406	0.81390	0.16406	0.09537	0.09983	0.03362	0.05000	20
3	1.06102	0.28045	0.18365	0.56356	0.30617	0.75071	0.55333	0.15145	0.16230	0.19738	0.03362	0.05000	20
4	1.06102	0.25077	0.30709	0.49684	1.07025	0.70487	1.03453	0.16095	0.13346	0.32966	0.03362	0.05000	20
5	1.06102	0.26336	0.30592	0.59272	0.51498	0.76988	0.71762	0.17816	0.11565	0.05226	0.03362	0.05000	20

Based from Table 2, Mutual Information (MI) feature selection with 20 selected features, a lower and consistent runtime of 1.06102 seconds was obtained, indicating computational efficiency compared to without feature selection. In terms of accuracy, the MAE value of the test data is in the range of 0.18365–0.30709, the MSE test is 0.30617–1.07025, and the RMSE test is 0.55333–1.03453, which indicates that the prediction error still fluctuates but tends to be moderate. The R² value of the test ranges from 0.08474–0.16230, indicating a relatively limited variance explanation ability although more stable than using all features. From a model quality perspective, generalization is in the range of 0.05226–0.32966 with a constant stability of 0.03362 and interpretability increasing to 0.05000, indicating that feature reduction has succeeded in improving model readability without sacrificing stability. Overall, the MI approach provides advantages in runtime efficiency and model interpretability but still has weaknesses in terms of less-than-optimal levels of accuracy and explainability.

3.3 Boruta

Table 3. Model Performance with Boruta

Fold	Runtime	MAE		MSE		RMSE		R2		RMSE			Feature
		Train	Test	Train	Test	Train	Test	Train	Test	Generalization	Stability	Interpretability	
1	0.60838	0.34695	0.27946	0.70184	0.41682	0.83776	0.64562	0.11265	0.09974	0.19214	0.03458	0.50000	2
2	0.60838	0.29026	0.28422	0.54271	0.66761	0.73669	0.81707	0.11024	0.08829	0.08039	0.03458	0.50000	2
3	0.60838	0.29354	0.17874	0.58871	0.30149	0.76727	0.54908	0.11359	0.17511	0.21819	0.03458	0.50000	2
4	0.60838	0.26163	0.29669	0.51519	1.07477	0.71777	1.03671	0.12996	0.12981	0.31894	0.03458	0.50000	2
5	0.60838	0.27547	0.30427	0.61562	0.53011	0.78461	0.72809	0.14641	0.08966	0.05653	0.03458	0.50000	2

Based Table 3, on the test results using Boruta feature selection with only 2 selected features, the lowest and most stable runtime of 0.60838 seconds was obtained, indicating the highest computational efficiency compared to previous methods. In terms of accuracy, the MAE value of the test data was in the range of 0.17874–0.30427, the MSE test was 0.30149–1.07477, and the RMSE test was 0.54908–1.03671, indicating that the prediction error was still fluctuating and relatively high in some folds. The R² test value ranged from 0.08829–0.17511, indicating that the model's ability to explain target variance was still limited despite an increase in certain folds. From a model quality perspective, generalization was in the range of 0.05653–0.31894 with a constant stability of 0.03458 and a very high interpretability of 0.50000, confirming that extreme feature reduction significantly improved model explainability.

3.4 SHAP

In Table 4 is a results using SHAP feature selection with 20 selected features obtained a relatively stable runtime of 1.14144 seconds, indicating better computational efficiency compared to no feature selection but slightly higher than MI and Boruta. In terms of accuracy, the MAE value of the test data is in the range of 0.17855–0.29942, the MSE test is 0.27050–0.99642, and the RMSE test is 0.52009–0.99821, which indicates that the prediction error tends to be lower and consistent compared to other feature selection methods. The R² test value ranges from 0.12162–0.25992, indicating a relatively better and stable variance explanation ability. From the perspective of model quality, generalization is in the range of 0.03822–0.32513 with a constant stability of 0.03213 and interpretability of 0.05000, indicating a balance between model readability and performance stability.

Table 4. Model Performance with SHAP

Fold	Runtime	MAE		MSE		RMSE		R2		RMSE			Feature
		Train	Test	Train	Test	Train	Test	Train	Test	Generalization	Stability	Interpretability	
1	1.14144	0.31376	0.29942	0.58877	0.40669	0.76731	0.63772	0.25561	0.12162	0.12959	0.03213	0.05000	20
2	1.14144	0.26501	0.26247	0.46488	0.61197	0.68182	0.78228	0.23783	0.16428	0.10046	0.03213	0.05000	20
3	1.14144	0.26828	0.17855	0.50682	0.27050	0.71191	0.52009	0.23689	0.25992	0.19182	0.03213	0.05000	20
4	1.14144	0.23967	0.28097	0.45303	0.99642	0.67308	0.99821	0.23492	0.19324	0.32513	0.03213	0.05000	20
5	1.14144	0.25060	0.27662	0.53513	0.48068	0.73152	0.69331	0.25801	0.17455	0.03822	0.03213	0.05000	20

3.5 mRMR

Table 5. Model Performance with mRMR

Fold	Runtime	MAE		MSE		RMSE		R2		RMSE			Feature
		Train	Test	Train	Test	Train	Test	Train	Test	Generalization	Stability	Interpretability	
1	0.30004	0.32353	0.28994	0.63541	0.42652	0.79712	0.65308	0.19664	0.07880	0.14404	0.03286	0.05000	20
2	0.30004	0.27488	0.30298	0.51002	0.67171	0.71415	0.81958	0.16384	0.08269	0.10543	0.03286	0.05000	20
3	0.30004	0.28085	0.18683	0.56462	0.30365	0.75141	0.55104	0.14986	0.16921	0.20037	0.03286	0.05000	20
4	0.30004	0.25060	0.29313	0.49660	1.05411	0.70470	1.02670	0.16136	0.14653	0.32200	0.03286	0.05000	20
5	0.30004	0.26365	0.30498	0.59203	0.51334	0.76943	0.71648	0.17911	0.11846	0.05296	0.03286	0.05000	20

Based Table 5, mRMR feature selection with 20 selected features obtained the fastest and most stable runtime of 0.30004 seconds, indicating the highest computational efficiency compared to all other feature selection methods. In terms of accuracy, the MAE value of the test data ranged from 0.18683 to 0.30498, the MSE of the test data was 0.30365 to 1.05411, and the RMSE of the test data was 0.55104 to 1.02670, so the prediction error was moderate but still fluctuated between folds. The R² value of the test ranged from 0.07880 to 0.16921, indicating that the model's ability to explain target variance was still relatively limited and had not shown significant improvement compared to other methods. From a model quality perspective, generalization ranged from 0.05296 to 0.32200 with a constant stability of 0.03286 and interpretability of 0.05000, indicating a balance between model stability and readability even without a strong increase in accuracy.

3.6 ReliefF

Table 6. Model Performance with ReliefF

Fold	Runtime	MAE		MSE		RMSE		R2		RMSE			Feature
		Train	Test	Train	Test	Train	Test	Train	Test	Generalization	Stability	Interpretability	
1	1.06212	0.36059	0.31149	0.65727	0.42576	0.81072	0.65250	0.16900	0.08043	0.15822	0.03246	0.05000	20
2	1.06212	0.30190	0.29437	0.50697	0.66401	0.71202	0.81487	0.16883	0.09321	0.10285	0.03246	0.05000	20
3	1.06212	0.30643	0.18471	0.55886	0.31083	0.74757	0.55752	0.15853	0.14955	0.19005	0.03246	0.05000	20
4	1.06212	0.27420	0.31276	0.49421	1.05945	0.70300	1.02930	0.16539	0.14220	0.32630	0.03246	0.05000	20
5	1.06212	0.28695	0.32533	0.58332	0.51243	0.76375	0.71585	0.19119	0.12002	0.04791	0.03246	0.05000	20

Based Table 6, results using ReliefF feature selection with 20 selected features obtained a stable runtime of 1.06212 seconds, indicating better computational efficiency compared to without feature selection and comparable to the MI method. In terms of accuracy, the MAE value of the test data is in the range of 0.18471–0.32533, the MSE test is 0.31083–1.05945, and the RMSE test is 0.55752–1.02930, which indicates that the prediction error is still moderate with quite visible fluctuations between folds. The R² test value ranges from 0.08043–0.14955, indicating that the model's ability to explain target variance is still relatively low and has not provided significant improvements. From the perspective of model quality, generalization is in the range of 0.04791–0.32630 with a constant stability of 0.03246 and interpretability of 0.05000, indicating the stability of the model structure and good readability although not accompanied by an increase in accuracy.

3.7 Hybrid Feature Selection

Based hybrid scenario in Table 7, with 35 selected features obtained a relatively stable runtime of 1.22233 seconds, indicating better computational efficiency compared to no feature selection, although not as fast as mRMR or Boruta. In terms of accuracy, the MAE value of the test data is in the range of 0.17610–0.29938, the MSE test is 0.27704–1.01519, and the RMSE test is 0.52635–1.00757, which indicates a moderate prediction error with visible inter-fold variation. The R² value of the test ranges from 0.12724–0.24200, indicating a relatively better variance explanation ability compared to several single feature selection methods, although it has not yet reached a high level. From the perspective of model quality, generalization is in the range of 0.04546–0.33268 with a constant stability of 0.03315 and interpretability of 0.02857, indicating a compromise between model stability and a level of interpretability that is not as high as Boruta.

Table 7. Model Performance with Hybrid Feature Selection

Fold	Runtime	MAE		MSE		RMSE		R2		RMSE			Feature
		Train	Test	Train	Test	Train	Test	Train	Test	Generalization	Stability	Interpretability	
1	1.22233	0.31337	0.29938	0.58542	0.40409	0.76513	0.63568	0.25984	0.12724	0.12945	0.03315	0.02857	35
2	1.22233	0.26887	0.27078	0.47230	0.62642	0.68724	0.79147	0.22567	0.14454	0.10423	0.03315	0.02857	35
3	1.22233	0.27169	0.17610	0.52141	0.27704	0.72209	0.52635	0.21492	0.24200	0.19574	0.03315	0.02857	35
4	1.22233	0.24119	0.28566	0.45547	1.01519	0.67488	1.00757	0.23081	0.17805	0.33268	0.03315	0.02857	35
5	1.22233	0.25307	0.28704	0.54475	0.47971	0.73807	0.69261	0.24467	0.17622	0.04546	0.03315	0.02857	35

3.8 Comparation Performance

Table 8. Comparison Model Performance

Model	Run Time	MAE		MSE		RMSE		R2		RMSE			Feature
		Train	Tes	Train	Tes	Train	Tes	Train	Tes	Generalization	Stability	Interpretability	
Base	1.91779	0.26831	0.26758	0.50745	0.56381	0.71160	0.73282	0.24774	0.16873	0.02122	0.03348	0.01235	81
MI	1.06102	0.27860	0.27409	0.55987	0.59552	0.74745	0.75407	0.17001	0.11831	0.00662	0.03362	0.05000	20
BORUTA	0.60838	0.29357	0.26868	0.59281	0.59816	0.76882	0.75531	0.12257	0.11652	0.01351	0.03458	0.50000	2
SHAP	1.14144	0.26746	0.25961	0.50973	0.55325	0.71313	0.72632	0.24465	0.18272	0.01319	0.03213	0.05000	20
mRMR	0.30004	0.27870	0.27557	0.55973	0.59387	0.74736	0.75338	0.17016	0.11914	0.00601	0.03286	0.05000	20
ReliefF	1.06212	0.30601	0.28573	0.56013	0.59450	0.74741	0.75401	0.17059	0.11708	0.00659	0.03246	0.05000	20
Hybrid	1.22233	0.26964	0.26379	0.51587	0.56049	0.71748	0.73073	0.23518	0.17361	0.01325	0.03315	0.02857	35

Table 8 summarizes and shows a comparison of model performance across all feature selection scenarios. Model without feature selection exhibits the highest runtime at 1.91779 seconds, with test MAE = 0.26758, test MSE = 0.56381, test RMSE = 0.73282, and test R^2 = 0.16873. These results indicate that the model has a better prediction error than Boruta, mRMR, and ReliefF despite using 81 features, but is the worst in terms of computational efficiency. In addition, the stability value = 0.03348 and generalization ability = 0.02122 are the largest compared to feature selection. When using single feature selection methods such as MI, ReliefF, mRMR, SHAP, and Boruta, the computation time can be significantly reduced to around 0.30004–1.14144 seconds because the number of features used is much smaller, namely around 2–20 features. However, the testing RMSE value remains in the range of 0.726–0.755 and the R^2 value is still relatively low, namely 0.11652–0.18272. This indicates that feature selection does not always reduce the RMSE value. Only SHAP can reduce computation and RMSE. The main advantage of single feature selection lies in increasing model interpretability. SHAP analysis identified relative humidity, cloud cover, and surface pressure as dominant predictors, which physically relate to condensation, atmospheric instability, and moisture transport processes influencing rainfall formation. The interpretability value increased significantly from 0.01235 in the baseline model to 0.50000 in the Boruta method. Meanwhile, the hybrid feature selection model provided a better balance between reducing the RMSE value, computational efficiency, and model generalization. The hybrid feature selection model produced a runtime of 1.22233 seconds, an MAE of 0.26379, an MSE of 0.56049, an RMSE of 0.73073, and an R^2 of 0.17361. In terms of error metrics, the hybrid model performed better than the non-feature selection model. However, its runtime was still lower than that of single feature selection because the hybrid model used 35 features. The accompanying generalization=0.01325 and stability=0.03315 suggest consistent learning behavior without clear evidence of overfitting, and interpretability =0.02857 improves upon the no-selection baseline even if it does not reach Boruta's simplicity. Consequently, the absence of feature selection is chiefly limited by computational cost and weak interpretability, single feature selection excels in efficiency and transparency but with uneven predictive gains, and the hybrid strategy offers the most coherent balance among predictive performance, robustness, and feature complexity within the reported empirical results. The relatively low R^2 values (0.116–0.182) reflect high stochastic rainfall variability and nonlinear atmospheric dynamics.

3.9 Discussion

Based on the performance comparison graph on Figure 2, a clear trend is visible that Single Feature Selection (FS) and Hybrid FS provide different but complementary model characteristic improvements compared to No FS. Specifically, the Single FS approach shows the most consistent advantages in the generalization gap and interpretability aspects, where almost all single FS methods produce a smaller generalization gap (≈0.006–0.013) compared to No FS (≈0.021), as well as significantly increased interpretability, especially Boruta which reaches 0.50 due to the very minimal number of features. This trend confirms that single FS is effective as an implicit regularization mechanism that reduces feature redundancy and noise, thereby increasing model robustness and transparency, as also reported in recent hydrological and environmental studies that show that single feature selection can improve generalization without significantly increasing model complexity [49]. Meanwhile, Hybrid FS shows a major advantage in global performance balance, indicated by a consistently low RMSE (≈0.731), a relatively high R^2 (≈0.174), and maintained stability (≈0.033) with moderate runtime, which visually appears more balanced compared to the performance fluctuations between single FS. Hybrid selection improves stability because only features consistently ranked high across multiple selection mechanisms are retained, reducing method-specific bias. This pattern indicates that hybrid FS is able to combine the advantages of filter and model-based approaches, thus preserving key predictive features while controlling complexity, in line with recent findings that hybrid feature selection excels in handling high-dimensional data with an optimal trade-off between accuracy, generalization, and interpretability [50]. Thus, this graph strongly supports that Single FS excels in interpretability and generalization control, while Hybrid FS excels in consistency and multi-metric balance, while No FS lags behind primarily in generalization and interpretability, though not necessarily in absolute error.

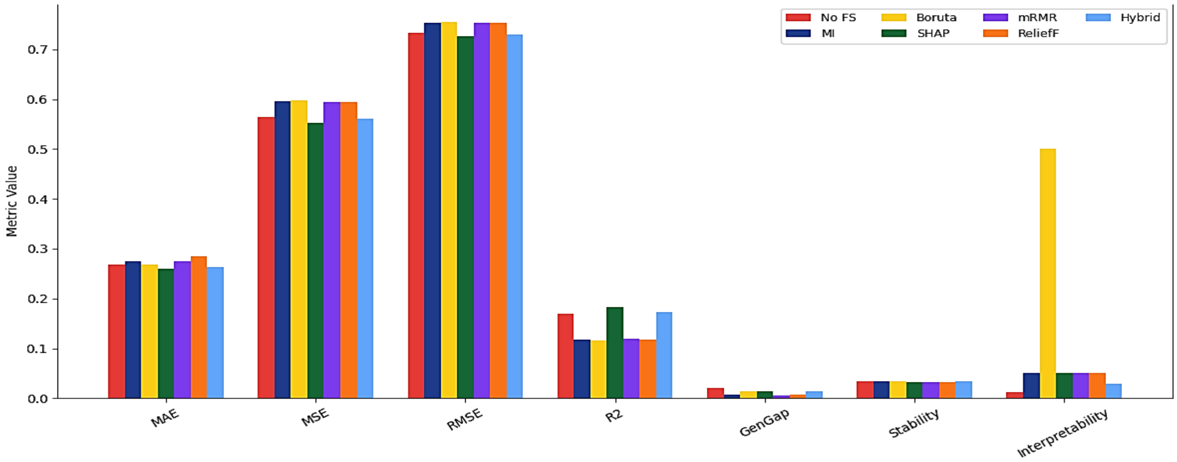


Figure 2. Bar Comparison Performance

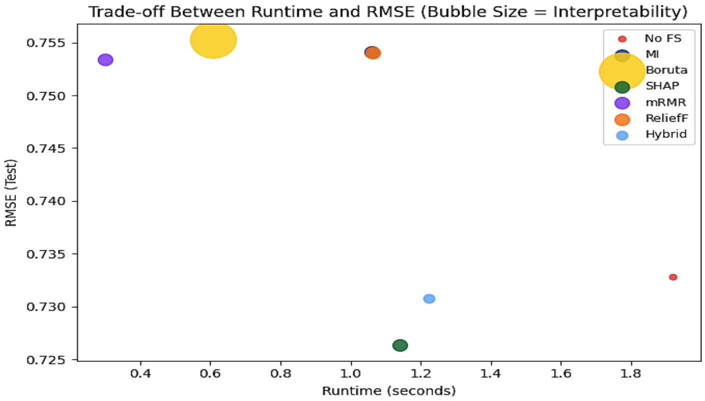


Figure 3. Trade-off Between Runtime and RMSE

Figure 3 is a trade-off graph between runtime and RMSE with bubble size as interpretability explicitly shows that the difference in computational time is primarily determined by the number of effective features and the complexity of the selection mechanism, rather than solely by the XGBoost algorithm itself. The No FS model has the highest runtime (≈1.92 seconds) because all 81 features are processed without reduction, thus increasing the cost of split searches on each tree and increasing the complexity of gradient optimization, although its RMSE (≈0.733) is not proportionally lower. Using all 81 features increases model complexity and risk of overfitting, as reflected by the larger generalization gap (0.02122) compared to feature-selected models. In contrast, Single FS displays significantly more efficient runtimes, particularly mRMR (≈0.30 seconds) and Boruta (≈0.61 seconds), which is directly due to the aggressiveness of feature reduction (down to only 2–20 features), which narrows the model's search space and speeds up the boosting process; this phenomenon is consistent with recent literature showing that reducing input dimensionality significantly reduces the computational complexity of tree-based ensembles without drastically increasing prediction errors. Because the model with Hybrid feature selection uses more features than single feature selection, its execution time is slower, approximately 1.22 seconds. Computational efficiency of this approach is still higher than that of model without feature selection and single feature selection. This pattern demonstrates that the hybrid FS's advantage stems from layered selection, which strikes a balance between efficiency and better prediction error, whereas the single FS's execution time advantage stems from dimensionality reduction.

4 CONCLUSION

This study successfully achieved its objective of comparing single and hybrid feature selection across five evaluation dimensions and identifying hybrid selection as the most balanced approach. Our research used the XGBoost model to predict rainfall in three different datasets: without feature selection, with single feature selection, and with hybrid feature selection. Our results showed that single feature selection with SHAP was the most reliable of all single feature selection methods. Other single feature selection methods, such as Boruta, MI, ReliefF, and mRMR, improved stability and interpretability, but yielded lower error metrics compared to those without feature selection. This suggests that using feature selection does not automatically improve error and accuracy. However, feature selection clearly contributes in other important ways: faster computation due to fewer features, addressing the gap between training and testing values, and reducing overfitting. Meanwhile, the hybrid method produced MAE values of 0.26964 for training and 0.26379 for testing, with RMSE values for training and testing of 0.71748 and 0.73073, respectively. Meanwhile, the generalization gap remained low at 0.01325, and performance across all folds remained stable at 0.03315, with an interpretation of 0.02857. These results indicate that hybrid feature selection significantly reduced the error metric compared to non-feature selection, but still outperformed single-feature selection in terms of running time, which was 1.22 seconds, because hybrid feature selection used more features, namely 35. Overall, this study successfully demonstrated that hybrid feature selection is more effective for handling meteorological features than relying on one type of feature selection alone. A limitation of this study is the use of only one machine learning model. Future research should compare XGBoost with LSTM, Random Forest, and LightGBM, validate across multi-regional datasets, and apply advanced explainable AI techniques for deeper meteorological interpretation

ACKNOWLEDGEMENTS

We would like to say thank to Enterprise Architecture and System Research Group at Sebelas Maret University for the support and facilities provided during this research. They also thank the reviewers for their constructive feedback that helped improve the quality of this manuscript.

REFERENCES

[1] A. Mosavi, P. Ozturk, and K. W. Chau, "Flood prediction using machine learning models: Literature review," *Water (Switzerland)*, vol. 10, no. 11, p. 1536, 2018, doi: 10.3390/w10111536.

[2] B. T. Pham and others, "A comparative study of rainfall prediction using statistical and machine learning methods," *Theor. Appl. Climatol.*, vol. 142, no. 3--4, pp. 885–900, 2020, doi: 10.1007/s00704-020-03247-3.

[3] P. Abbaszadeh, H. Moradkhani, and H. Yan, "Enhancing hydrologic prediction with machine learning," *Hydrol. Earth Syst. Sci.*, vol. 23, no. 8, pp. 3621–3635, 2019, doi: 10.5194/hess-23-3621-2019.

[4] S. Kundu, S. K. Biswas, D. Tripathi, S. Mandal, and S. Majumdar, "Rainfall forecasting using ensemble learning: A review," *e-Prime*, vol. 6, p. 100296, 2023, doi: 10.1016/j.prime.2023.100296.

[5] Y. Liu, W. Zhang, X. Yang, and Y. Sun, "Short-term and hourly rainfall forecasting using machine learning approaches," *Atmos. Res.*, vol. 276, p. 106244, 2022, doi: 10.1016/j.atmosres.2022.106244.

[6] J. Li *et al.*, "Feature selection: A data perspective," *ACM Comput. Surv.*, vol. 50, no. 6, p. 94, 2018, doi: 10.1145/3136625.

[7] I. Iguyon and A. Elisseeff, "An introduction to variable and feature selection," *J. Mach. Learn. Res.*, vol. 3, pp. 1157–1182, 2003.

[8] Y. Saeys, T. Abeel, and Y. de Peer, "Robust feature selection using ensemble feature selection techniques," *Bioinformatics*, vol. 24, no. 13, pp. i555--i563, 2008, doi: 10.1093/bioinformatics/btn630.

[9] Y. Zhang, X. Li, L. Wang, and H. Zhang, "Ensemble feature selection methods for environmental applications," *Remote Sens.*, vol. 15, no. 4, p. 1096, 2023, doi: 10.3390/rs15041096.

[10] L. Huang, X. Zhou, L. Shi, and L. Gong, "Time series feature selection based on mutual information," *Appl. Sci.*, vol. 14, no. 5, p. 1960, 2024, doi: 10.3390/app14051960.

[11] H. Peng, F. Long, and C. Ding, "Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 8, pp. 1226–1238, 2005, doi: 10.1109/TPAMI.2005.159.

[12] M. B. Kursa and W. R. Rudnicki, "Feature selection with the Boruta package," *J. Stat. Softw.*, vol. 36, no. 11, pp. 1–13, 2010, doi: 10.18637/jss.v036.i11.

[13] M. Robnik-Šikonja and I. Kononenko, "Theoretical and empirical analysis of ReliefF," *Mach. Learn.*, vol. 53, no. 1--2, pp. 23–69, 2003.

[14] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in *Advances in Neural Information Processing Systems*, 2017. doi: 10.48550/arXiv.1705.07874.

[15] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016, pp. 785–794. doi:

10.1145/2939672.2939785.

[16] Y. Liu, W. Zhang, X. Yang, and Y. Sun, "Hourly rainfall forecasting using machine learning approaches," *Atmos. Res.*, vol. 276, p. 106244, 2022, doi: 10.1016/j.atmosres.2022.106244.

[17] H. Hersbach and others, "The ERA5 global reanalysis," *Q. J. R. Meteorol. Soc.*, vol. 146, no. 730, pp. 1999–2049, 2020, doi: 10.1002/qj.3803.

[18] J. Muñoz-Sabater and others, "ERA5-Land: A state-of-the-art global reanalysis dataset," *Earth Syst. Sci. Data*, vol. 13, no. 9, pp. 4349–4383, 2021, doi: 10.5194/essd-13-4349-2021.

[19] D. N. Karger and others, "Climatologies at high resolution for the Earth's land surface areas," *Sci. Data*, vol. 4, p. 170122, 2017, doi: 10.1038/sdata.2017.122.

[20] A. Bárdossy and G. Pegram, "Space--time conditional disaggregation of precipitation," *Water Resour. Res.*, vol. 52, no. 7, pp. 5216–5232, 2016, doi: 10.1002/2015WR018129.

[21] Q. Yang, H. Wang, and Y. Chen, "Hourly precipitation forecasting using machine learning," *Atmos. Res.*, vol. 252, p. 105432, 2021, doi: 10.1016/j.atmosres.2021.105432.

[22] R. J. Hyndman and G. Athanasopoulos, "Forecasting: principles and practice," *Int. J. Forecast.*, vol. 34, no. 4, pp. 693–694, 2018, doi: 10.1016/j.ijforecast.2018.05.001.

[23] I. Lauriola, A. Lavelli, and F. Aiolli, "An introduction to deep learning in natural language processing," *Found. Trends Inf. Retr.*, vol. 14, no. 1, pp. 1–107, 2020, doi: 10.1561/1500000064.

[24] R. Taormina, K. W. Chau, and R. Sethi, "Artificial neural network simulation of hourly groundwater levels," *J. Hydrol.*, vol. 531, pp. 102–115, 2015, doi: 10.1016/j.jhydrol.2015.10.037.

[25] F. Kratzert, D. Klotz, C. Brenner, K. Schulz, and M. Herrnegger, "Rainfall--runoff modelling using LSTM networks," *Hydrol. Earth Syst. Sci.*, vol. 23, no. 12, pp. 5089–5110, 2019, doi: 10.5194/hess-23-5089-2019.

[26] X. H. Le, H. Van Ho, G.-H. Lee, and S. Jung, "Application of Long Short-Term Memory Neural Network for Rainfall Forecasting," *Water*, vol. 11, no. 7, p. 1387, 2019, doi: 10.3390/w11071387.

[27] S. Poornima and M. Pushpalatha, "Prediction of rainfall using machine learning techniques," *Int. J. Eng. Adv. Technol.*, vol. 8, no. 3, pp. 656–660, 2019, doi: 10.35940/ijeat.C1243.0283S19.

[28] C. Shen and others, "Deep learning in hydrology," *Water Resour. Res.*, vol. 56, no. 1, p. e2019WR025816, 2020, doi: 10.1029/2019WR025816.

[29] J. Fan *et al.*, "Comparison of Support Vector Machine and Extreme Gradient Boosting for Precipitation Forecasting," *Atmosphere (Basel).*, vol. 12, no. 6, p. 709, 2021, doi: 10.3390/atmos12060709.

[30] E. Y. Obsie, I. Masih, and A. Shiferaw, "Enhancing drought prediction through machine learning with mutual information-based feature selection," *Sci. Total Environ.*, vol. 912, p. 168704, 2025, doi: 10.1016/j.scitotenv.2024.168704.

[31] I. E. Kumar, S. Venkatasubramanian, C. Scheidegger, and S. Friedler, "Problems with Shapley-value-based explanations as feature importance measures," in *Proceedings of the 37th International Conference on Machine Learning (ICML)*, 2020.

[32] Z. He, S. Liu, P.-W. Chan, and K. Zhao, "Integrating SHAP with machine learning for quantitative precipitation estimation," *Front. Environ. Sci.*, vol. 10, p. 1057081, 2022, doi: 10.3389/fenvs.2022.1057081.

[33] A. Idris, A. Khan, and Y. S. Lee, "Intelligent churn prediction in telecom: Employing mRMR feature selection and RotBoost based ensemble classification," *Appl. Intell. 51*, 2021, doi: 10.1007/s10489-020-01871-z.

[34] S. Aljawarneh, M. Aldwairi, and M. B. Yassein, "Anomaly-based intrusion detection system through feature selection analysis and building hybrid efficient model," *J. Comput. Sci.*, vol. 25, no. 1, pp. 152–160, 2018, doi: 10.1016/j.jocs.2017.03.006.

[35] A. Zulfa, "Hybrid feature selection framework for weather variable identification," *J. Sist. Komput.*, vol. 15, no. 1, pp. 45–56, 2025, doi: 10.30743/jsk.v15i1.4157.

[36] A. Zulfa, A. Saikhu, H. Pradana, and I. Budiawan, "An adaptive stacking approach for monthly rainfall prediction with hybrid feature selection," *Ultim. Comput. J. Sist. Komput.*, vol. 17, no. 1, 2025, doi: 10.31937/sk.v17i1.4157.

[37] J. Fan *et al.*, "Prediction of precipitation using XGBoost and deep learning models," *Atmosphere (Basel).*, vol. 12, no. 6, p. 709, 2021, doi: 10.3390/atmos12060709.

[38] 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.

[39] R. J. Hyndman, "Measuring forecast accuracy," *Int. J. Forecast.*, vol. 34, no. 1, pp. 1–2, 2018, doi: 10.1016/j.ijforecast.2017.08.001.

[40] C. J. Willmott and K. Matsuura, "Advantages of the mean absolute error over the root mean square error in assessing average model performance," *Clim. Res.*, vol. 30, no. 1, pp. 79–82, 2005, doi: 10.3354/cr030079.

[41] T. Chai and R. R. Draxler, "Root mean square error (RMSE) or mean absolute error (MAE)?," *Geosci. Model Dev.*, vol. 7, no. 3, pp. 1247–1250, 2014, doi: 10.5194/gmd-7-1247-2014.

[42] M. El Hafyani, K. El Himdi, and S.-E. El Adlouni, "Improving monthly precipitation prediction accuracy using machine learning models: a multi-view stacking learning technique," *Front. Water*, vol. Volume 6-, 2024, doi: 10.3389/frwa.2024.1378598.

[43] V. N. Vapnik, "An overview of statistical learning theory," *IEEE Trans. Neural Networks*, vol. 10, no. 5, pp. 988–999, 1999, doi: 10.1109/72.788640.

[44] P. Domingos, "A few useful things to know about machine learning," *Commun. ACM*, vol. 55, no. 10, pp. 78–87, 2012, doi: 10.1145/2347736.2347755.

[45] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, 2nd ed. Springer, 2009.

[46] R. Kohavi and G. H. John, “Wrappers for feature subset selection,” *Artif. Intell.*, vol. 97, no. 1--2, pp. 273–324, 1997, doi: 10.1016/S0004-3702(97)00043-X.

[47] C. Nadeau and Y. Bengio, “Inference for the generalization error,” *Mach. Learn.*, vol. 52, no. 3, pp. 239–281, 2003, doi: 10.1023/A:1024068626366.

[48] C. Rudin, “Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead,” *Nat. Mach. Intell.*, vol. 1, pp. 206–215, 2019, doi: 10.1038/s42256-019-0048-x.

[49] G. Chandrashekar and F. Sahin, “A Survey on Feature Selection Methods,” *Comput. \& Electr. Eng.*, vol. 40, no. 1, pp. 16–28, 2014, doi: 10.1016/j.compeleceng.2013.11.024.

[50] L. Zhang *et al.*, “Interpolated or satellite-based precipitation? Implications for hydrological modeling in a meso-scale mountainous watershed on the Qinghai-Tibet Plateau,” *J. Hydrol.*, vol. 583, p. 124629, 2020, doi: 10.1016/j.jhydrol.2020.124629.

BIBLIOGRAPHY OF AUTHORS

Bambang Widoyono is a lecturer in Data Science Study Program, Faculty of Information Technology and Data Science, Sebelas Maret University (UNS). He obtained his master’s degree from the University of Indonesia (UI) after completing his undergraduate studies in Informatics Engineering at IT Telkom Bandung. His research interests include data management, machine learning, and software engineering.

Muhammad Fahmy Nadhif is a lecturer in the Data Science Study Program, Faculty of Information Technology and Data Science, Universitas Sebelas Maret. He received his bachelor’s degree from Brawijaya University (UB) and a master’s degree in informatics from the Bandung Institute of Technology (ITB). His research focuses on computer vision, machine learning, natural language processing, and cybersecurity.

Ridha Adjie Eryadi is a lecturer in Informatics at the University of Digital Technology. She completed his undergraduate studies at Telkom Institute of Technology and earned his master’s degree from the University of Indonesia in 2019. Her academic activities include teaching, research, and scholarly contributions in the field of information technology.