Journal of Artificial Intelligence and Engineering Applications

Website: https://ioinformatic.org/

15th February 2026. Vol. 5. No. 2; e-ISSN: 2808-4519

Extreme Gradient Boosting for Daily Rainfall Forecasting in Medan City

Rabiahtul Adawiah Hasyani[1]*, Yulita Molliq Rangkuti[2], Elmanani Simamora[3], Said Iskandar Al Idrus[4], Kana Saputra S[5], Melanthon Pardamean Haloho[6]

[1,2,3,4,5]*Universitas Negeri Medan*
[6]*BMKG Wil. 1 Medan*
rabiahtuladawiah384@gmail.com[1]*, yulitamolliq@gmail.com[2], elmanani_simamora@unimed.ac.id [3],saidiskandar@unimed.ac.id [4], kanasaputras@unimed.ac.id[5], melanthon.haloho@bmkg.go.id.[6]

Abstract

High and irregular rainfall in Medan City frequently leads to flood events, making accurate rainfall forecasting essential for flood mitigation. This study aims to forecast daily rainfall by applying the Extreme Gradient Boosting (XGBoost) method. The modeling process involves data preprocessing, feature engineering, and data standardization. The novelty of this research lies in the integration of XGBoost with advanced feature engineering techniques and feature importance thresholding to improve model efficiency and accuracy. The evaluation results indicate that, using a 90:10 data split ratio, the model achieved an MAE of 1,05 and an RMSE of 1,91. Feature importance analysis reveals that CH_diff7, CH_diff1, and CH_lag1 are the most dominant predictors in daily rainfall forecasting. Furthermore, model accuracy was improved through feature selection, where the use of the top five features reduced the MAE to 0,99 and the RMSE to 1,66. These findings demonstrate that the XGBoost method, combined with feature engineering and feature selection processes, provides an effective approach for daily rainfall forecasting in Medan City.

Keywords: *XGBoost, Forecast, Rainfall, Feature Importance, Feature Engineering*

1. Introduction

Rainfall in Indonesia has unique characteristics that are influenced by factors such as altitude, climate, and season, resulting in different rainfall levels in each region [1]. One of the provinces in Indonesia that has relatively high rainfall is North Sumatra province. This was evident in the events that occurred in 2020, when rainfall intensity reached 4,380 mm, which is relatively high compared to the normal annual rainfall of 1,000-3,000 mm per year [2]. This high rainfall has caused the city of Medan, the capital of North Sumatra province, to remain vulnerable to flooding [3].

According to BNPB, the worst flooding in Medan occurred in 2022. Data from the Deli Serdang Climatology Station shows that the flooding occurred from February 27, 2022. Based on information obtained from the BMKG in the North Sumatra region, February should have been the peak of the first dry season. However, in reality, flooding still occurred in Medan [4]. This discrepancy highlights the importance of accurate rainfall forecasting as part of flood risk mitigation efforts.

The accuracy of forecasting depends on two important things, namely the selection of the right method and the availability of complete rainfall data. The importance of making predictions has encouraged researchers to create more accurate forecasting techniques [5]. The use of different techniques will produce different accuracy values. Therefore, it is important to select the appropriate algorithm and model it according to the needs [6].

With the development of technology, researchers are using data mining techniques, big data analysis, and various machine learning algorithms to improve accuracy. Based on the results of research conducted by researchers, machine learning algorithms have a higher level of effectiveness than conventional data mining techniques for predicting rainfall and weather [7]. One of the machine learning algorithms used for prediction or regression is XGBoost (eXtreme Gradient Boosting).

The XGBoost algorithm is a modern and popular algorithm based on the Gradient Boosting Machine (GBM) framework. This algorithm works well on various tasks, including regression, classification, and ranking [8]. The advantages of this algorithm are its speed in processing and its ability to perform calculations outside the core. However, this algorithm has a weakness, namely the possibility of overfitting, which can be overcome by experimenting with modeling parameters [9].

Previous studies have demonstrated that XGBoost is effective for daily rainfall prediction. Anwar et al. reported satisfactory results in forecasting daily rainfall in Semarang City, achieving an RMSE of 2.7 mm and an MAE of 8.8 mm [8]. Liyew et al. evaluated several machine learning techniques for daily rainfall prediction, including Multivariate Linear Regression (MLR), Random Forest (RF), and XGBoost, and found that XGBoost outperformed the other methods in terms of predictive accuracy [7]. In addition, Muslim Karo Karo investigated the application of XGBoost combined with feature importance for forest and land fire classification, showing that feature importance analysis improved classification accuracy by identifying six to seven dominant variables influencing hotspot occurrences [10].

Based on the above explanation, this study will forecast rainfall intensity using the XGBoost method and feature importance. The use of feature importance aims to identify variables that have a significant influence on rainfall. With this approach, it is hoped that the resulting rainfall forecasts will have better accuracy so that the government and the citizen can take preventive and mitigating measures in facing the potential impact of high rainfall in the city of Medan.

2. Literature review

2.1. Rainfall

Rainfall is the amount of rainwater collected in a measuring device on a flat surface that does not absorb, seep, or flow. One millimeter of rainfall means that in an area of one square meter, rainwater has accumulated to a height of one millimeter, or a volume equivalent to one liter [11]. The BMKG uses millimeters (mm) as the standard unit of measurement for rainfall. Rain gauges are divided into two types, namely conventional rain gauges of the Observatory (Obs.) or non-recording type and automatic rain gauges or self-recording rain gauges. Automatic rain gauges consist of two types, namely Hellmann-type automatic rain gauges that use a float system and automatic rain gauges that use a tipping bucket system [12].

2.2. XGBoost

XGBoost, or Extreme Gradient Boosting, is included in ensemble learning that relies on decision trees and uses gradient boosting. XGBoost was developed by Chen and Guestrin in 2016 at the University of Washington as part of a research project [13]. In regression calculations, the first step is to calculate the initial prediction value by calculating the average value of the target variable [14]. Once the initial prediction is obtained, XGBoost continues the training process by gradually building a model using a decision tree ensemble approach. The model is formed gradually to correct errors from previous predictions. This improvement process is carried out by optimizing the loss function, which is a function that measures the difference between the actual value and the predicted value [15]. In regression cases, the commonly used loss function is mean squared error (MSE), which can be denoted in the following equation (1).

$$L(\theta) = \sum_{i=1}^{\mathrm{n}} (y_i - \hat{y}_i)^2 \tag{1}$$

Where y_i is the actual value in the sample, $\hat{y}_i$is the predicted model value.

To minimize the loss function value, XGBoost calculates the first derivative (gradient) of the loss function against the current prediction value. Therefore, the residual value to be used next is described in equation (2) below.

$$g_i = y_i - \hat{y}_i \tag{2}$$

The value g_i is called the residual, which is the difference between the actual value and the previous prediction. This residual will be the learning target in each iteration so that the new model formed will try to correct the errors from the previous prediction.

The next step is to gradually build a decision tree using an additive approach, where a number of simple models (weak learners) are formed sequentially. The process of adding each tree is optimized by an objective function, which is a combination of a loss function to measure prediction errors and a regularization term to prevent overfitting. The objective function in the t-th iteration can be described in the following equation (3).

$$\tilde{\mathcal{L}}^{(t)} = \sum_{i=1}^{n} l(y_i, \hat{y}^{t-1} + f_k(x_i)) + \Omega(f_k) \tag{3}$$

Where, $\tilde{\mathcal{L}}^{(t)}$ is the objective function, y^t is the prediction at iteration t, $\hat{y}^{t-1}$ is the prediction at the previous iteration $(t-1)$, $f_k(x_i)$ is the prediction result from tree k for data i, and $\Omega(f_k)$ is the regularization parameter.

In order for the addition of new models to be effective, XGBoost minimizes the objective function using a second-order Taylor expansion approach. Through this simplification, the objective function can be re-expressed as in equation (4) as follows [16].

$$\tilde{\mathcal{L}}^{(t)} = \sum_{j=1}^{T} \left[(G_j) w_j + \frac{1}{2} (H_j + \lambda) w_j^2 \right] + \gamma T \tag{4}$$

Where, w_j is the weight value (output) on leaf j, G_j is the number of gradients, H_j is the number of Hessians, λ is the regularization parameter, and T is the number of leaves on the tree. The weights on each leaf can be calculated using equation (5).

$$w_j^* = -\frac{G_j}{H_j + \lambda} \tag{5}$$

Where, w_j^* is the weight value (output) at leaf j, G_j is the number of gradients, H_j is the number of Hessians, λ is the regularization parameter.

Then, the optimal objective function is obtained as described in equation (6).

$$\tilde{\mathcal{L}}^{(t)}(q) = -\frac{1}{2}\sum_{j=1}^{T}\frac{G_j^2}{H_j+\lambda} + \gamma T \tag{6}$$

Where G_j is the sum of gradients, H_j is the sum of Hessians, λ is the regularization parameter, γ is the regulation parameter, and T is the number of leaves in the tree.

The predicted value $\hat{y}_i$ is obtained from the sum of the predicted scores from all K trees, as formulated in equation (7) [17].

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

Where, $\hat{y}_t$ is the model prediction value, x_i is the independent variable for sample i, f_k is the kth model function, $f_k(x_i)$ is the prediction result from the -kth tree for data -ith, and F is the set of all functions containing all the regression trees that have been formed.

During the process, the model's performance is also greatly influenced by the hyperparameter settings, which are parameters determined before training. Some important hyperparameters used include the number of decision trees (n_estimators), learning rate (learning_rate), and maximum tree depth (max_depth). These values are adjusted to ensure that the model does not experience overfitting. Hyperparameters play an important role in the performance of the XGBoost method [18]. Hyperparameters are parameters that are set before the learning process of a model and are not parameters obtained through the training process. The method for determining the hyperparameters of the XGBoost algorithm is as follows [19].

a First, set the number of estimators to optimize XGBoost by setting the learning rate and other parameters.
b Then, combine max_depth to optimize XGBoost.
c Next, adjust gamma to make the model more conservative with the parameters specified in steps a and b.
d Afterthat, combine subsample and colsample_bytree to prevent overfitting.
e Then, increase the parameters to make the model more conservative.
f Finally, reduce the learning rate to prevent overfitting.

2.3. Feature importance

Feature importance is one way to evaluate the features or variables used in a model [20]. Feature importance aims to measure how much a feature contributes to or correlates with the dependent variable. The higher the value of a feature, the more important it is to the prediction results [18]. By determining feature importance, variables that are highly influential in improving model performance can be identified. This is done explicitly and requires a threshold value. Features with values smaller than the threshold will be removed, while those with values greater than the threshold will be used in the study [10].

2.4. Matrix evaluation

To assess the performance of the prediction model, this study uses two evaluation metrics, namely Root Mean Square Error (RMSE) and Mean Absolute Error (MAE). These two metrics are used to measure the error rate between the model's predicted values and the actual values. Root Mean Square Error (RMSE) is a measurement method that measures the difference between the predicted value of a model and the observed value or label. RMSE is calculated from the square root of Mean Square Error (MSE). The accuracy of the measurement error estimation method is indicated by a small RMSE value [21]. Equation (8) is the description of the RMSE calculation formula.

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(y_i-\hat{y}_i)^2}{n}} \tag{8}$$

Where, y_i is the actual data value, $\hat{y}_i$ is the predicted value, and n is the amount of data.

Meanwhile, Mean Absolute Error (MAE) is a method of calculating the average absolute difference between the actual value and the predicted value or a measure of the deviation of the recommendation from the actual value that has been determined. A low MAE value indicates that the prediction is better or has a small error [22]. Equation (9) is the derivation of the MAE calculation formula.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i-\hat{y}_i| \tag{9}$$

Where, y_i is the actual data value, $\hat{y}_i$ is the predicted value, and n is the number of data points.

3. Research Metodology

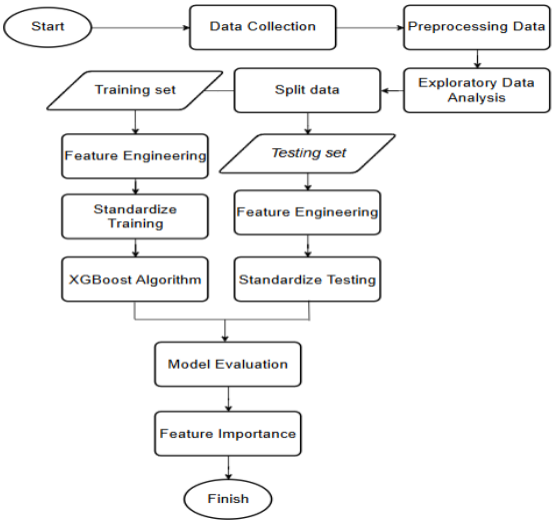

Fig. 1: Research Flowchart

Figure 1 shows the flowchart of the entire research process. The following is an explanation of the research flowchart illustrated in Figure 1.

3.1. Data Colection

The dataset used in this research consists of 3,652 days of observations from 2013 to 2022. The dataset used has six variables, including rainfall, duration of sunshine, maximum temperature, minimum temperature, average temperature, and average humidity.

3.2. Data Pre-Processing

After the data collection process, the next step is to perform data preprocessing. The data to be analyzed will undergo preprocessing based on the problems found in the dataset, so that the data is ready to be processed in the next stage. In this study, there are missing values, so a technique is needed to handle missing values using linear interpolation.

3.3. EDA (Exploratory Data Analysis)

This stage is the initial process in data processing, which aims to understand the characteristics, structure, and important components in the dataset. This exploration stage helps researchers understand the structure of the dataset, recognize extreme characteristics, and provides a clear basis for the next modeling process.

3.4. Data Split

In this study, the dataset was divided into training and testing sets using a time series–based approach. The data were split by year with a ratio of 90:10. This time-based splitting strategy was adopted to ensure a realistic evaluation scenario, in which the model is trained on historical data and evaluated on future observations. Such an approach preserves the temporal dependency within the data and allows the evaluation results to more accurately reflect real-world forecasting conditions [23].

3.5. Data Standardization

After the dataset has been cleaned, the data are standardized to ensure that all variables are measured on a comparable scale. Standardization transforms each feature to have a mean of zero and a standard deviation of one. This step is necessary because the variables originate from different units and value ranges. By applying standardization, each variable contributes proportionally to the model training process, preventing features with larger numerical ranges from dominating the learning process. Standardization is therefore applied to normalize continuous variables so that they contribute equally to the analysis [24].

3.6. Feature Engineering

In this study, feature engineering was used to generate new variables that could help XGBoost recognize patterns in rainfall data. This study used feature engineering that included lag features, rolling mean, differencing, and cyclic features. Using this feature engineering can improve XGBoost performance by adding relevant information.

3.7. Building a Model Using the XGBoost Method and Feature Importance

At this stage, the model was built using the XGBoost method to perform manual forecasting. This method was built using the Gradient Boosting approach, which combines trees gradually. In building this model, hyperparameters play a role in determining the complexity and performance of the model. Proper parameter settings are very important so that the model does not experience overfitting or

underfitting, thereby producing optimal predictions. Next, we look at the correlation between variables using feature importance. The goal is to measure the contribution or correlation of features to the dependent variable.

3.8. Evaluation

The forecasting results are evaluated using MAE and RMSE. These evaluation calculations are based on the prediction results of the test data against the actual data. MAE is used to measure the average absolute error between the predicted value and the actual value, providing an overview of the prediction error in general. Meanwhile, RMSE calculates the square root of the mean squared error, thereby imposing a greater penalty on prediction errors with large values. MAE and RMSE do not have specific threshold values to determine whether a model is good or not, as their values are highly influenced by the scale of the data used. Therefore, in this study, MAE and RMSE values are considered effective if they show relatively small prediction errors relative to the scale of the data used. MAE and RMSE are most effective when used to compare model performance rather than as absolute evaluation measures [25].

4. Result and Discussion

The dataset containing rainfall, average temperature, minimum temperature, maximum temperature, average humidity, and duration of sunshine obtained from BMKG Region 1 Medan City requires preprocessing to prepare the data for analysis. The dataset used in this study is visualized in Table 1.

Table 1: Rainfall Dataset

Date	Trata-rata	Tmax	Tmin	Rhrata-rata	CH	LPM
01/01/2013	28,1	32,2	24,4	82	0,2	1,0
02/01/2013	28,1	24,2	23,8	78	1,6	6,5
03/01/2013	25,9	30,6	24,0	88	15,5	3,0
04/01/2013	27,6	33,6	24,0	81	0	5,5
...	...	...	...	...	...	....
28/12/2022	26,7	32	23,2	83	24	0,0
29/12/2022	26,2	30,3	23,4	80	11	4,0
30/12/2022	26,5	30,3	23,5	78	4	3,8
31/12/2022	25,3	30,9	21,8	78	26	2,9

Next, the dataset preprocessing stage was carried out, which consisted of two stages, namely data type conversion and missing value handling. The first stage was to convert the data type in the 'date' column to 'datetime' and set it as a set index. This was done so that 'date' could be extracted into more informative time features. Next, the maximum temperature (Tmax) column, which has an 'object' data type, is converted to 'float'. The second stage is missing value handling. Based on the results of the examination, there are three variables that have missing values, namely maximum temperature (Tmax) with 5 values, minimum temperature (Tmin) with 2 values, and rainfall (CH) with 769 values. To overcome this, the linear interpolation method is used.

Not only that, but the date column is set as an index to facilitate the time series analysis process. Once the data has been verified as clean, the next step is to perform feature analysis to see the relationship between each variable. This analysis is performed using a correlation heatmap to determine how strong the relationship is between the target variable (rainfall) and other variables used as predictors

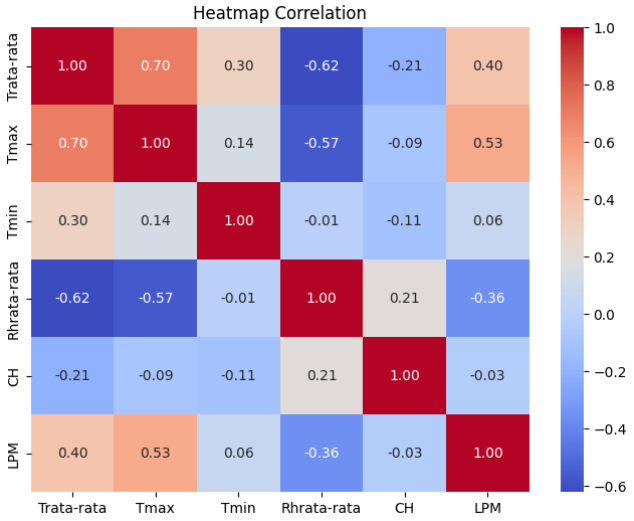

Fig. 2: Heatmap Correlation

Figure 2 visualizes the correlation between the target variable and the predictor variable. CH has a weak positive correlation with average humidity (Rhrata-rata) of 0.21, which means that rainfall tends to increase as humidity rises. Meanwhile, CH has a weak negative correlation with temperature (Trata-rata, Tmax, and Tmin), so that changes in temperature do not greatly affect rainfall. A strong positive correlation is seen between Tmax and Trata-rata of 0.70, while Trata-rata and Rhrata-rata show a negative correlation of -0.62. Overall, humidity is the factor most closely related to rainfall compared to other climate variables.

After going through a series of research dataset preprocessing stages, the data used is in optimal condition and ready to be used in the modeling process. The next stage is to divide the data. The data division used is 90:10. The training data range is from 2013 to 2021, and the testing data is from 2022. After splitting the dataset, feature engineering is performed to generate new informative variables, including differencing features (diff1, diff7), cyclic features (day sin, day cos), lag features (1, 2, 7, 14, 30), rolling mean windows (2 and 7), and standardization to normalize the data scale across all features except the target variable (rainfall).

The modeling process using XGBoost to predict rainfall was carried out after obtaining the best parameter values. The parameters used in this study are listed in Table 4.8, which has the function of regulating how the XGBoost method works so that it does not experience overfitting or underfitting. In this study, the parameters used include the number of trees (n_estimators), learning rate (learning_rate), maximum tree depth (max_depth), number of samples used in each tree (subsample), and the proportion of features used in each iteration (colsample_bytree).

Table 2: Parameters value

Parameters	Value
n_estimator	800
Learning_rate	0,01
Max_depth	3
Colsample_bytree	0,8
subsample	0,8
gamma	1
Reg_lamda	1

Table 2 presents the parameters used in this study. As shown in Table 2, the main parameters applied include the number of decision trees (*n_estimators*) set to 800, a learning rate (*learning_rate*) of 0.01, a maximum tree depth (*max_depth*) of 3, and *subsample* and *colsample_bytree* values of 0.8 each. In addition, the parameters *gamma* and *reg_lambda* were employed as regularization controls to ensure model stability and prevent excessive model complexity. These parameter values were obtained through an experimental tuning process by testing several combinations for each parameter until an optimal configuration was identified, resulting in lower prediction errors while maintaining a balance between overfitting and underfitting.

After modeling with the optimal parameters, the performance of the XGBoost model was evaluated using two evaluation metrics, MAE and RMSE. These metrics were selected because they provide a clear representation of prediction accuracy and error magnitude. MAE represents the average absolute difference between actual and predicted values, while RMSE places greater emphasis on larger errors, thereby capturing the variability of prediction errors. In general, the RMSE value is always greater than or equal to the MAE. A larger gap between RMSE and MAE indicates higher variability in prediction errors, whereas equal values suggest that the prediction errors are relatively uniform [7]. The evaluation results of the model on the test data under three data-splitting scenarios are presented in Table 3.

Table 3: Evaluation

Evaluation Matrix	90 : 10
MAE	1,05
RMSE	1,91

Table 3 presents the evaluation results of the third scenario, which applies a 90:10 ratio between training and testing data using the MAE and RMSE metrics. Based on Table 3, the training data yield an MAE value of 1,05 and an RMSE value of 1,91, indicating that the model is able to learn the underlying data patterns effectively. However, for the testing data, the MAE and RMSE values increase to 1.05 and 1.91, respectively. To provide a clearer representation of the model's performance, a visualization in the form of a comparison graph between the actual rainfall values and the XGBoost prediction results is presented below.

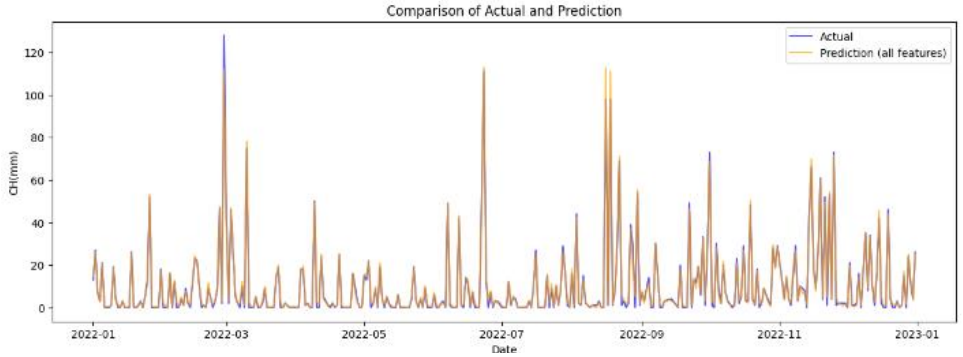

Fig. 3: Visualization Between Actual and Prediction

Figure 3 illustrates the comparison between actual rainfall and forecasting results using a 90:10 data split ratio. In Figure 3, the blue line represents the actual rainfall data, while the orange line indicates the model's predicted values. Overall, the predicted pattern is able to follow the trend of the actual data, both during periods of low rainfall and during significant increases in rainfall intensity.

The model is capable of capturing extreme spikes caused by heavy rainfall events that occurred in March, although the predicted values do not fully match the actual observations. This indicates that the XGBoost model demonstrates a strong ability to represent rainfall patterns, despite some discrepancies at peak rainfall values. The visualization of rainfall trend graphs for the training data, testing data, and the forecasting results on the testing data is presented in the figure.

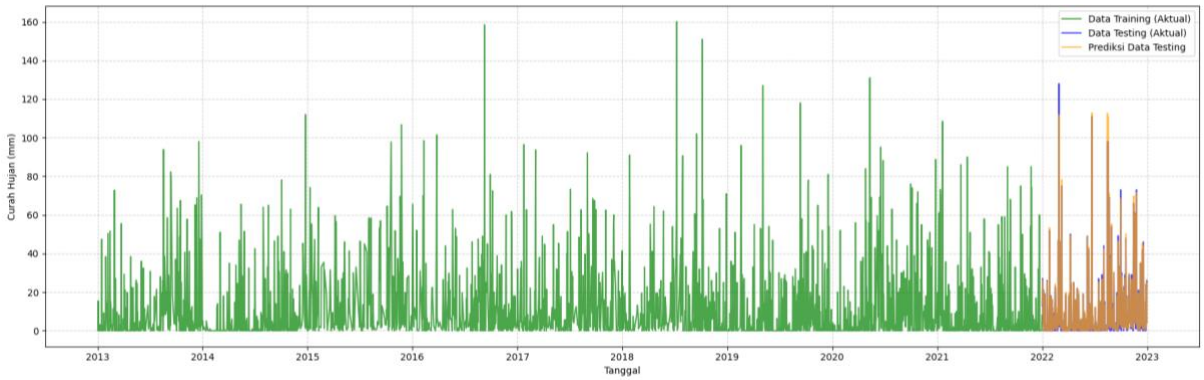

Fig. 4: Visualization of Daily Rainfall

Figure 4 presents a graph of daily rainfall measured in millimeters (mm). The green line represents the actual rainfall data from the training dataset, while the orange line indicates the predicted rainfall values on the testing dataset. In addition, the red line or points show the rainfall forecasts for the next seven days. The next step in this study is to conduct a feature importance analysis using the XGBoost model.

Feature importance analysis in the XGBoost model is performed to identify the contribution of each variable to the rainfall prediction results. The feature importance values indicate the extent to which a feature influences the model in producing more accurate predictions. A higher importance value signifies a greater contribution of the feature in the model building process.

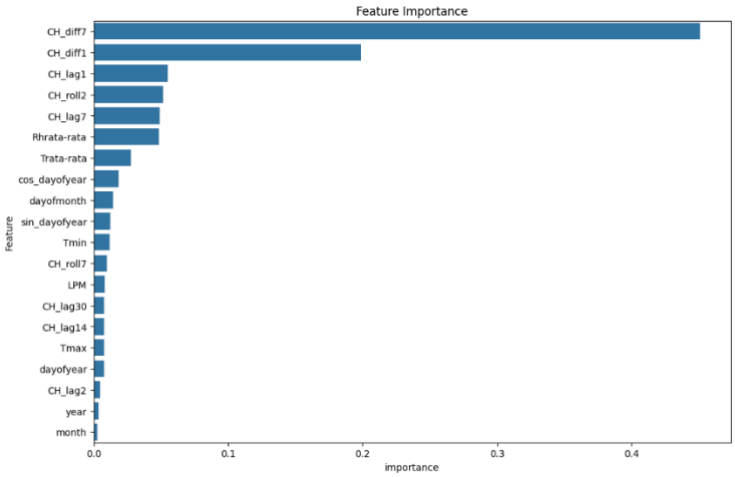

Fig. 5: Feature Importance Analysis

Figure 5 presents the visualization of feature importance from the XGBoost model used in this study. Based on Figure 5, the features CH_diff7 and CH_diff1 have the highest importance values, indicating that changes in rainfall over the previous seven days and one day, respectively, exert the greatest influence on rainfall prediction results. The variable CH_lag1 also shows a substantial contribution, followed by other features such as CH_roll2, CH_lag7, and Rhrata-rata. In contrast, variables such as dayofyear, CH_lag2, year, and month exhibit lower importance values, suggesting that their influence on the prediction outcomes is relatively limited.

Table 4: Feature Importances Value

Feature	Importance
CH_diff7	0,450994
CH_diff1	0,198759
CH_lag1	0,055247
CH_roll2	0,051588
CH_lag7	0,049197
Rhrata-rata	0,048496
Trata-rata	0,027627
cos_dayofyear	0,018321
dayofmonth	0,014683
sin_dayofyear	0,012646

Tmin	0,012014
CH_roll7	0,009633
LPM	0,008456
CH_lag30	0,007958
CH_lag14	0,007828
Tmax	0,007763
dayofyear	0,007707
CH_lag2	0,004591
year	0,003935
month	0,002557

Table 4 presents the feature importance values for each variable. These values reflect the magnitude of each feature's contribution to the model's forecasting results. As shown in Table 4, the most dominant feature is CH_diff7, with an importance value of 0,450994, followed by CH_diff1 with a value of 0,198759 and CH_lag1 with a value of 0.055247. Subsequently, the model is evaluated using various feature importance threshold values to examine the effect of the number of selected features on model performance.

Table 5: Evaluation Based on -n Features

Thresh	n	RMSE	MAE
0,00256	20	1,91	1,05
0,00393	19	1,94	1,07
0,00459	18	1,85	1,05
0,00771	17	1,87	1,04
0,00776	16	1,95	1,05
0,00783	15	1,93	1,05
0,00796	14	1,76	1,01
0,00846	13	1,90	1,05
0,00963	12	1,86	1,02
0,01201	11	1,97	1,09
0,01265	10	1,89	1,03
0,01468	9	1,92	1,03
0,01832	8	2,22	1,10
0,02763	7	1,91	1,08
0,04850	6	1,86	1,09
0,04920	5	1,66	0,99
0,05159	4	2,60	1,45
0,05525	3	2,56	1,47
0,19876	2	5,02	3,44
0,45099	1	5,60	3,63

Table 5 presents the evaluation results of the XGBoost model performance based on the number of features used after feature selection using feature importance values. Table 5 shows the model evaluation results according to the feature importance threshold applied to determine the optimal number of features. The threshold value serves as a minimum boundary for selecting features considered important by the model.

It can be observed that when the number of features is reduced to 5 using a threshold of 0,04920, the RMSE decreases to 1,66 and the MAE to 0,99. However, when too few features are used, the model performance declines significantly. This is evident when only one feature is selected with a threshold of 0,45099, where the RMSE increases sharply to 5,60 and the MAE rises to 3,63. This condition indicates that the model loses critical information required to produce accurate predictions. Overall, these results suggest that not all features contribute significantly to the model, and removing features with low importance can improve forecasting accuracy.

5. Conclusion

Based on the research results, daily rainfall forecasting in Medan City using the XGBoost method was conducted through a series of preprocessing, feature engineering, and data standardization steps aimed at preparing the dataset for modeling. The input variables used in this study included maximum temperature (Tmax), minimum temperature (Tmin), average temperature (Trata-rata), average relative humidity (Rhrata-rata), sunshine duration (LPM), as well as engineered features such as lag variables, rolling means, time-based features, differencing features, and cyclical features. These variables were proven to assist the model in capturing patterns in the time-series data.

The implementation of the XGBoost method using the Python programming language demonstrated strong performance in forecasting daily rainfall. The evaluation results using a 90:10 data split ratio yielded an MAE value of 1,05 and an RMSE value of 1,91. Both the evaluation metrics and visualizations indicate that XGBoost is capable of capturing daily rainfall patterns in Medan City, although some discrepancies remain during periods of high or extreme rainfall intensity.

Feature importance analysis in this study successfully identified variables that significantly influence model performance. Based on the experimental results, the optimal configuration was achieved by applying a feature importance threshold of 0,04920, which resulted in the selection of five key features. Under this configuration, the model achieved an RMSE value of 1,66 and an MAE value of 0,99. These findings indicate that the XGBoost model using five dominant features is able to produce more accurate rainfall predictions.

Acknowledgement

The author sincerely thanks all parties who supported this study. Special appreciation is extended to the faculty members for their valuable input and constructive feedback, as well as to the team at Balai Besar Meteorologi dan Klimatologi Wilayah I Medan for their collaboration and support during the preparation of this journal article. All contributions are greatly appreciated.

References

[1] M. F. Ihsan and Y. Muliati, "Analisis data curah hujan yang hilang dengan menggunakan metode rata-rata aljabar dan metode resiprokal," Institut Teknologi Nasional Bandung, pp. 1–6, 2021. [Online]. Available: http://eprints.itenas.ac.id/1545/
[2] Badan Pusat Statistik Provinsi Sumatera Utara, "Rata-rata kelembaban udara, curah hujan, penyinaran matahari, kecepatan angin, dan penguapan menurut stasiun tahun 2020," 2020. [Online]. Available: https://sumut.bps.go.id
[3] D. M. Pasaribu, "Tinjauan perundangan terhadap kebijakan dalam penanggulangan bencana banjir di Kota Medan," Prosiding Mitigasi Bencana, pp. 36–42, Nov. 2021.
[4] E. Paramita, S. Humaidi, and Y. Darmawan, "Rainfall characteristics in Medan City with Pearson correlation analysis (case study of February 27, 2022)," Prisma Sains, vol. 11, no. 2, p. 561, 2023, doi: 10.33394/jps.v11i2.7852.
[5] F. Insani, S. Fadilah, and S. Sanjaya, "Prediksi cuaca Pekanbaru menggunakan fuzzy Tsukamoto dan algoritma genetika," in Seminar Nasional Teknologi Informasi, Komunikasi dan Industri (SNTIKI), 2020, pp. 255–262.
[6] D. Mahajan and S. Sharma, "Prediction of rainfall using machine learning," in Proc. 4th Int. Conf. Emerging Research in Electronics, Computer Science and Technology (ICERECT), 2022, doi: 10.1109/ICERECT56837.2022.10059679.
[7] C. M. Liyew and H. A. Melese, "Machine learning techniques to predict daily rainfall amount," Journal of Big Data, vol. 8, no. 1, 2021, doi:
[8] 10.1186/s40537-021-00545-4. M. T. Anwar, E. Winarno, W. Hadikurniawati, and M. Novita, "Rainfall prediction using Extreme Gradient Boosting," Journal of Physics: Conference Series, vol. 1869, no. 1, 2021, doi: 10.1088/1742-6596/1869/1/012078.
[9] A. Lisanthoni, F. I. Sari, E. L. Gunawan, and C. A. Adhigiadany, "Model prediksi kepadatan lalu lintas: Perbandingan algoritma Random Forest dan XGBoost," Prosiding Seminar Nasional Sains Data, vol. 3, no. 1, pp. 296–303, 2023, doi: 10.33005/senada.v3i1.126.
[10] I. Muslim Karo Karo, "Implementasi metode XGBoost dan feature importance untuk klasifikasi pada kebakaran hutan dan lahan," Journal of Software Engineering, Information and Communication Technology, vol. 1, no. 1, pp. 11–18, 2020.
[11] BMKG Wilayah III Denpasar, "Daftar istilah musim," Accessed: Oct. 28, 2025. [Online]. Available: https://bbmkg3.bmkg.go.id/daftar-istilah-musim
[12] A. Azwar, E. Meilianda, and M. Masimin, "Kajian pola curah hujan durasi panjang terkait dengan waktu kejadian banjir di Kabupaten Aceh Utara," Jurnal Arsip Rekayasa Sipil dan Perencanaan, vol. 4, no. 1, pp. 39–48, 2022, doi: 10.24815/jarsp.v4i1.16723.
[13] Z. A. Ali, Z. H. Abduljabbar, H. A. Taher, A. B. Sallow, and S. M. Almufti, "Exploring the power of eXtreme Gradient Boosting algorithm in machine learning: A review," Academic Journal of Nawroz University, vol. 12, no. 2, pp. 320–334, 2023.
[14] W. Kurniawan and U. Indahyanti, "Prediksi angka harapan hidup penduduk menggunakan metode XGBoost," Indonesian Journal of Applied Technology, vol. 1, no. 2, p. 18, 2024, doi: 10.47134/ijat.v1i2.3045.
[15] A. Yaqin, "Penilaian kredit menggunakan algoritma XGBoost dan Logistic Regression," Jurnal Informatika: Jurnal Pengembangan IT, vol. 8, no. 1, pp. 4–10, 2022, doi: 10.30591/jpit.v8i1.4337.
[16] C. Wade, Hands-On Gradient Boosting with XGBoost and Scikit-Learn. Birmingham, U.K.: Packt Publishing, 2020.
[17] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in Proc. ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining, 2016, pp. 785–794, doi: 10.1145/2939672.2939785.
[18] D. Tarwidi, S. R. Pudjaprasetya, D. Adytia, and M. Apri, "An optimized XGBoost-based machine learning method for predicting wave run-up on a sloping beach," MethodsX, vol. 10, p. 102119, 2023, doi: 10.1016/j.mex.2023.102119.
[19] S. F. N. Islam, A. Sholahuddin, and A. S. Abdullah, "Extreme gradient boosting (XGBoost) method in forecasting application and analysis of USD exchange rates against rupiah," Journal of Physics: Conference Series, vol. 1722, no. 1, 2021, doi: 10.1088/1742-6596/1722/1/012016.
[20] R. M. Syafei and D. A. Efrilianda, "Machine learning model using XGBoost feature importance and LightGBM to improve bankruptcy prediction," Recursive Journal of Informatics, vol. 1, no. 2, pp. 64–72, 2023, doi: 10.15294/rji.v1i2.71229.
[21] I. Daqiqil, Machine Learning: Teori, Studi Kasus, dan Implementasi Menggunakan Python. Yogyakarta, Indonesia: UR Press, 2021.
[22] R. Fadilla, R. Andarsyah, R. M. Awangga, and R. Habibi, Data Analytics: Peningkatan Performa Algoritma Rekomendasi Collaborative Filtering Menggunakan K-Means Clustering. Bandung, Indonesia: Kreatif Industri Nusantara, 2020.
[23] V. Cerqueira, L. Torgo, and I. Mozetič, "Evaluating time series forecasting models: An empirical study on performance estimation methods," Machine Learning, vol. 109, no. 11, pp. 1997–2026, 2020, doi: 10.1007/s10994-020-05910-7.
[24] M. Arhami and M. Nasir, Data Mining: Algoritma dan Implementasi. Yogyakarta, Indonesia: ANDI, 2020.
[25] T. O. Hodson, "Root-mean-square error (RMSE) or mean absolute error (MAE): When to use them or not," Geoscientific Model Development, vol. 15, pp. 5481–5496, 2022.