BAREKENG: Journal of Mathematics and Its Applications
December 2026 Volume 20 Issue 4 Page 3559-3574
P-ISSN: 1978-7227 E-ISSN: 2615-3017
https://doi.org/10.30598/barekengvol20iss4pp3559-3574

EARLY DETECTION OF RAINFALL ANOMALIES USING LSTM AND ISOLATION FOREST

Adhystira Raihannoeza Almadiva [1], Atika Ratna Dewi [2*], Aina Latifa Riyana Putri [3]

[1,2,3]*Data Science Study Program, Telkom University, Purwokerto Campus*
D. I. Pandjaitan Street No. 128, Kec. Purwokerto Selatan, Banyumas, 53147, Indonesia.

*Corresponding author's e-mail: * **atikad@telkomuniversity.ac.id***

Article Info

Article History:

Received: 29th November 2025
Revised: 08th May 2026
Accepted: 05th June 2026
Available online: 24th August 2026

Keywords:

Anomaly detection;
Isolation forest;
LSTM;
Rainfall.

ABSTRACT

The uncertainty of daily rainfall patterns in Cilacap Regency with extreme variations makes it difficult to detect hydrological anomalies early using traditional methods. This study aims to obtain the most optimal LSTM parameters for rainfall prediction models, evaluate model performance using the Mean Squared Error (MSE) also Root Mean Squared Error (RMSE) metric, and predict rainfall anomalies for the next year using Isolation Forest. Daily BMKG data from January 2015 to December 2024 were processed through preprocessing stages, including missing data handling and time sequence creation. The Long Short Term Memory model was trained using regularization techniques to avoid overfitting, and the prediction results were analyzed using Isolation Forest to identify anomalies. The experiment showed that the best combination of hyperparameters was LSTM with 50 units and a tanh activation function, dropout 0.3, followed by a first dense layer of 50 units with ReLU activation and a single output layer. This configuration resulted in a validation MSE of 23.5247161 and RMSE 4.8502, this result identified five cases of anomalies that were validated for suitability based on rainfall data from BPBD. These results demonstrate the model's ability to reconstruct rainfall patterns and detect potential anomalies, so the system has potential to support early warning efforts for disaster mitigation and water resource management in Cilacap.

How to cite this article:

A. R. Almadiva, A. R. Dewi and A. L. R. Putri., "EARLY DETECTION OF RAINFALL ANOMALIES USING LSTM AND ISOLATION FOREST", *BAREKENG: J. Math. & App.*, vol. 20, no. 4, pp. 3559-3574, Dec, 2026.

Journal homepage: *https://ojs3.unpatti.ac.id/index.php/barekeng/*
Journal e-mail: *barekeng.math@yahoo.com; barekeng.journal@mail.unpatti.ac.id*

Research Article · Open Access

1. INTRODUCTION

Indonesia is located near the equator, so it has a tropical climate with two main seasons, namely the rainy season and the dry season, which repeat every year and can be identified on a monthly scale [1]. Rainfall serves as an essential climatic factor that significantly affects human activities and vital sectors, including agriculture and disaster mitigation [2], [3]. One area that reflects the dynamics of Indonesia's tropical weather is Cilacap Regency in Central Java, where long-term observations from the Meteorology, Climatology, and Geophysics Agency (BMKG) report a 10-year mean monthly rainfall of 340 mm, with the highest monthly total in November (547.8 mm) and the lowest in August (144.8 mm). Unstable rainfall patterns have substantial impacts on communities, infrastructure, and the regional economy, therefore early prediction of rainfall anomalies is essential for disaster mitigation, water resource management, and planning of critical services (transportation, irrigation, etc.) [4]. Improvements in weather forecasting techniques are crucial for various industrial sectors in Indonesia, as accurate weather information directly affects operational efficiency [5]. Rainfall is a basic parameter in forecasting model development, which must be processed using appropriate analytical techniques to achieve high prediction accuracy. Accurate forecasting plays an important role in supporting disaster mitigation and water resource management, especially in Cilacap Regency [6].

This study focuses on predicting monthly rainfall anomalies in Cilacap using historical BMKG records, with the primary objective of producing projections that can inform risk mitigation and planning for extreme hydrometeorological events [7]. An anomaly refers to the difference between actual rainfall and the long-term climate average, which can be a positive (wet) or negative (dry) deviation. Prolonged extreme anomalies are a major factor in disasters such as floods and droughts [8], [9]. Rainfall time series generally exhibit complex and non-linear patterns influenced by seasonal cycles, long-term trends, and short-term fluctuations. Conventional linear models are often unable to represent these dynamics, prompting the use of machine learning techniques that can learn non-linear temporal relationships from extensive historical data. In this study, a hybrid method was applied, combining Long Short-Term Memory (LSTM) networks for sequence forecasting with Isolation Forests for anomaly identification [10].

LSTM is used because it represents an advanced recurrent architecture that overcomes the vanishing and exploding gradient problems found in standard RNNs, allowing the model to capture long-term dependencies in sequential data and produce more reliable predictions based on historical observations [11]. Isolation Forest is used as an unsupervised and non-parametric anomaly detection technique, which isolates data points using randomly constructed decision trees, observations that can be separated with fewer splits receive higher anomaly scores and are therefore considered more likely to be anomalous [13]. The combination of LSTM's strength in capturing temporal patterns and Isolation Forest's strength in identifying abnormal deviations makes this hybrid approach particularly well-suited for this problem. Meanwhile, the model performance will be evaluated using the standard error metric for regression Mean Squared Error (MSE). MSE will calculate larger errors by squaring the deviations, lower values indicate higher prediction accuracy [5].

Several previous studies have applied LSTM and hybrid approaches to rainfall and weather-related forecasting, one of which examined daily rainfall in Pekanbaru where an LSTM model yielded a best 50/50 train/test result of RMSE = 21.328 and MSE = 454.901, indicating solid baseline performance but room for hyperparameter tuning [14]. Another study combined PCA and LSTM to forecast daily rainfall in Luwu Utara, where PCA extracted dominant features and the PCA–LSTM model produced a MAPE = 0.0018 and successfully identified very heavy rainfall events [15]. In Semarang, researchers paired LSTM with Isolation Forest to flag the top 5% rainfall days as potential flood events, anomaly detections in the forecast table were validated against actual floods on 1–6 January 2022, demonstrating accurate early-warning capability [16]. A multi sector study using a four-layer LSTM reported best results at 100 epochs and batch size 50, with RMSE = 1.7444 mm for rainfall and MAPE = 1.9499% for temperature, supporting LSTM's applicability for transport, agriculture, and industry planning [17]. Finally, a monthly LSTM study in Denpasar using one hidden layer with 30 neurons and window size 6 achieved training MAE = 199.19 and testing MAE = 166.57, although extreme spikes elevated errors and suggested the need for further tuning or alternative methods [18].

Developing a reliable rainfall-prediction model for Cilacap Regency is strategically important for multiple sectors. The proposed framework combines Long Short-Term Memory (LSTM) networks with Isolation Forest anomaly detection to capture nonlinear temporal patterns and enhance disaster planning and mitigation, particularly for anticipating droughts and floods. Although similar approaches have been applied

in other regions, this study offers a novel methodological contribution by applying anomaly detection to one-year LSTM forecast outputs while deliberately preserving outliers during preprocessing to retain extreme-event signals. To the best of our knowledge, this is the first implementation that represents an early step toward developing a data-driven decision support framework for hydrometeorological hazard monitoring in Cilacap.

2. RESEARCH METHODS

2.1 Research Data

The primary dataset used in this study consists of daily meteorological records obtained from Stasiun Meteorologi Tunggul Wulung (Station ID: 96805), a Unit Pelaksana Teknis (UPT) station operated by BMKG (Badan Meteorologi, Klimatologi, dan Geofisika) under Region II, located in Cilacap Regency, Central Java, Indonesia. The station is situated at coordinates 7.7189°S, 109.0149°E, at an elevation of 8 meters above sea level, and operates under the WIB time zone (UTC+07:00). The dataset covers the period from 1 January 2015 to 31 December 2024, comprising 3,633 daily observations across 11 meteorological variables as presented in Table 1. Data were accessed through the BMKG online data portal (https://dataonline.bmkg.go.id) under the open-data provision for research purposes. Table 1 presents the variables used in this study.

Table 1. Research Variables

No	Variable	Description	Data Type	Unit
1	Date	Observation date	object	date type
2	TN	Minimum temperature	object	Degrees Celsius (°C)
3	TX	Maximum temperature	object	Degrees Celsius (°C)
4	TAVG	Average temperature	object	Degrees Celsius (°C)
5	RH_AVG	Average humidity	object	Percentage (%)
6	RR	Rainfall	object	Millimeters (mm)
7	SS	Sunshine duration	object	Hours
8	FF_X	Maximum wind speed	int64	Meters per second (m/s)
9	DDD_X	Wind direction	int64	Degrees (°)
10	FF_AVG	Average wind speed	int64	Meters per second (m/s)
11	DDD_CAR	Most frequent wind direction	object	Cardinal direction (N, S, E, W, NE, NW, SE, SW)

Data source: (BMKG - Meteorology, Climatology, and Geophysics Agency)

In addition to the BMKG meteorological dataset, validation data for anomaly detection were sourced from the Regional Disaster Management Agency of Central Java Province (BPBD Jawa Tengah) through the Cevadis (Central Java Disaster Information System) platform (https://cevadis.jatengprov.go.id). This dataset contains official records of reported disaster events in Cilacap Regency during the prediction period of January–December 2025, and was used to cross-validate the anomaly dates identified by the Isolation Forest model against actual occurrences of floods, landslides, and extreme weather events.

2.2 Recurrent Neural Network

Recurrent Neural Networks (RNNs) are a type of neural network in which neurons are connected to form a directed graph that corresponds to the order of input data. With the help of internal states or memory, RNNs are able to handle sequential inputs of varying lengths and incorporate information from previous time steps when generating output. These properties make RNNs particularly well suited for applications that rely on temporal patterns, including language modeling, speech recognition, and time series forecasting [19]. RNNs can be understood as the repeated use of the same neural network for each different input, where input variations are accommodated by conditions stored in the hidden layer from the previous step. An illustrative image about RNNs Structure is shown in Fig. 1.

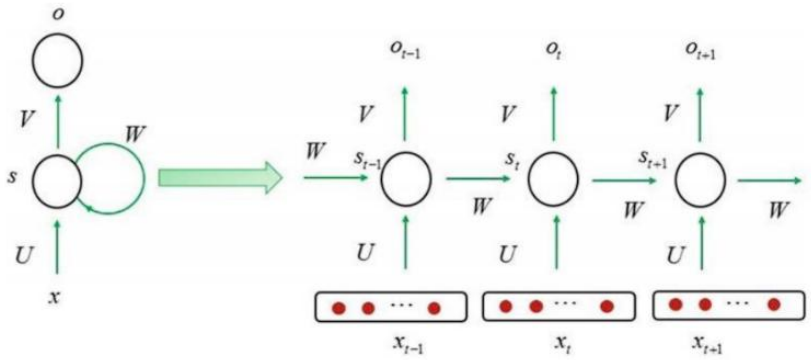


Figure 1. RNN Structure

For time *t* is :

$$\boldsymbol{S_t = \phi(Ux_t + WS_{t-1} + b_1)}, \quad (1)$$

$$\boldsymbol{o_t = \phi(VS_t + b_2)}, \quad (2)$$

$$\hat{y}_t = \boldsymbol{\phi(o_t)} \quad (3)$$

At time t, x_t is the input unit, S_t is the hidden unit, and o_t is the output unit. The connection weights in the network are denoted by V, W, and U, while b is the bias and ŷ is the predicted output value. The activation function is denoted by ϕ. As the volume and dimensionality of data grow, RNNs are required to retain increasingly more information from earlier time steps, which eventually leads to the vanishing gradient issue, where gradients diminish during the backpropagation through time process [20]. This is why LSTM was developed to overcome this weakness.

2.3 Long Short Term Memory (LSTM)

Long Short Term Memory (LSTM) is a type of Recurrent Neural Network (RNN) designed to learn which information needs to be stored (used) and which needs to be forgotten (ignored) at each time step. The LSTM structure consists of four main components, namely the input gate, forget gate, cell state, and output gate, which interact with each other to control the flow of information. The three main gates (input gate, forget gate, and output gate) play a role in controlling the information that enters the memory, thereby helping to overcome the problem of missing or exploding gradients, as well as a memory unit that regulates the flow of information to the next step [19], [21]. An illustrative image about LSTMs Structure is shown in Fig. 2.

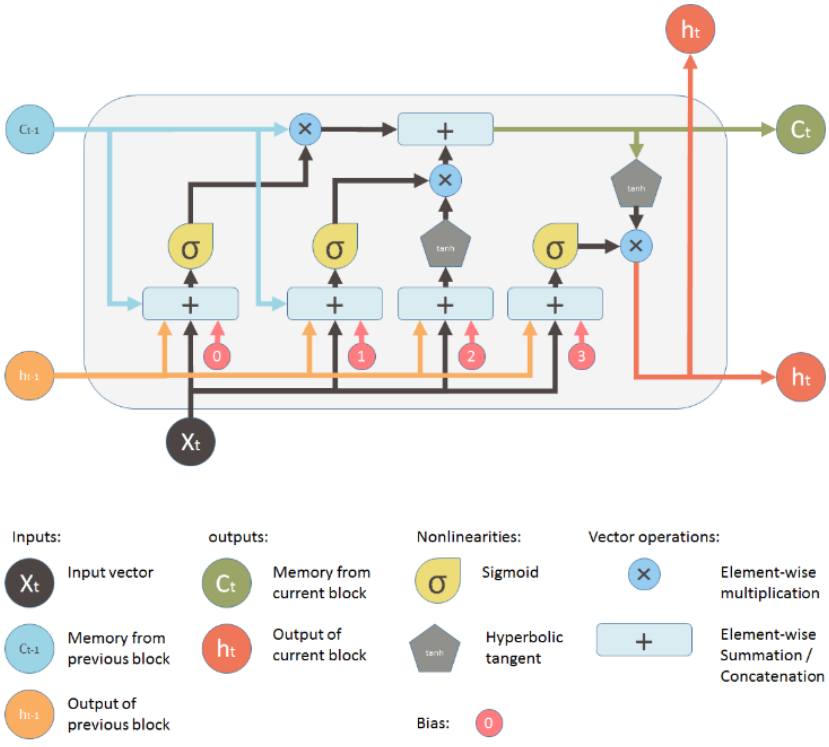


Figure 2. LSTM Structure

The forget gate, input gate, and output gate work together in the LSTM block to regulate the flow of information at each time step. The forget gate uses a sigmoid function to determine how much information from the previous cell state should be forgotten (values close to 0) or retained (values close to 1), while the input gate also uses sigmoid to control the proportion of new data from the current input x_t that should be stored in the cell state. After that, the output gate decides which part of the cell state will be converted into the hidden state h_t, by combining sigmoid and tanh so that only the most relevant information is passed on[19]. The combination of these three gates allows LSTM to learn when to store or discard information, which is very important in processing long-term sequential data. Following are the functions and formulas of each gate.

The First Gate is the Forget gate, plays a role in determining what information should be passed on or what should be discarded. Forget gate formulated in Eq. (4)

$$\mathbf{f_t} = \boldsymbol{\sigma}(\boldsymbol{W_f x}[\boldsymbol{h_{t-1}}, \boldsymbol{x_t}] + \boldsymbol{b_f}). \tag{4}$$

$\boldsymbol{W_f}$ represents the weight matrix, while b_f denotes the bias in the forget gate. The term x_t refers to the input data at time step t, and h_{t-1} is the output from the memory block at the previous time step. The symbol σ indicates the sigmoid activation function used within the LSTM gate operations.

The second gate, known as the input gate, is responsible for selecting which information will be incorporated as new knowledge and stored in the cell state. Input gate formulated in Eq. (5).

$$\boldsymbol{i_t} = \boldsymbol{\sigma}(\boldsymbol{W_i x}[\boldsymbol{h_{t-1}}, \boldsymbol{x_t}] + \boldsymbol{b_i}). \tag{5}$$

After determining the incoming information, the network then evaluates the output gate in Eq. (6), which controls how much of the updated cell state will be exposed as the hidden state.

$$\boldsymbol{o_t} = \boldsymbol{\sigma}(\boldsymbol{W_o x}[\boldsymbol{h_{t-1}}, \boldsymbol{x_t}] + \boldsymbol{b_o}). \tag{6}$$

The entire transition process from the old cell state c_{t-1} to the new cell state c_t can be explained by the following Eqs. (7), (8), and (9) where the activation function used is tanh.

$$\tilde{\boldsymbol{c}}_\mathbf{t} = \boldsymbol{tanh}(\boldsymbol{W_c x}[\boldsymbol{h_{t-1}}, \boldsymbol{c_t}] + \boldsymbol{b_c}), \tag{7}$$

$$\boldsymbol{c_t} = \boldsymbol{f_t}\, \boldsymbol{x}\, \boldsymbol{c_{t-1}} + \boldsymbol{i_t}\, \boldsymbol{x}\, \tilde{\boldsymbol{c}}_\mathbf{t}, \tag{8}$$

$$\boldsymbol{h_t} = \boldsymbol{o_t}\, \boldsymbol{x}\, \boldsymbol{tanh}\, \boldsymbol{c_t}. \tag{9}$$

Recurrent connections in LSTM add state or memory to the network and enable it to utilize sequential observations, where the internal memory makes the network output dependent on the last context of the input queue, rather than solely on the input being presented. LSTM networks incorporate an internal memory cell and recurrent connections regulated by gating mechanisms (input, forget, and output gates), which allow the network to retain information across long sequences, this enables modeling of long term dependencies in sequential data (text, signal, time-series) [12], [22].

2.4 Isolation Forest

Isolation Forest is an ensemble-based algorithm designed to detect anomalies by directly isolating them rather than modelling normal behavior first. Because anomalies are rare and structurally different, they can be separated more easily from regular observations [23]. The algorithm builds multiple isolation trees (iTrees), where each tree randomly selects a feature and a split value between its minimum and maximum range. This recursive splitting continues until a sample is isolated or a maximum depth is reached. Anomalies typically require fewer splits to isolate, resulting in shorter path lengths, which serve as the main indicator of anomaly likelihood. An illustration of this process is presented in Fig. 3.

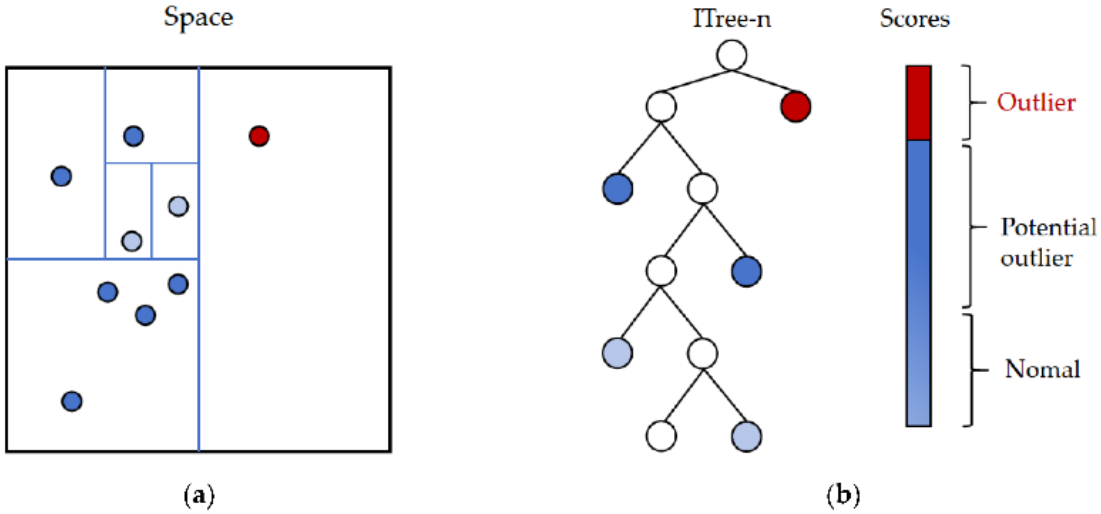


Figure 3. How Isolation Forest Works

The average depth of isolation across the tree was then used to calculate an anomaly score for each sample [24]. Key components in the mathematical formulation included in Eq. (10):

$$s(x,n) = 2^{-\frac{E\,|\,h\,(x)|}{c\,(n)}}. \tag{10}$$

The Isolation Tree (iTree) recursively partitions the data by randomly selecting a feature and choosing a split value within that feature's range. The path length $h(x)$ represents the number of splits required to isolate a sample x, where shorter paths typically indicate a higher likelihood of anomaly. The anomaly score $s(x,n)$, ranging from 0 (normal) to 1 (anomalous), is computed from the average path length across $n$samples and then normalized to produce the final score.

2.5 KNN Imputer

Data comes from various sources and is used for analysis, insight gathering, theory verification, and other purposes. However, it is often found that some attributes in a data set are empty due to human error or technical problems during extraction and collection, making the handling of missing values a crucial stage in data preprocessing. The choice of imputation method is important because it significantly affects the performance of the model being built. As an alternative to conventional imputation techniques, K-Nearest Neighbors (KNN) Imputer in the scikit-learn library can be used to fill in missing values by finding the most similar neighboring observations based on the Euclidean distance matrix [25], [26]. In its calculation, missing values are ignored so that distances are only calculated on available coordinates, meaning that non-missing dimensions are practically given greater weight. Mathematically, the Euclidean distance between two observations x and y that only considers non-missing features is expressed in the Eqs. (11) and (12) as follows.

$$D_{xy} = \sqrt{W * SqD}, \tag{11}$$

where :

$$W = \frac{T}{N}. \tag{12}$$

The parameters used in this formulation are defined as follows, where Wrepresents the weight, while SqD denotes the squared distance from the current coordinate. The term T refers to the total value associated with the coordinate, and Nindicates the number of coordinates currently considered.

2.6 Mean Squared Error (MSE)

Mean Squared Error (MSE) is an evaluation metric that calculates the average square difference between the predicted value and the actual value across all data. By squaring each difference, MSE gives greater weight to larger errors, so that models with low MSE values are considered more accurate overall.

Because errors are squared before being averaged, MSE is sensitive to outliers and penalizes large deviations more heavily. MSE can be calculated using the following mathematical Eq. (13).

$$\boldsymbol{MSE} = \frac{\mathbf{1}}{\boldsymbol{N}}\sum_{\boldsymbol{i=1}}^{\boldsymbol{N}}(\boldsymbol{y_i} - \hat{\mathbf{y}}_{\boldsymbol{i}})^{\mathbf{2}}. \tag{13}$$

y_i represents the actual value of sample i, $\hat{y}_i$ denotes the predicted value of sample i, and N is the total number of samples. The Mean Squared Error (MSE) is calculated by taking the difference between each actual value y_i and its corresponding prediction $\hat{y}_i$, squaring this difference to ensure all values are positive and to give larger errors greater weight. All squared differences are then summed and divided by the total number of observations N. Thus, MSE emphasizes large deviations in predictions and reflects the overall average error of the model.

2.7 Root Mean Squared Error (RMSE)

Root Mean Squared Error (RMSE) is a complementary evaluation metric derived by taking the square root of MSE, returning the error to the same unit as the original data (mm), which makes it more interpretable in the context of rainfall prediction. Like MSE, RMSE penalizes larger errors more heavily, but its scale consistent output allows for a more intuitive assessment of prediction accuracy. RMSE can be calculated using the following mathematical Eq. (14).

$$\boldsymbol{RMSE} = \sqrt{\frac{\mathbf{1}}{\boldsymbol{N}}\sum_{\boldsymbol{i=1}}^{\boldsymbol{N}}(\boldsymbol{y_i} - \hat{\mathbf{y}}_{\boldsymbol{i}})^{\mathbf{2}}}, \tag{14}$$

y_i represents the actual value of sample i, $\hat{y}_i$denotes the predicted value of sample i, and N is the total number of samples. A lower RMSE value indicates that the predicted values are closer to the actual observations, reflecting higher model accuracy.

3. RESULTS AND DISCUSSION

3.1 Data Exploration

In this study, rainfall data from the BMKG was used, consisting of 3633 rows and 11 columns taken from 1 January 2015 to 31 December 2024. Table 2 below shows the final results of the cleaned data.

Table 2. Data After Cleaning

No	Variable	Data Type	Unit
1	TANGGAL	datetime64	Date type
2	TN	float64	Degrees Celsius (°C)
3	TX	float64	Degrees Celsius (°C)
4	TAVG	float64	Degrees Celsius (°C)
5	RH_AVG	float64	Percent (%)
6	RR	float64	Millimeters (mm)
7	SS	float64	Hours
8	FF_X	int64	Meters per second (m/s)
9	DDD_X	int64	Degrees (°)
10	FF_AVG	int64	Meters per second (m/s)

Based on Table 2, the final data type for each variable consists of datetime64 for the date column, float64 for the TN, TX, TAVG, RH_AVG, RR, and SS columns, and int64 for the FF_X, DDD_X, and FF_AVG columns. The DDD_CAR column is not used because the DDD_CAR variable represents the dominant wind direction in the form of wind categories, but this information is already covered in DDD_X, which uses a degree representation. With other wind variables such as FF_X and FF_AVG that are more informative in describing wind dynamics, DDD_CAR becomes redundant and does not provide additional predictive value for rainfall modelling. In addition, its categorical nature has the potential to add complexity through the encoding process without commensurate benefits. The removal of this variable is considered more appropriate to maintain the simplicity and stability of the model.

Table 3. Missing Value in Dataset

Column	Missing Value
TANGGAL	0
TN	484
TX	220
TAVG	40
RH_AVG	42
RR	1222
SS	256
FF_X	0
DDD_X	0
FF_AVG	0

The analysis of missing values in Table 3 above shows variations in the amount of missing data for each variable, with rainfall (RR) being the most problematic with 1,222 missing data points, requiring special attention as it is the main variable of the study. In addition, the temperature variable also experienced data loss, with minimum temperature (TN) losing 484 data points and maximum temperature (TX) losing 220 data points, while sunshine duration (SS) recorded 256 missing values, and average temperature (TAVG) and average humidity (RH_AVG) lost 40 and 42 values, respectively. Conversely, FF_X, DDD_X, and FF_AVG had no missing values and could be considered consistent throughout the observation period. In addition to these missing values, this dataset also contains complete duplicate records that require deletion or handling. A brief examination found identical rows on 10 July 2021, 7 August 2021, and 18 September 2021 (same values in all variables such as temperature, humidity, and wind speed), such complete duplicates can affect statistical analysis and machine learning models by giving excessive weight to repeated observations.

Table 4. Descriptive Statistics

Variable	count	mean	std	min	25%	50%	75%	max
TN	3149	24.76	1.06	20	24	25	25	27.7
TX	3413	30.84	1.52	24	30	31	32	35.9
TAVG	3593	27.26	1.13	23	27	27	28	31.3
RH_AVG	3591	83.33	3.57	66	81	83	86	97
RR	2411	12.63	23.8	0	0	2.6	14	356
SS	3377	6.31	2.95	0	4.3	7	8.7	11.9
FF_X	3633	4.63	1.56	1	4	4	5	21
DDD_X	3633	162.7	62.3	0	120	140	210	360
FF_AVG	3633	1.95	0.95	0	1	2	2	6

Table 4 shows that rainfall (RR) exhibits high variability, with a mean of 12.63 mm, a large standard deviation of 23.82 mm, and a maximum value of 356 mm, indicating the presence of extreme events relevant to flood and landslide risks. In contrast, temperature variables demonstrate more stable distributions where the minimum temperature (TN) averages is 24.76 °C, maximum temperature (TX) averages is 30.84 °C, and mean temperature (TAVG) averages is 27.26 °C, all with relatively narrow ranges typical of a warm tropical climate. Average humidity (RH_AVG) is also stable, with a mean of 83.33% and a range of 66–97%, reflecting consistently high humidity throughout the year. Compared to these supporting atmospheric variables, rainfall remains the most dynamic and influential indicator for detecting potential extreme events.

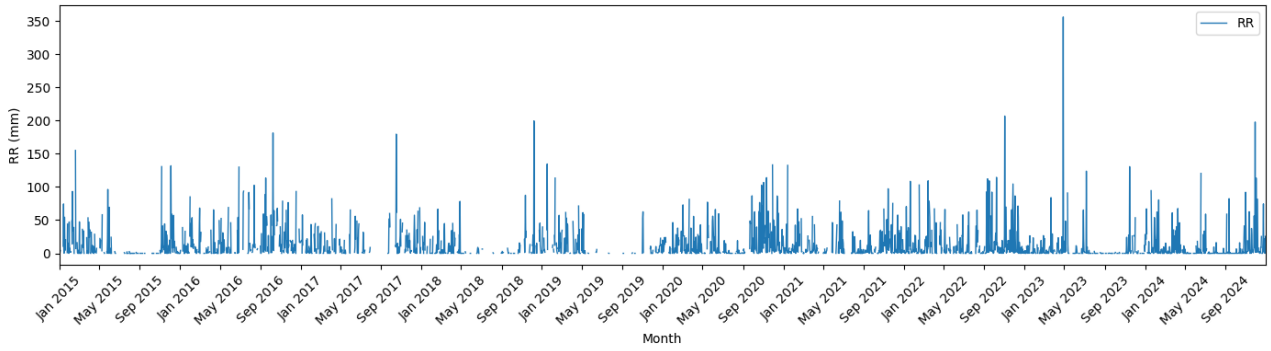
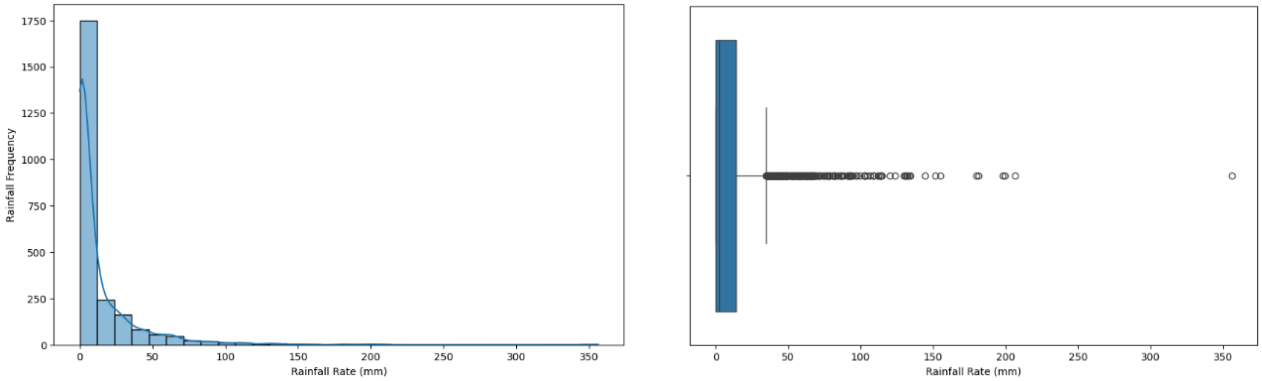


Figure 4. RR Variable Time Series Graph

Fig. 4 shows the daily rainfall (RR) time series for the study period (see Methods). The series exhibits clear seasonal variation, with increased rainfall during the wet months and pronounced reductions during the dry season. Several isolated spikes exceed typical seasonal levels, indicating extreme rainfall events that may be associated with hydrological hazards such as flooding and landslides. These features highlight the need for further anomaly analysis and risk-based interpretation.



(a) (b)

Figure 5. RR Data Visualization : (a) Histogram of RR; (b) Boxplot of RR

Fig. 5 presents the distribution of daily rainfall (RR) using a histogram (left) and a boxplot (right). The histogram demonstrates a strong right-skew: most days record little or no rainfall while a small number of days exhibit high-intensity precipitation. The boxplot corroborates this asymmetry, with the majority of observations clustered in the 0–20 mm range and numerous high-value outliers extending above 100 mm and up to over 300 mm. These outliers represent extreme rainfall events that are informative for anomaly detection and hazard assessment, therefore they are retained for further analysis rather than being removed as noise.

3.2 Preprocessing Data

The rainfall variable (RR) had the highest proportion of missing values, with 1,222 missing entries out of 3,633 observations. KNN Imputer was selected over simpler alternatives such as mean or median imputation because rainfall is a temporally correlated variable, where values on nearby days tend to be similar due to persistent weather systems. Mean or median imputation ignores this temporal structure and risks distorting the natural variability of extreme events, whereas KNN Imputer estimates each missing value from the 20 most similar neighboring observations, thereby preserving local temporal patterns and the spread of the rainfall distribution. To assess the effect of imputation, the statistical properties of RR before and after imputation were compared. Prior to imputation, the 2,411 available RR observations had a mean of 12.63 mm and a standard deviation of 23.82 mm. After KNN imputation across all 3,633 observations, the mean slightly decreased to 11.16 mm and the standard deviation to 19.97 mm, reflecting the inclusion of previously missing low-rainfall days. Despite this modest shift, the right-skewed nature of the data and the presence of extreme values were retained, indicating that the imputation did not distort the overall distributional characteristics of rainfall in Cilacap. The imputation results are shown in Table 5.

Table 5. Data After Imputation

TANGGAL	TN	TX	TAVG	RH_AVG	RR	SS	FF_X	DDD_X	FF_AVG
1/1/2015	25.01	32.22	27.7	82	5.03	6.07	7	240	3
1/2/2015	24.86	31.8	27.9	83	6.74	4.2	7	260	3
1/3/2015	25.13	32.39	27.5	80	0.8	5.31	5	250	2
1/4/2015	25	30.4	27.3	82	4.2	4.46	6	230	2
1/1/2015	25.01	32.22	27.7	82	5.03	6.07	7	240	3

This method fills missing values by leveraging similarity between data rows. Each incomplete record is compared to other complete rows to find the 20 nearest neighbors using Euclidean distance, and the missing value is estimated as the average of those neighbors. This approach produces more accurate and consistent imputations than simple methods such as mean or median filling, while preserving the stability of the data distribution prior to modeling. After imputation, input sequences of length 15 were generated, where each input sample represents rainfall from the previous 15 days and the target is the value on day 16. This was done using a sliding-window approach, and the resulting data were reshaped into a three-dimensional format (samples, timesteps, features) to match the input requirements of the LSTM model.

Table 6. Data After Noise Addition

No	Date	Rainfall Noise (RR)	Real Rainfall (RR)
1	1/1/2015	[4.958015579, 4.881939763, 5.074543624, 5.173489782, 4.906999915]	5.03
2	2/1/2015	[6.687082576, 6.823292245, 6.802479813, 6.688271817, 6.757360908]	6.735

The technique in Table 6 involved adding random noise to each sample, a data augmentation technique was applied by adding Gaussian noise to the training samples only, prior to model training. Each sequence in the training set was replicated five times, with independent random noise drawn from a normal distribution ($\mu = 0$, $\sigma = 0.1$) added to the rainfall variable (RR) in each replication. This process expanded the training dataset from its original size to 21,690 augmented entries, increasing data variability while preserving the underlying temporal patterns of the original observations. To maintain the temporal integrity of the time series, the dataset was split chronologically. The first 80% of the sequential samples, corresponding to the period from 1 January 2015 to 30 November 2022, were used for training (approximately 17,352 samples after augmentation), while the remaining 20%, from 1 December 2022 to 31 December 2024, were reserved for testing (approximately 4,338 samples). This method ensures that the model is always trained on past observations and evaluated on future data, preventing any form of data leakage.

3.3 LSTM Modelling

Although the dataset contains multiple meteorological variables, the LSTM model in this study was configured as a univariate model using only the rainfall variable (RR) as both the input predictor and the prediction target. The remaining variables (TN, TX, TAVG, RH_AVG, SS, FF_X, DDD_X, and FF_AVG) were retained during preprocessing solely to support KNN imputation of missing RR values, and were not included as model inputs. In this study, the network was constructed using a set of input variables in accordance with the features used in modelling, and produced a single output in the form of rainfall prediction values. Experiments were conducted by exploring various combinations of hyperparameters to obtain the most optimal model configuration. The combinations of hyperparameters tested included in Table 7.

Table 7. Hyperparameter for LSTM

Model Component	Parameter	Parameter Value
LSTM Layer	Units	20, 30, 50
	Activation	tanh, relu, sigmoid
Dropout Layer 1	Rate	0.3, 0.5
Dense Layer	Units	20, 30, 50
	Activation	tanh, relu, sigmoid
Dense Layer 2	Units	1

The results of the evaluation of each hyperparameter combination are then summarized in Table 8 as the basis for selecting the model configuration used in this study.

Table 8. Testing with Parameter Combinations

No	Units	Activation	Dropout	Dense1U	Dense1A	Dense2U	MSE Val	RMSE Val
1	20	Tanh	0.3	20	relu	1	183.45386	13.5445
2	20	Tanh	0.3	20	sigmoid	1	242.38417	15.5687
3	20	Tanh	0.5	20	relu	1	192.38079	13.8701
4	20	Tanh	0.5	20	sigmoid	1	245.65683	15.6734
5	20	Relu	0.3	20	tanh	1	360.70067	18.9921
6	20	Relu	0.3	20	sigmoid	1	358.42557	18.9321
7	20	Relu	0.5	20	tanh	1	364.81505	19.1001
8	20	Relu	0.5	20	sigmoid	1	399.71894	19.9930
9	20	Sigmoid	0.3	20	tanh	1	288.80337	16.9942
10	20	Sigmoid	0.3	20	relu	1	376.87201	19.4132
11	20	Sigmoid	0.5	20	tanh	1	418.87928	20.4665
12	20	Sigmoid	0.5	20	relu	1	418.04226	20.4461
13	30	Tanh	0.3	30	relu	1	129.27861	11.3701
14	30	Tanh	0.3	30	sigmoid	1	97.393716	9.8688
15	30	Tanh	0.5	30	relu	1	103.23621	10.1605
16	30	Tanh	0.5	30	sigmoid	1	133.75783	11.5654
17	30	Relu	0.3	30	tanh	1	203.03374	14.2490
18	30	Relu	0.3	30	sigmoid	1	389.62757	19.7390

No	Units	Activation	Dropout	Dense1U	Dense1A	Dense2U	MSE Val	RMSE Val
19	30	Relu	0.5	30	tanh	1	312.1461	17.6677
20	30	Relu	0.5	30	sigmoid	1	278.32504	16.6831
21	30	Sigmoid	0.3	30	tanh	1	321.56134	17.9321
22	30	Sigmoid	0.3	30	relu	1	224.2029	14.9734
23	30	Sigmoid	0.5	30	tanh	1	329.59471	18.1547
24	30	Sigmoid	0.5	30	relu	1	313.35235	17.7018
25	***50***	***Tanh***	***0.3***	***50***	***relu***	***1***	***23.52472***	***4.8502***
26	50	Tanh	0.3	50	sigmoid	1	26.521545	5.1499
27	50	Tanh	0.5	50	relu	1	38.339212	6.1919
28	50	Tanh	0.5	50	sigmoid	1	41.290225	6.4257
29	50	Relu	0.3	50	tanh	1	205.61332	14.3392
30	50	Relu	0.3	50	sigmoid	1	172.41543	13.1307
31	50	Relu	0.5	50	tanh	1	183.49828	13.5462
32	50	Relu	0.5	50	sigmoid	1	99.268196	9.9633
33	50	Sigmoid	0.3	50	tanh	1	254.72413	15.9601
34	50	Sigmoid	0.3	50	relu	1	218.73544	14.7897
35	50	Sigmoid	0.5	50	tanh	1	221.76198	14.8917
36	50	Sigmoid	0.5	50	relu	1	186.38125	13.6522

The *Unit* parameter refers to the number of neurons in the LSTM layer, while *Activation* denotes the activation function applied within that layer. The *Dropout* value represents the proportion of neurons temporarily deactivated during training to reduce overfitting. The *Dense1U* parameter indicates the number of units in the first Dense layer following the LSTM, and *Dense1A* specifies the activation function used in that Dense layer. Meanwhile, *Dense2U* corresponds to the number of units in the second Dense layer, which serves as the output layer. Finally, the *MSE* and *RMSE Val* value reflects the Mean Squared Error obtained on the validation dataset, functioning as a key metric for evaluating model performance.

Based on Table 8, the best hyperparameter configuration consists of an LSTM layer with 50 units and a tanh activation, a dropout rate of 0.3, followed by a dense layer with 50 ReLU units, and a single output node. This setup achieved the lowest validation MSE of 23.5247161 and RMSE 4.8502, indicating an optimal balance between model capacity and generalization. The 50 LSTM units were sufficient to capture temporal rainfall patterns without causing overfitting, while the tanh activation allowed richer internal representations. The 0.3 dropout provided effective regularization, and the 50-unit ReLU dense layer supported efficient nonlinear mapping. Using the Adam optimizer with a 0.001 learning rate also contributed to stable training. Overall, this configuration delivered the most reliable performance for modeling rainfall in this study.

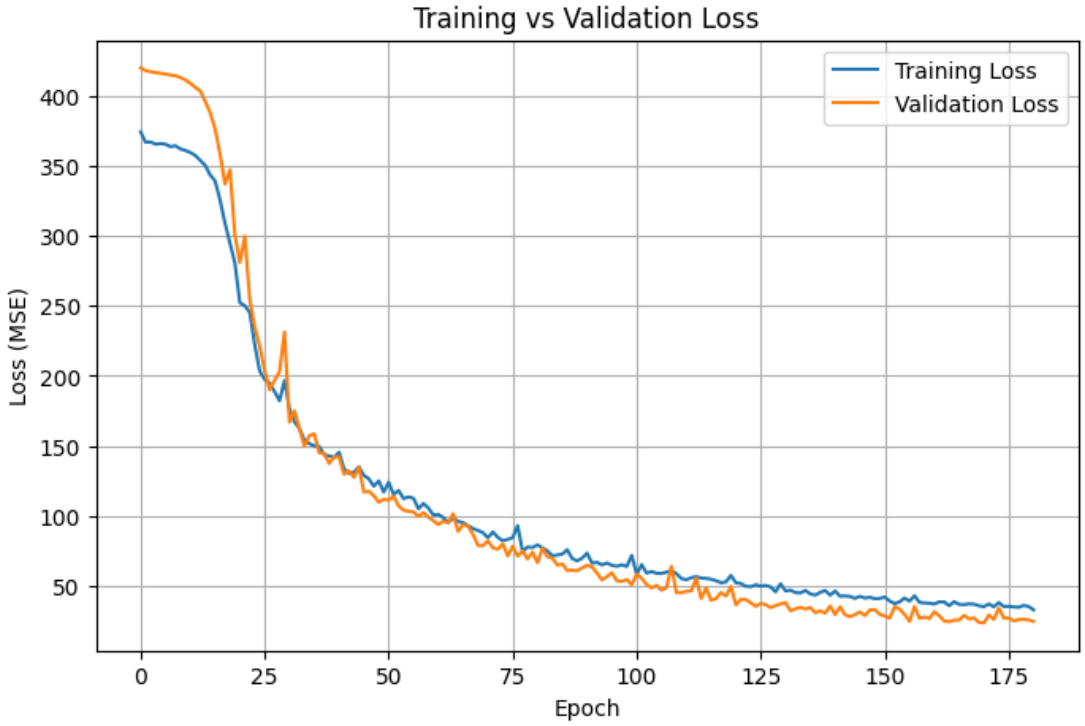


Figure 6. Learning Curve LSTM

Fig. 6 shows a learning curve with a sharp decrease in loss during the early training phase, dropping from around 400 MSE at epoch 1 to much lower values within the first few dozen epochs. Both the training and validation curves decline in parallel and begin to stabilize between epochs 20 and 40, gradually converging toward a final loss of approximately 25–35 MSE by epoch 180. Although minor fluctuations appear in the validation curve around epochs 20–30, the two curves remain close without any clear divergence. This pattern indicates that the model achieves a good balance between learning and generalization, with no significant signs of overfitting, making it reliable for producing consistent predictions on unseen data.

3.4 Final Result

Isolation Forest was selected as the anomaly detection algorithm because it does not require assumptions about the underlying data distribution and scales efficiently to large datasets. Unlike density-based methods such as Local Outlier Factor (LOF) or clustering-based approaches such as DBSCAN, its isolation-based mechanism is particularly well suited for rainfall data, which is highly right-skewed and contains rare but extreme values that are structurally distinct from normal observations. This characteristic makes anomalous days easier to isolate with fewer tree splits. Furthermore, the use of Isolation Forest in combination with LSTM for rainfall anomaly detection has been validated in previous studies [16], who successfully applied the same hybrid approach to identify flood-prone rainfall events in Semarang. The contamination parameter was set to 0.05, indicating that approximately 5% of the predicted daily values were expected to be anomalous. This value was selected based on the historical rainfall distribution of Cilacap, where extreme rainfall events at or above the 95th percentile are commonly associated with flood and landslide hazards, and is consistent with prior studies applying Isolation Forest to hydrometeorological data, The input to the Isolation Forest model consisted solely of the predicted daily rainfall values (RR) generated by the LSTM model for the 365-day forecast period of January to December 2025, as the primary objective was to identify days with anomalously high predicted rainfall that may indicate potential extreme weather events.

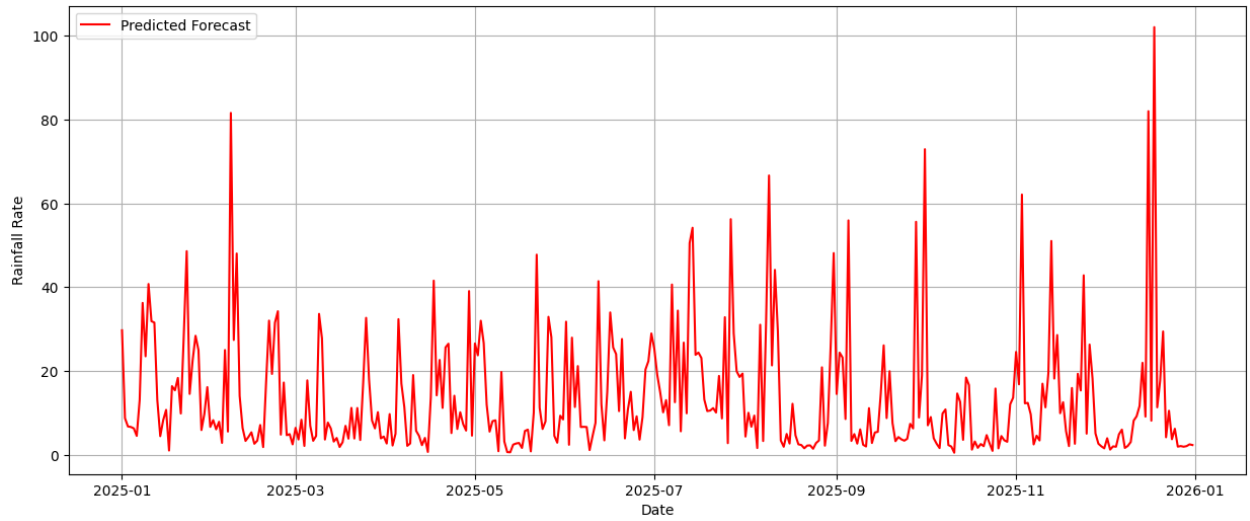


Figure 7. LSTM Prediction Results

Through a combination of LSTM and Isolation Forest methods, the forecasting results can be interpreted directly, so that prediction values that deviate from the general pattern can be recognized as indications of critical events, such as extreme rainfall that has the potential to cause flooding. Fig. 7 shows the predicted daily rainfall from January 2025 to January 2026, revealing clear seasonal patterns with daily fluctuations and several extreme peaks. Rainfall is high from January to March, with extreme spikes around 60–100 mm/day, followed by a notable decrease from April to June with some moderate peaks typical of the transition season. July to September is dominated by very low values, reflecting the dry season, while rainfall increases again from October to December with peaks reaching 60–90 mm/day. The LSTM model successfully captures these seasonal rhythms and daily variations, although the predicted extreme values require further validation and anomaly analysis.

After identifying the seasonal pattern, the next step is to detect extreme events using the Isolation Forest anomaly detection algorithm.

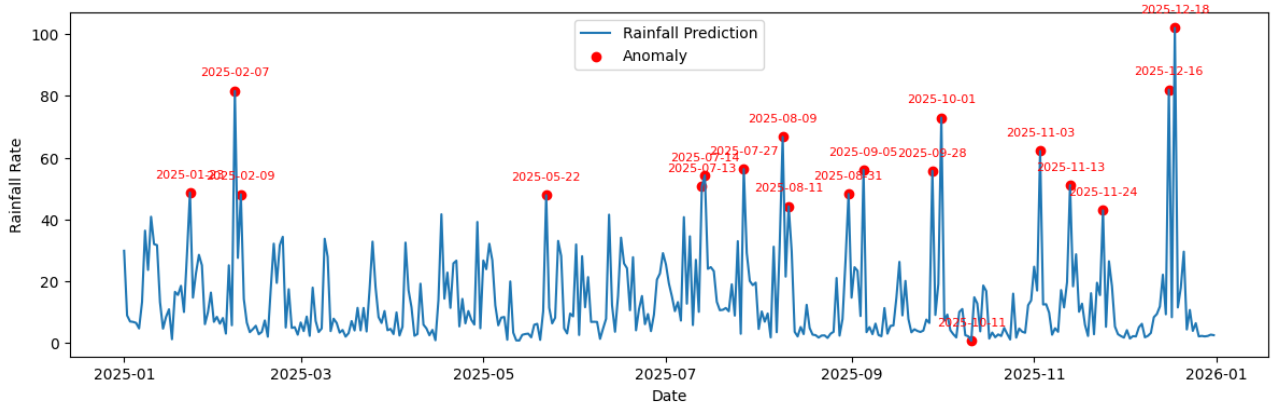


Figure 8. Rainfall Prediction Results and Anomalies

Fig. 8 presents the predicted daily rainfall for 365 days along with anomaly points detected using the Isolation Forest algorithm with a contamination rate of 0.05, indicating that around 5% of days are classified as anomalies. These anomalies do not appear uniformly but form clusters at the beginning, middle, and end of the year, coinciding with periods of extreme rainfall approaching 100 mm/day. This pattern suggests multiple intervals with a high potential for significant weather events, including flood risk due to intense rainfall. Although the number of anomalies aligns with the model parameters, some points may reflect over-sensitive detection and should be interpreted cautiously to account for uncertainty and possible false positives. The results serve as an initial indicator for risk assessment, mitigation planning, and early warning system development, while acknowledging the model's limitations. Further validation will be conducted using rainfall data and actual disaster records from the Regional Disaster Management Agency (BPBD) to evaluate the accuracy of the detections listed in Table 9.

Table 9. Cilacap Natural Disaster Management Agency Data

No	ID	District / City	Type of Disaster	Time of Occurrence
1	3.30119E+15	Cilacap	Extreme Weather	2025-10-13 00:00 WIB
2	3.3011E+15	Cilacap	Flood	2025-10-12 00:00 WIB
3	3.3011E+15	Cilacap	Flood	2025-05-24 23:00 WIB
4	3.30119E+15	Cilacap	Extreme Weather	2025-04-10 13:45 WIB
5	3.3011E+15	Cilacap	Landslide	2025-04-09 09:24 WIB
6	3.3011E+15	Cilacap	Flood	2025-04-08 09:57 WIB
7	3.30119E+15	Cilacap	Extreme Weather	2025-04-02 14:20 WIB
8	3.30119E+15	Cilacap	Extreme Weather	2025-03-12 15:00 WIB
9	3.3011E+15	Cilacap	Landslide	2025-02-19 16:56 WIB
10	3.30119E+15	Cilacap	Extreme Weather	2025-02-09 11:03 WIB
11	3.30119E+15	Cilacap	Extreme Weather	2025-02-05 11:30 WIB
12	3.3011E+15	Cilacap	Flood	2025-01-27 20:30 WIB
13	3.3011E+15	Cilacap	Flood	2025-01-10 09:31 WIB

Data source: (BPBD JatengProv - Cevadis)

The comparison between the 19 anomaly dates detected by the rainfall prediction model and the 13 reported events in the BPBD Cilacap dataset shows several meaningful alignments as well as temporal gaps that likely represent early signals of unreported potential events. Notable matches include anomalies on 28 January 2025 near the flood report on 27 January, 7 February near the extreme weather report on 5 February, an exact match on 9 February, 22 May close to the flood event on 24 May, and 11 October occurring one to two days before the flood/extreme weather events on 12–13 October. These matches indicate that the model can capture rainfall spikes associated with reported hazards. Some anomalies, especially in July until September, parts of November, and mid-December do not directly match BPBD entries, yet may still represent early warning signals, as floods, landslides, and extreme weather often result from accumulated rainfall several days before or after an event. Thus, these unmatched anomalies may reflect precursor conditions or residual effects that remain relevant for hazard assessment. Our findings align strongly with established literature across both prediction and anomaly detection methodologies. The LSTM network performance confirms previous research by effectively modeling nonlinear and extreme rainfall, supporting its utility for short-term hydrometeorological forecasting in areas with high variability. Similarly, the Isolation Forest anomaly-detection results are consistent with earlier studies showing that tree-based detectors

can highlight extreme values and precursor conditions. While this confirms Isolation Forest's sensitivity, the findings reinforce the need for complementary indicators as noted in prior research to improve interpretability and ensure the reliability of warnings.

4. CONCLUSION

Based on the research process and discussion, several conclusions can be drawn. This study identified an optimal LSTM-based model for rainfall prediction in Cilacap that effectively learns historical rainfall patterns: an LSTM layer with 50 units and tanh activation, a dropout of 0.3, followed by a dense layer of 50 units with ReLU activation and a single output layer, which produced a validation MSE of 23.5247161 and RMSE 4.8502. The predictive model is complemented by an Isolation Forest–based anomaly detection module to group extreme values and identify days with anomalously high predicted rainfall that may be associated with floods, landslides, and extreme weather events, anomalies were detected on 28 January 2025 (close to the 27 January 2025 flood report), 7 February 2025 (within ±2 days of the 5 February 2025 extreme weather report), 9 February 2025 (coinciding with the 9 February 2025 extreme weather report), 22 May 2025 (close to the 24 May 2025 flood report), and 11 October 2025 (several days before the 12–13 October 2025 flood/extreme weather reports). These temporal alignments indicate a correlation between detected anomalies and observed flood/extreme weather events, suggesting that the system has practical potential to support monitoring and early-warning functions.

Author Contributions

Adhystira Raihannoeza Almadiva: Conceptualization, methodology, data curation, formal analysis, writing original draft preparation. Atika Ratna Dewi: Software, resources, visualization, validation. Aina Latifa Riyana Putri: Investigation, project administration, funding acquisition. All authors have read and agreed to the published version of the manuscript, discussed the results together, and contributed to the refinement of the final draft.

Funding Statement

The authors would like to express their sincere gratitude to Telkom University Purwokerto Campus for funding and supporting this research.

Acknowledgment

The authors wish to express their sincere appreciation to Telkom University Purwokerto Campus and the Data Science Study Program for their institutional support. We also thank the Meteorological, Climatological and Geophysical Agency (BMKG) and the Regional Disaster Management Agency (BPBD) for making publicly accessible datasets available to researchers. Additional thanks go to colleagues and reviewers who provided valuable feedback and assistance during the preparation of this manuscript.

Declarations

The authors declare that there is no conflict of interest related to this article

Declaration of Generative AI and AI-assisted technologies

The authors declare that no generative AI or AI-assisted technologies were used in the preparation of this manuscript, including for writing, editing, data analysis, or the creation of tables and figures.

REFERENCES

[1] B. Susilo, *MENGENAL IKLIM DAN CUACA DI INDONESIA*. DIVA PRESS, 2021. [Online].

[2] A. M. Siregar, S. Faisal, Y. Cahyana, and B. Priyatna, "PERBANDINGAN ALGORITME KLASIFIKASI UNTUK PREDIKSI CUACA," *Jurnal Accounting Information System (AIMS)*, vol. 3, no. 1, p. 15, 2020, doi: https://doi.org/10.32627/aims.v3i1.280.

[3] A. M. Siregar, Tukino, S. Faisal, A. Fauzi, and I. Kadori, "KLASIFIKASI UNTUK PREDIKSI CUACA MENGGUNAKAN ESEMBLE LEARNING," *Jurnal Pengkajian dan Penerapan Teknik Informatika (PETIR)*, vol. 13, no. 2, pp. 138–147, Sep. 2020, doi: https://doi.org/10.33322/petir.v13i2.998.

[4] N. R. Aswarisman, A. S. Handayani, and I. Hadi, "PENERAPAN ALGORITMA RANDOM FOREST UNTUK MEMPREDIKSI CURAH HUJAN PADA MASA MENDATANG DI DAERAH BERPOTENSI BANJIR," *Building of Informatics, Technology and Science (BITS)*, vol. 6, no. 2, pp. 1222–1230, Sep. 2024, doi: https://doi.org/10.47065/bits.v6i2.5593.

[5] M. Taqiyuddin and T. Bayu Sasongko, "PREDIKSI CUACA KABUPATEN SLEMAN MENGGUNAKAN ALGORITMA RANDOM FOREST," *Jurnal Media Informatika Budidarma*, vol. 8, no. 3, p. 1683, Jul. 2024, doi: https://doi.org/10.30865/mib.v8i3.7897.

[6] R. Torhino and P. N. Andono, "PENERAPAN ALGORITMA RANDOM FOREST DALAM PREDIKSI CURAH HUJAN UNTUK MENDUKUNG ANALISIS CUACA," *Building of Informatics, Technology and Science (BITS)*, vol. 6, no. 3, pp. 1688–1699, Dec. 2024, doi: https://doi.org/10.47065/bits.v6i3.6404.

[7] J. Vieria dos Santos, V. Farias Felipe, E. Fialho Morais Xavier, T. Almeida de Oliveira, J. Silva Jale, and S. F. Alves Xavier Junior, "A STUDY OF THE STANDARDIZED PRECIPITATION INDEX (SPI) AND MACHINE LEARNING TECHNIQUES FOR DROUGHT PREDICTION IN THE STATE OF PARAÍBA, BRAZIL," *Revista Científica Multidisciplinar (RECIMA21)*, vol. 5, no. 10, p. e5105736, Oct. 2024, doi: https://doi.org/10.47820/recima21.v5i10.5736.

[8] A. Adane and B. Asmerom, "ANALYSIS OF SPATIOTEMPORAL DISTRIBUTION, VARIABILITY, AND TRENDS OF RAINFALL IN WOLLO AREA, NORTHEASTERN ETHIOPIA," *PLoS One*, vol. 20, no. 1, Jan2025, doi: https://doi.org/10.1371/journal.pone.0312889.

[9] E. Zaveri, J. Russ, and R. Damania, "RAINFALL ANOMALIES ARE A SIGNIFICANT DRIVER OF CROPLAND EXPANSION," *Proceedings of the National Academy of Sciences (PNAS)*, Mar. 2020, doi: https://doi.org/10.1073/pnas.1910719117.

[10] K. Alaskar Bharati Vidyapeeth, D. Patil, and K. Alaskar, "ENHANCED RAINFALL FORECASTING: ETR, MLP, AND XGB MODELS ANALYSED WITH ADVANCED OPTIMIZATION," *Journal of the Maharaja Sayajirao University of Baroda*, vol. 58, no. 2, pp. 25–0422, Jan. 2024, [Online].

[11] M. Waqas and U. W. Humphries, "A CRITICAL REVIEW OF RNN AND LSTM VARIANTS IN HYDROLOGICAL TIME SERIES PREDICTIONS," *MethodsX*, vol. 13, Dec. 2024, doi: https://doi.org/10.1016/j.mex.2024.102946.

[12] I. Malashin, V. Tynchenko, A. Gantimurov, V. Nelyub, and A. Borodulin, "APPLICATIONS OF LONG SHORT-TERM MEMORY (LSTM) NETWORKS IN POLYMERIC SCIENCES: A REVIEW," *Polymers (Basel)*, vol. 18, no. 16, pp. 1–44, Sep. 2024, doi: https://doi.org/10.3390/polym16182607.

[13] Zulfikar Ahmad, F. A. Rahmani, and N. Azizah, "DETEKSI ANOMALI MENGGUNAKAN ISOLATION FOREST BELANJA BARANG PERSEDIAAN KONSUMSI PADA SATUAN KERJA KEPOLISIAN REPUBLIK INDONESIA," *Jurnal Manajemen Perbendaharaan*, vol. 4, no. 1, pp. 1–15, Jun. 2023, doi: https://doi.org/10.33105/jmp.v4i1.435.

[14] Y. Hendra, H. Mukhtar, and R. Hafsari, "PREDIKSI CURAH HUJAN DI KOTA PEKANBARU MENGGUNAKAN LSTM (LONG SHORT TERM MEMORY)," *Jurnal Software Engineering and Information System (SEIS)*, vol. 3, no. 2, pp. 74–81, Aug. 2023.doi : https://doi.org/10.37859/seis.v3i2.5606.

[15] M. Musfiroh, D. C. R. Novitasari, P. K. Intan, and G. G. Wisnawa, "PENERAPAN METODE PRINCIPAL COMPONENT ANALYSIS (PCA) DAN LONG SHORT-TERM MEMORY (LSTM) DALAM MEMPREDIKSI PREDIKSI CURAH HUJAN HARIAN," *Building of Informatics, Technology and Science (BITS)*, vol. 5, no. 1, Jun. 2023, doi: https://doi.org/10.47065/bits.v5i1.3114.

[16] A. Wijayanto, A. Sugiharto, and R. Santoso, "IDENTIFIKASI DINI CURAH HUJAN BERPOTENSI BANJIR MENGGUNAKAN ALGORITMA LONG SHORT-TERM MEMORY (LSTM) DAN ISOLATION FOREST," *Jurnal Teknologi Informasi dan Ilmu Komputer*, vol. 11, no. 3, pp. 637–646, Jul. 2024, doi: https://doi.org/10.25126/jtiik.938718.

[17] T. Lattifia, P. Wira Buana, N. Kadek, and D. Rusjayanthi, "MODEL PREDIKSI CUACA MENGGUNAKAN METODE LSTM," *JITTER-Jurnal Ilmiah Teknologi dan Komputer*, vol. 3, no. 1, Apr. 2022.doi : https://doi.org/10.24843/JTRTI.2022.v03.i01.p35

[18] P. Sugiartawan and S. Gusprio Santoso, "MULTIVARIATE FORECASTING CURAH HUJAN MENGGUNAKAN ALGORITMA LSTM DI KOTA DENPASAR," in *Seminar Nasional Corisindo*, Aug. 2022, pp. 580–585.

[19] F. Ramadhan and M. Fachrie, "IMPLEMENTASI ALGORITMA LONG SHORT-TERM MEMORY (LSTM) PADA SISTEM PREDIKSI HASIL PANEN SAWIT," *Jurnal Informatika Teknologi dan Sains*, vol. 6, no. 4, pp. 937–944, Oct. 2024.doi: https://doi.org/10.51401/jinteks.v6i4.4876

[20] Roni Merdiansah, Khofifah Wulandari, Mentari Hasibuan, and Yuyun Umaidah, "PERBANDINGAN KINERJA MODEL RNN, LSTM, DAN BLSTM DALAM MEMPREDIKSI JUMLAH GEMPA BULANAN DI INDONESIA," *Jurnal Penelitian Rumpun Ilmu Teknik*, vol. 3, no. 1, pp. 262–277, Feb. 2024, doi: https://doi.org/10.55606/juprit.v3i1.3466.

[21] A. A. Ningrum, I. Syarif, A. I. Gunawan, E. Satriyanto, and R. Muchtar, "ALGORITMA DEEP LEARNING-LSTM UNTUK MEMPREDIKSI UMUR TRANSFORMATOR," *Jurnal Teknologi Informasi dan Ilmu Komputer*, vol. 8, no. 3, pp. 539–548, Jun. 2021, doi: https://doi.org/10.25126/jtiik.2021834587.

[22] Y. Kong *et al.*, "UNLOCKING THE POWER OF LSTM FOR LONG TERM TIME SERIES FORECASTING," *ArXiv*, Feb. 2025, [Online].

[23] G. Airlangga, "ADVANCED MACHINE LEARNING TECHNIQUES FOR SEISMIC ANOMALY DETECTION IN INDONESIA: A COMPARATIVE STUDY OF LOF, ISOLATION FOREST, AND ONE-CLASS SVM," *Jurnal Ilmiah Pendidikan Matematika, Matematika dan Statistika (Lebesgue)*, vol. 5, no. 1, Apr. 2024, doi: https://doi.org/10.46306/lb.v5i1.

[24] D. B. Santoso and Y. Wahyuni, "SESTEM LOG WEB SERVER SEBAGAI PENDETEKSI ANOMALI MENGGUNAKAN ISOLATION FOREST," *Jurnal Aplikasi Bisnis dan Komputer (JUBIKOM)*, vol. 4, no. 3, pp. 2807–5986, Oct. 2024, [Online] doi: https://doi.org/10.33751/jubikom.v4i3.10941

[25] A. Juna *et al.*, "WATER QUALITY PREDICTION USING KNN IMPUTER AND MULTILAYER PERCEPTRON," *Water (Switzerland)*, vol. 14, no. 17, Sep. 2022, doi: https://doi.org/10.3390/w14172592.

[26] T. Aljrees, "IMPROVING PREDICTION OF CERVICAL CANCER USING KNN IMPUTER AND MULTI-MODEL ENSEMBLE LEARNING," *PLoS One*, vol. 19, no. 1 January, Jan. 2024, doi: https://doi.org/10.1371/journal.pone.0295632.