Journal of Soft Computing Exploration
Homepage: htttps://shmpublisher.com/index.php/joscex
p-ISSN: 2746-7686
e-ISSN: 2746-0991

.

Comparative evaluation of random forest, GRU, and transformer models for soil moisture prediction using reanalysis and meteorological time series data

Defilia Fatikasari[1], Miftahul Walid[2], Muhsi[3], Moh.Aminollah Hamzah[4]
[1,2,4]Department of Informatics Engineering, Faculty of Engineering, Universitas Islam Madura, Indonesia
[3]Department of Information System, Faculty of Engineering, Universitas Islam Madura, Indonesia

Article Info

Article history:

Received April 16, 2026
Revised May 7, 2026
Accepted May 14, 2026

Keywords:

Soil moisture prediction
Random forest
GRU
Transformer
Time series

ABSTRACT

Soil moisture is an important indicator influenced by climate change and water resource availability, with significant impacts across various sectors, particularly agriculture. Limited continuous observational data necessitate historical data-driven approaches for accurate soil moisture prediction. This study aims to comparatively evaluate the performance of Random Forest, Gated Recurrent Unit (GRU), and Transformer models in soil moisture prediction using time series data based on a combination of NASA POWER reanalysis and BMKG meteorological data for the period 2020–2025. The methodology involves data acquisition, preparation, preprocessing (train–test split, min-max normalization, and windowing), model training, and performance evaluation based on MAE, RMSE, and R^2. The results show that the Random Forest model with a window size of 21 days achieves the best performance, yielding an MAE of 0.0396, an RMSE of 0.0534, and an R^2 of 0.8585. The Random Forest model produces predictions closest to the actual values and demonstrates better stability in capturing time series patterns, outperforming the GRU and Transformer models in soil moisture prediction using integrated global and local time series data.

Corresponding Author:

Miftahul Walid,
Department of Informatics Engineering,
Universitas Islam Madura,
Jl. PP. Miftahul Ulum Bettet, Pamekasan, East Java 69351, Indonesia
Email: miftahul.walid@uim.ac.id
https://doi.org/10.52465/joscex.v7i2.54

1. INTRODUCTION

Soil moisture is an important indicator across various sectors, particularly agriculture, as it determines water availability for crops and influences land productivity [1]. In Indonesia, the agricultural sector plays a strategic role in supporting national food security, with a large portion of the population still relying on it as their primary source of livelihood [2]. Global climate change and limited water resources pose serious challenges to agriculture, leading to soil moisture instability and directly affecting crop productivity [3]. In addition, rainfall variability and air temperature fluctuations further influence soil moisture dynamics [4], [5].

In this context, climate variability has been widely reported to significantly impact agricultural productivity. Previous studies indicate that each 1°C increase in global temperature can reduce crop yields by approximately 3%–7% for major crops [6]. Despite its significance, continuous and direct observational data remain limited, especially at high temporal resolution. Consequently, data-driven predictive models using historical data are needed to model and forecast soil moisture dynamics more accurately.

Data-driven approaches using machine learning and deep learning have become increasingly important in soil moisture prediction as artificial intelligence technologies continue to advance. Several studies have reported that these methods are capable of delivering strong predictive performance, commonly assessed using regression metrics such as MAE, RMSE, and R^2. Prior research has consistently shown that machine learning models, especially Random Forest, perform well in soil moisture prediction tasks. For example, Alahmad et al. (2025) reported that Random Forest produced stable and reliable predictions compared to LSTM and XGBoost, with RMSE values in the range of 0.498–0.89 and MAE values of 0.416–0.74 across different soil types and depths [7]. Similarly, Li and Yan (2024) found that ensemble-based models, especially Random Forest, tend to perform better than deep learning architectures in remote sensing-based soil moisture estimation [8]. In the Indonesian context, Random Forest also shows competitive performance, achieving an RMSE value of 0.0158 alongside an R^2 score of 0.7018, confirming its effectiveness for local soil moisture prediction [9], [10]. These findings indicate that Random Forest is robust and reliable for handling structured and multivariate environmental data.

In contrast, deep learning models such as Gated Recurrent Unit (GRU) and Transformer are particularly suitable for time series data because they can capture temporal patterns effectively. Wang and Zha (2024) reported an R^2 value of 0.523 for the Transformer model, exceeding the performance of LSTM, which obtained an R^2 of 0.485 in soil moisture prediction tasks [11]. Additionally, hybrid and GRU-based models have shown performance improvements ranging from 5% to 30% over single models in capturing temporal patterns [12], [13]. Furthermore, Custódio and Prati (2024) found that Random Forest remains more stable across datasets, while deep learning models are more sensitive to temporal variability, as reflected in higher error variations [14]. Recent systematic reviews also highlight that Random Forest and GRU are among the most widely used approaches due to their strong predictive capability under diverse environmental conditions [15], [16], [17]. Overall, these results suggest a balance between model stability and temporal modeling capability, motivating a comparative evaluation of Random Forest, GRU, and Transformer for soil moisture prediction.

However, most previous studies focused on global or remote sensing data without directly integrating local meteorological observations. In addition, systematic comparative evaluation of the performance of Random Forest, GRU, and Transformer models using time series data based on a combination of reanalysis and local meteorological data remains limited, particularly in tropical regions such as Indonesia.

To overcome this limitation, this study examines the comparative performance of Random Forest, Gated Recurrent Unit (GRU), and Transformer models in predicting soil moisture using time series data obtained through the integration of NASA POWER global reanalysis data and BMKG local meteorological data, an approach that remains underexplored in tropical regions. Model performance was evaluated through MAE, RMSE, and R^2 to determine the most accurate method [8]. The results provide deeper insights into temporal soil moisture dynamics.

2. METHOD

This research adopts a quantitative approach with a computational experimental design to assess the performance of soil moisture prediction models using time series data. The modeling process is carried out with the Python programming language on the Google Colaboratory platform, which provides a cloud-based environment for data processing, model development, and visualization. In this study, different Python libraries are utilized based on their specific functionalities. The Pandas and NumPy libraries are used for data preprocessing and numerical computations because they handle large-scale time series data efficiently. During the development of models, the scikit-learn library is used to build machine learning models such as Random Forest. Meanwhile, TensorFlow/Keras is employed to develop deep learning models, including GRU and Transformer, as they are effective in capturing sequential patterns and complex temporal dependencies. Data visualization is conducted using Matplotlib and Seaborn, which are integrated within the Google Colaboratory environment. These libraries are specifically chosen because they provide flexible and comprehensive tools for visualizing data distributions, trends, and model performance. Therefore, although all processes are executed within a single platform (Google Colaboratory), multiple libraries are used to support different analytical tasks effectively.

Figure 1 presents the research workflow, which is carried out through several systematic stages as follows:

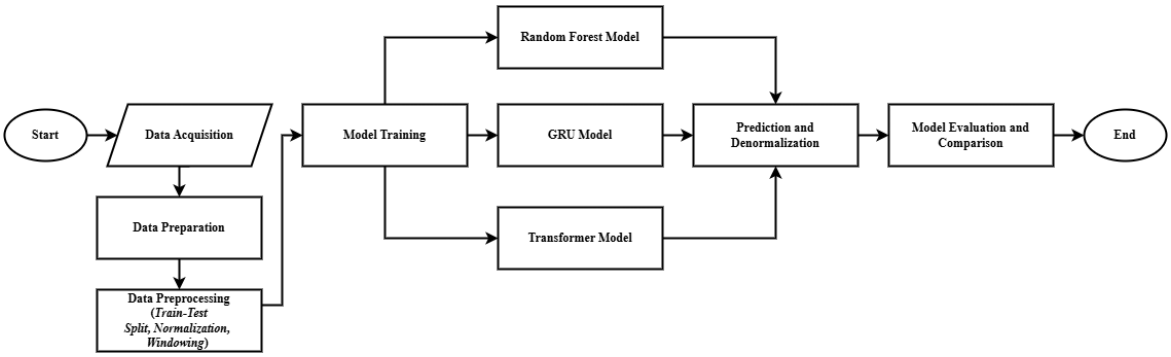

Figure 1. Research flow diagram

Data Acquisition

This research employs a daily historical dataset spanning six years, from January 1, 2020, to December 31, 2025, totaling 2,190 samples. The dataset is secondary and obtained from official online sources. Meteorological variables were collected from the Indonesian Agency for Meteorology, Climatology, and Geophysics (BMKG), specifically from the Trunojoyo Meteorological Station, including average air temperature (TAVG, °C), relative humidity (RH_AVG, %), rainfall (RR, mm), and sunshine duration (SS, hours/day). Soil moisture data were obtained from NASA POWER based on MERRA-2 reanalysis, corresponding to the coordinates of the BMKG station. The variable used is root zone soil wetness (GWETROOT), ranging from 0 to 1, representing the ratio of actual soil moisture to saturated conditions at a depth of 0–100 cm. A sample of the dataset is presented in Table 1 to provide an overview of its structure.

Table 1. Dataset sample utilized in this research

DATE	TAVG (°C)	RH_AVG (%)	RR (mm)	SS (hours)	GWETROOT
2020-01-01	26.3	88.0	23.5	3.4	0.74
2020-01-02	28.5	82.0	12.4	0.0	0.75
2020-01-03	28.2	82.0	0.0	8.3	0.74
2020-01-04	28.3	82.0	0.4	5.8	0.75
2020-01-05	27.7	84.0	0.1	2.1	0.76
...	...	...	...	...	...
2025-12-31	26.3	88.0	0.8	1.4	0.82

The datasets were integrated according to the date column to ensure temporal alignment. TAVG, RH_AVG, RR, and SS were used as the input variables (X), while GWETROOT was assigned as the target variable (Y). This approach is consistent with the methodology of Alahmad et al. (2025), which demonstrates that combining meteorological variables and soil moisture data can improve predictive performance in machine learning models [7]. Access to the dataset supporting this research can be found at the Mendeley Data repository via the following link: https://data.mendeley.com/datasets/4c7tj3nx32/2.

Data Preparation

Prior to modeling, this stage ensures data consistency and quality. Date variables are converted to datetime format, and numerical variables are transformed into float types. Invalid values and missing values are identified and handled accordingly. Since this study is based on time series data, missing values in continuous variables are interpolated linearly to preserve temporal continuity [18], [19]. Missing or invalid values for the rainfall variable (RR) are replaced with 0, indicating no rainfall conditions [20]. To maintain chronological order, the date column is set as the time index (DatetimeIndex) [21]. Furthermore, to better understand the dataset's characteristics, exploratory data analysis is conducted using trend visualization, correlation analysis, and descriptive statistics.

Data Preprocessing

The preprocessing stage is performed sequentially to prevent data leakage between training and testing datasets. To preserve the temporal structure, the dataset was split chronologically into 80% training and 20% testing data [8]. The earlier data points were used for model training, while the later portion was reserved for testing to assess model generalization on unseen data.

After splitting, normalization was applied to both input and target variables to improve model stability and training efficiency [7]. Min-max scaling is employed to transform each variable into the range [0, 1] as expressed in Equation (1):

$$X_{\text{scaled}} = \frac{X - X_{\min}}{X_{\max} - X_{\min}} \tag{1}$$

where X denotes the value before normalization, $X_{\min}$ and $X_{\max}$ represent the minimum and maximum values obtained from the training set, and X_{scaled} denotes the normalized output. To avoid information leakage, the normalization parameters were derived from the training set and subsequently applied to the testing set.

The next step involves constructing sequential data using the sliding window technique to generate input–target pairs based on temporal sequences, allowing the models to accurately capture temporal dependencies [22]. Three window sizes, 7, 14, and 21 days, are used in this study. The windowing process is performed separately on the training and testing datasets after data splitting to avoid data mixing between subsets. For the GRU and Transformer models, this process produces three-dimensional data representations, enabling the models to directly learn temporal relationships [23], [24]. In contrast, although the Random Forest model is not inherently sequence-based, the windowing results are treated as lag features to incorporate historical information [25].

Model Training

This study employs three models: Random Forest, Gated Recurrent Unit (GRU), and Transformer. Each model is trained using the same training dataset to ensure a fair comparison. The hyperparameters are optimized according to the characteristics of each model to improve performance.

Random forest

As a bagging-based ensemble method, Random Forest constructs multiple decision trees and integrates their outputs to achieve more robust and precise forecasts [26]. During the training stage, several key hyperparameters were tuned. For n_estimators, trial values of 100 and 200 were used to improve predictive stability; the max_depth parameter was tested using settings of None, 10, and 20 to regulate model complexity and minimize overfitting; and the splitting criterion (criterion) was set to squared_error, which is widely used for regression problems. Grid search with cross-validation (GridSearchCV) was used to find the ideal hyperparameters.

Using bootstrap sampling, multiple training subsets are generated, and a separate decision tree is constructed for each subset. The final soil moisture prediction is calculated as the mean of the outputs from all trees, as shown in Equations (2)–(5):

$$D = \{(x_i, y_i)\}_{i=1}^{n} \tag{2}$$

$$D_1, D_2, \dots, D_B \tag{3}$$

$$\hat{Y}_b(x) = T_b(x), \quad b = 1,2, \dots, B \tag{4}$$

$$\hat{Y}(x) = \frac{1}{B}\sum_{b=1}^{B} \hat{Y}_b\,(x) \tag{5}$$

where D stands for the training dataset. The variable x_i represents the input feature set of the i-th sample, while y_i denotes its corresponding target output. The symbol n signifies the overall sample count. D_B represents the bootstrap (b-th) sample drawn from D, and B denotes the cumulative number of decision trees. Furthermore, T_b denotes the predictive model derived from the b-th tree training on D_B. $\hat{Y}_b(x)$ represents the prediction produced by the b-th tree for input x, while $\hat{Y}(x)$ denotes the final Random Forest output obtained by averaging the predictions of all trees.

Gated recurrent unit (GRU)

As a recurrent neural architecture, the GRU extends the conventional RNN structure to better model temporal dependencies and alleviate the vanishing gradient problem [13]. Key hyperparameters tuned in this study include the number of neurons (units) with values of 32 and 64, the number of layers with values of 1 and 2; learning rates of 0.001 and 0.0005; and batch sizes with values of 16 and 32. The number of neurons and the structure of network layers determine the model's capability in capturing complex temporal patterns, while parameters such as learning rate and batch size control the convergence behavior and training stability during optimization. Hyperparameter tuning is performed manually.

The GRU model consists of several key components. The reset gate regulates the extent to which information from the previous hidden state contributes to the formation of the candidate hidden state.

Meanwhile, the update gate controls the balance between newly learned information and previously retained knowledge. The candidate hidden state ($\tilde{h}_t$) is computed based on the current input and the reset gate output, while the final hidden state (h_t) is obtained by combining the previous hidden state with the candidate state according to the update gate mechanism. The mathematical formulation is presented in Equations (6)–(9):

$$r_t = \sigma(W_r \cdot [h_{t-1}, x_t] + b_r) \quad (6)$$
$$z_t = \sigma(W_z \cdot [h_{t-1}, x_t] + b_z) \quad (7)$$
$$\tilde{h}_t = \tanh(W \cdot [r_t \odot h_{t-1}, x_t] + b_h) \quad (8)$$
$$h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t \quad (9)$$

where r_t and z_t signify the reset and update gates, while W_r, W_z, and W represent the weight matrices. The bias vectors are indicated by b_r, b_z, and b_h. x_t denotes the input at time step t. Furthermore, h_t signifies the hidden state at time t, and the symbol $\odot$ is used for element-wise multiplication. The soil moisture prediction is subsequently generated by passing the hidden state h_t through a dense layer.

Transformer

Transformer is a deep learning architecture that uses parallel sequence processing to efficiently represent long-range dependencies. It is based on the self-attention mechanism [11], [27]. The model has shown excellent performance in time series-based environmental prediction issues, despite its initial introduction for natural language processing tasks [28].

In this study, several key hyperparameters of the Transformer model were manually tuned, including the number of encoder layers (num_layers: 1 and 2), the number of attention heads (num_heads: 2 and 4), the embedding dimension (d_model: 32 and 64), and the feed-forward network dimension (dff: 64 and 128), as well as a dropout rate of 0.1. These hyperparameter settings were chosen to evaluate their effects on model complexity and predictive performance. Specifically, dropout was used to reduce overfitting, whereas the attention heads allowed the model to capture multiple feature representations simultaneously. Meanwhile, the number of layers and dimensional settings determined the representational capacity of the model.

The Transformer model is employed to learn the relationships among input variables and temporal patterns before passing the resulting representation through a dense layer for soil moisture prediction.

The time series input follows this formulation:

$$X = [x_1, x_2, \dots, x_T] \in \mathbb{R}^{T \times d} \quad (10)$$

Before entering the attention mechanism, the input is processed through embedding and positional encoding:

$$Z = XW_E + PE \quad (11)$$

The self-attention module subsequently transforms the input into Query, Key, and Value (Q, K, and V) vectors, as formulated below:

$$Q = ZW_Q, \qquad K = ZW_K, \qquad V = ZW_V \quad (12)$$

The attention scores are computed using the scaled dot-product formulation:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (13)$$

To capture information from multiple representation subspaces, multi-head attention is applied in parallel as follows:

$$\text{MultiHead}(Q, K, V) = \text{Concat}(head_1, \dots, head_h)W_O \quad (14)$$

The resulting representation is refined using residual connections and layer normalization:

$$Z' = \text{LayerNorm}(Z + \text{MultiHead}(Q, K, V)) \quad (15)$$

A feed-forward network (FFN) is then applied for further transformation:

$$\text{FFN}(x) = \max(0, xW_1 + b_1)W_2 + b_2 \quad (16)$$

Finally, the output is produced through another residual connection:

$$\text{Output} = \text{LayerNorm}(Z' + \text{FFN}(Z')) \quad (17)$$

where Z denotes the embedded representation, T represents the length of the time window, d refers to the number of input variables, W_E is the embedding weight matrix, and PE denotes positional encoding, which is used to preserve temporal order. d_k indicates the key dimension used for scaling to ensure numerical stability. W_Q, W_K, and W_V are projection matrices corresponding to Query, Key, and Value, respectively, and

W_O is the output projection matrix. W_1 and W_2 denote feed-forward network weight matrices, b_1 and b_2 are bias vectors, and h represents the number of attention heads.

Prediction and Denormalization

The trained models were then employed to produce soil moisture predictions using the testing dataset after the completion of the training phase. Subsequently, an inverse min–max scaling procedure was applied to convert the predicted values back to their original scale.

$$X = X_{\text{scaled}} \, (X_{\max} - X_{\min}) + \, X_{\min} \tag{18}$$

where X denotes the denormalized (original) value, X_{scaled} represents the normalized value, and $X_{\min}$ and $X_{\max}$ indicate the minimum and maximum values obtained from the training data.

Model Evaluation and Comparison

To analyze how well the models performed on the test data, three evaluation metrics were employed: Mean Absolute Error (MAE), Root Mean Square Error (RMSE), and Coefficient of Determination (R^2) [8].

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{19}$$

The MAE serves to quantify the average magnitude of prediction errors, where smaller values reflect a higher level of model accuracy.

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{20}$$

Because of the squaring process, RMSE gives more significance to larger errors. Enhanced model accuracy is reflected in a reduced RMSE value.

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{21}$$

The R^2 value represents the proportion of variance in the observed data that is captured by the model. Values closer to 1 indicate stronger explanatory capability.

In these formulas, n signifies the total observations, while y_i and $\hat{y}_i$ represent the actual and forecasted values. Additionally, $\bar{y}$ refers to the mean of the observed values.

Furthermore, a comparative analysis was conducted to determine the best-performing model by evaluating the performance of Random Forest, Gated Recurrent Unit (GRU), and Transformer across different window sizes, namely 7, 14, and 21 days. The optimal model was selected based on the window size that produced the lowest prediction errors, measured using MAE and RMSE, as well as the highest explanatory power indicated by R^2. After identifying the best-performing model and optimal window size, visual comparisons between actual and predicted values, along with scatter plot analysis, were presented for each model under the best window setting to further evaluate their effectiveness in capturing temporal patterns and the relationship between predicted and observed values.

3. RESULTS AND DISCUSSIONS

Data Quality and Characteristics

Several missing values were identified in the dataset, including TAVG (10 entries), RH_AVG (7 entries), RR (244 entries), SS (11 entries), and GWETROOT (2 entries). Most of these missing values were caused by invalid codes such as "8888," "9999," "-999," or the symbol "-". These values were converted into NaN, after which continuous variables (TAVG, RH_AVG, SS, and GWETROOT) were imputed using linear interpolation, while the RR variable was filled with 0. As a result, all variables were successfully processed without missing values [18], [19], [20].

Table 2 provides an overview of the statistical characteristics for each variable. This includes the count of observations, average results, and dispersion through standard deviation, alongside the range (minimum to maximum) and quartile distributions.

Table 2. Descriptive statistics of the dataset

Variable	Count	Mean	Std	Min	25%	50%	75%	Max
TAVG	2190	28.434	1.097	24.3	27.7	28.4	29.1	32.1
RH_AVG	2190	80.236	6.802	59	75	81	85	98
RR	2190	4.932	12.501	0	0	0	2.8	116.4
SS	2190	6.541	3.295	0	4	7.95	9.5	11.7
GWETROOT	2190	0.818	0.148	0.59	0.66	0.85	0.96	1.0

As shown in Table 2, all variables have the same number of observations, 2,190, indicating that the data processing produced a complete dataset. The TAVG and RH_AVG variables remained relatively stable during the observation period, with means of 28.434°C and 80.236%, respectively. In contrast, the RR variable exhibited the highest variability, with a standard deviation of 12.501, a median of 0, and a maximum value of 116.4 mm, reflecting a right-skewed distribution in which most days experienced little or no rainfall, while a limited number of days showed extreme precipitation events that were retained due to their relevance to soil moisture dynamics [29]. The SS variable shows moderate variability in sunshine duration and plays a role in evapotranspiration processes affecting soil water balance. The target variable, GWETROOT, had an average value of 0.818 with a standard deviation of 0.148, suggesting generally stable and relatively high soil moisture conditions. Overall, these statistical characteristics indicate that the dataset exhibits relevant patterns and variability, supporting both model development and comparative evaluation. To analyze the relationships among variables, a correlation heatmap was constructed and is shown in Figure 2.

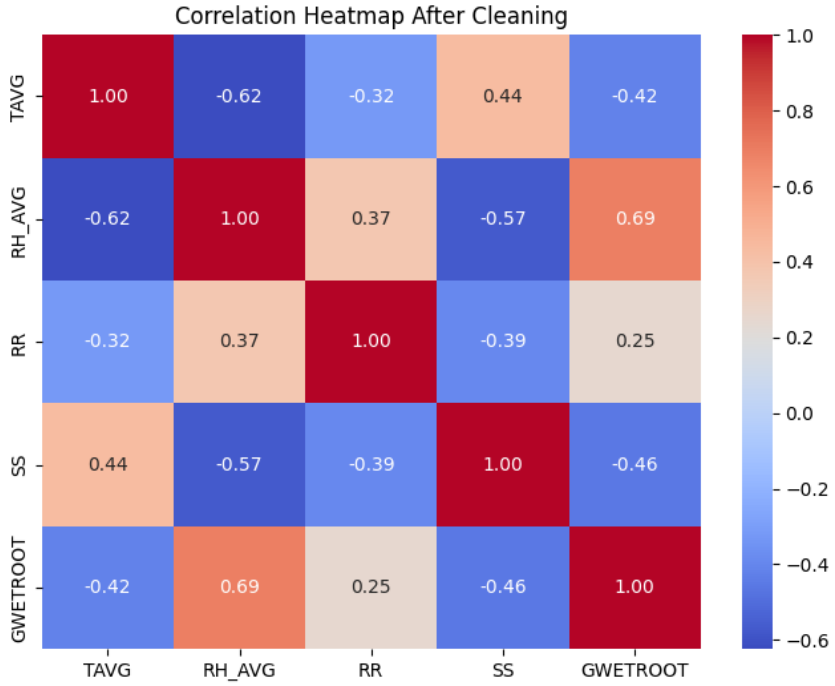

Figure 2. Correlation heatmap between variables

The correlation heatmap of the variables following data cleaning is shown in Figure 2. The strength and direction of the correlation between variables are represented by the color gradient in this heatmap, with warmer colors (red) denoting positive correlations and cooler colors (blue) denoting negative correlations. The Pearson correlation coefficient is represented by the numerical values in each cell, which range from -1 to 1. Strong positive relationships are indicated by values closer to 1, strong negative relationships are shown by values closer to -1, and weak or no correlation is indicated by values around 0. The results show that GWETROOT exhibits a moderately strong positive correlation with RH_AVG (0.69) and a weak positive correlation with RR (0.25), indicating that both air humidity and rainfall contribute to variations in soil moisture levels. In contrast, GWETROOT shows negative correlations with TAVG (-0.42) and SS (-0.46), indicating that higher air temperature and longer sunshine duration tend to reduce soil moisture. Additionally, moderately

strong negative correlations were observed between TAVG and RH_AVG (-0.62) and between SS and RH_AVG (-0.57), while RR demonstrated a moderate positive correlation with RH_AVG (0.37). These findings provide evidence of meaningful associations between meteorological variables and soil moisture before model development. Figure 3 presents the time series plot of GWETROOT, illustrating the temporal variation of root-zone soil moisture throughout the observation period.

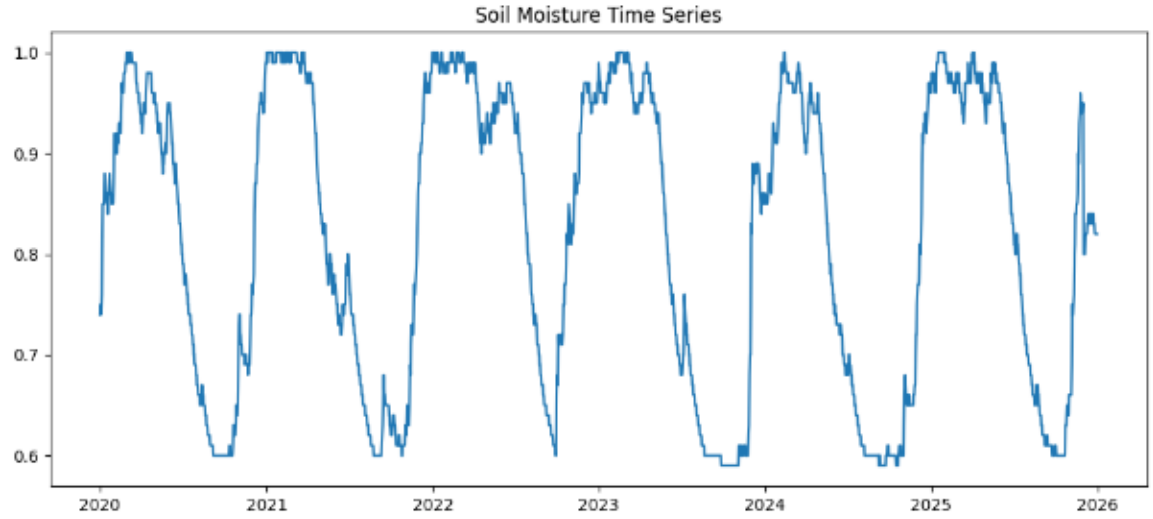

Figure 3. Time series plot of root-zone soil moisture (GWETROOT)

Root-zone soil moisture showed temporal fluctuations ranging from approximately 0.60 to nearly 1.00 throughout the observation period. The pattern demonstrates recurring and relatively consistent annual variations, indicating the influence of seasonal factors. Soil moisture values tended to increase during periods of greater water availability and decrease under drier conditions [30], [31]. This consistent fluctuation pattern suggests that the dataset possesses clear time series characteristics and temporal dependencies, making it suitable for evaluating Random Forest, GRU, and Transformer models for soil moisture prediction.

Dataset Structure for Modeling

After the preprocessing, which included data splitting, normalization, and windowing, the dataset was prepared for model implementation. A total of 1,752 observations were allocated for training and 438 observations for testing. The dataset was split into training and testing subsets, where the training set spans from January 1, 2020, to October 19, 2024, while the testing set covers the period from October 20, 2024, to December 31, 2025. The input features consist of four meteorological variables, structured as X_train (1752 × 4) and X_test (438 × 4), with corresponding targets y_train (1752 × 1) and y_test (438 × 1). Prior to model training, all variables were normalized within the range of 0 to 1.

To model temporal dependencies in the time series data, a windowing approach was applied using sequence lengths of 7, 14, and 21 days. These window sizes were designed to represent short-term, medium-term, and longer-term historical contexts, enabling the evaluation of how different temporal horizons influence prediction performance. As a result, the training data were transformed into shapes of 1745 × 7 × 4, 1738 × 14 × 4, and 1731 × 21 × 4, while the testing data were structured into 431 × 7 × 4, 424 × 14 × 4, and 417 × 21 × 4. For the Random Forest model, the windowed sequences were further reshaped into a two-dimensional representation, producing training data shapes of 1745 × 28, 1738 × 56, and 1731 × 84, with corresponding testing shapes of 431 × 28, 424 × 56, and 417 × 84. This data structure allows each model to utilize historical meteorological information for soil moisture prediction.

Model Performance Evaluation and Comparative Analysis

The soil moisture prediction performance of the Random Forest, Gated Recurrent Unit (GRU), and Transformer models was evaluated and compared in a systematic manner. Each model was implemented using the optimal hyperparameter configuration obtained from the tuning experiments. Three commonly used evaluation metrics for model performance are the coefficient of determination (R^2), Mean Absolute Error (MAE), and Root Mean Square Error (RMSE). The extent of prediction mistakes is quantified by MAE and RMSE, with smaller values indicating greater predictive accuracy. R2, on the other hand, gauges how well the model accounts for the variation in observed soil moisture; values nearer 1 indicate greater explanatory ability. To further analyze the influence of temporal context, model testing was conducted using window sizes of 7, 14, and 21 days. These sequence lengths were selected to examine how different historical time horizons affect prediction performance. The comparative results for all evaluated models are presented in Table 3.

Table 3. Performance evaluation of soil moisture prediction models

Window	Model	MAE	RMSE	R^2
7	Random Forest	0.0597	0.0843	0.6572
7	Transformer	0.0731	0.0894	0.6143
7	GRU	0.0759	0.0954	0.5611
14	Random Forest	0.0462	0.0635	0.8014
14	Transformer	0.0589	0.0732	0.7357
14	GRU	0.0637	0.0793	0.6901
21	Random Forest	0.0396	0.0534	0.8585
21	Transformer	0.0527	0.0680	0.7703
21	GRU	0.0561	0.0703	0.7546

Table 3 indicates that for the 7-day window size, the Random Forest model achieved the highest performance among the evaluated methods, with an MAE of 0.0597, an RMSE of 0.0843, and an R^2 of 0.6572. In comparison, the Transformer and GRU models demonstrated lower performance, indicated by higher error values and lower R^2 scores. For the 14-day window size, all models exhibited improved performance, as indicated by reduced MAE and RMSE values and increased R^2 scores. The Random Forest model consistently remained the best-performing approach, achieving an MAE of 0.0462, an RMSE of 0.0635, and an R^2 of 0.8014, followed by the Transformer and GRU models. Further improvement was observed at the 21-day window size, where all models reached their best performance. Random Forest again outperformed the other models, recording an MAE of 0.0396, an RMSE of 0.0534, and an R^2 value of 0.8585, while Transformer and GRU also improved but remained below Random Forest.

An increase in window size from 7 to 21 days consistently improved the performance of all models. This trend was reflected by a decrease in MAE and RMSE values, along with an increase in R^2 scores, as longer historical sequences were incorporated. The results indicate that extended input sequences provide richer temporal context, enabling the models to better learn root-zone soil moisture dynamics over time.

However, increasing the window size also raises input complexity and therefore must be balanced with model capacity and available data volume. In this study, Random Forest utilized longer historical sequences more effectively than GRU and Transformer. This is attributed to its ability to capture nonlinear relationships among variables through an ensemble decision-tree approach, thereby reducing variance and improving model stability [7], [8]. Furthermore, similar findings have been reported in previous studies, which highlight the effectiveness of Random Forest in soil moisture prediction tasks [9], [10]. In addition, all models were evaluated using the same dataset, input variables, and temporal window configurations to ensure a consistent experimental setting. This comparison was intended to examine how different model architectures, including non-temporal and sequence-based approaches, perform under the same data constraints.

The comparative results demonstrate that the Random Forest model with a 21-day window size consistently outperformed the other configurations. Its lower error values and higher explanatory capability indicate that longer historical sequences significantly enhance prediction accuracy. These results highlight that both model architecture and the selection of temporal window size play a crucial role in time series-based root-zone soil moisture prediction.

Analysis and Visualization of Prediction Results at the Best Window Size

To further validate the quantitative evaluation results, prediction outputs from all models were analyzed using the best-performing window size. As indicated in Table 3, the 21-day window size produced better performance compared the other window size variations. Therefore, the analysis focused on the 21-day window size to evaluate each model's capability in capturing actual data behavior and representing the relationship between actual and predicted values. Visualization was used to facilitate a clearer comparison of model performance. Figure 4 presents a line plot comparing the actual values with predictions generated by Random Forest, GRU, and Transformer models.

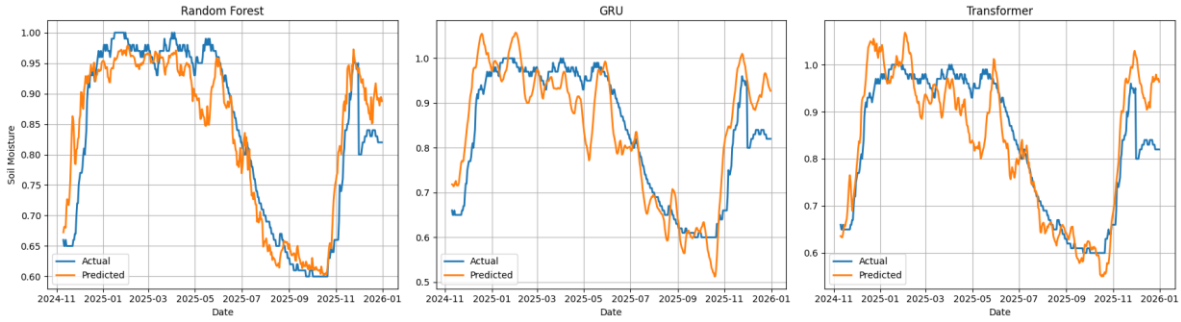

Figure 4. Comparison of actual and predicted value line plots for the three models

The Random Forest model demonstrated the strongest ability to follow the temporal pattern of soil moisture time series data. Its prediction curve was nearly coincident with the actual values, particularly during stable periods and decreases in soil moisture. In contrast, the GRU and Transformer models captured the general temporal pattern but exhibited larger deviations from the actual values, with overestimation and underestimation occurring across multiple periods, especially during high-variability conditions. These findings indicate that although GRU and Transformer were capable of learning temporal dynamics, their prediction stability remained inferior to that of the Random Forest model. This suggests that Random Forest possesses better generalization capability in capturing temporal patterns without producing excessive prediction fluctuations.

The relationship between observed and forecasted values for each model is illustrated in the scatter plots in Figure 5. A dashed diagonal represents the perfect prediction line, where the predictions match the actual data exactly. Points distributed closer to this line reflect higher predictive accuracy.

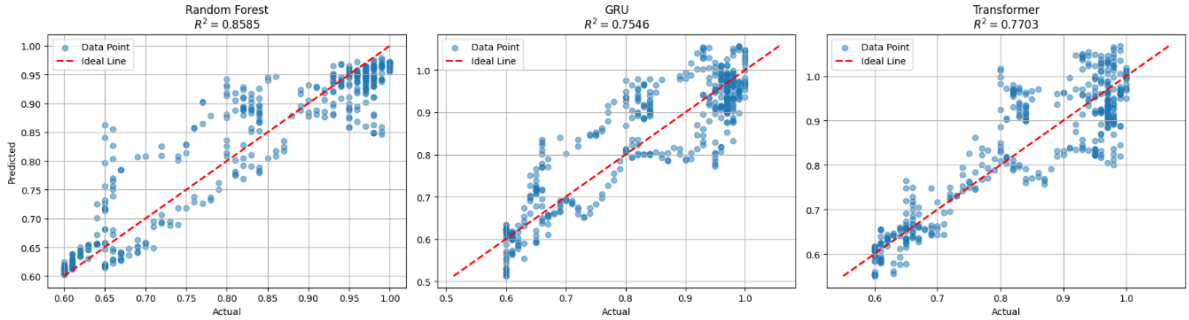

Figure 5. Comparison of actual and predicted value scatter plots the three models

As observed in Figure 5, the Random Forest model exhibits the point distribution that is closest to the diagonal reference line, reflecting strong agreement between predicted and actual values. This result is consistent with its highest R^2 score among all models. In contrast, the GRU and Transformer models present more widely scattered distributions that deviate further from the diagonal line, indicating greater prediction error. Although the Transformer performs slightly better than GRU, its predictive accuracy remains below that of Random Forest. The GRU model shows the widest point dispersion, particularly in the higher soil moisture range. Overall, these findings suggest that the Random Forest model using a 21-day window size provides the most accurate and stable predictions in capturing soil moisture time series patterns, achieving lower prediction errors and more consistent performance than the GRU and Transformer models in representing the relationship between actual and predicted soil moisture data.

4. CONCLUSION

This study comparatively evaluated the performance of Random Forest, GRU, and Transformer models for soil moisture prediction based on time series data derived from a combination of reanalysis and meteorological datasets. The results indicate notable performance differences among the evaluated models in capturing root-zone soil moisture dynamics. The Random Forest model with a 21-day window size delivered the best overall performance, achieving an MAE of 0.0396, an RMSE of 0.0534, and an R^2 value of 0.8585. Increasing the window size was shown to improve model performance, as indicated by reduced prediction errors and enhanced explanatory capability. Random Forest demonstrated superior ability in following temporal patterns and generated predictions closest to the actual values, whereas the GRU and Transformer models exhibited greater deviations, particularly during periods of substantial variation. These findings suggest that model performance is influenced by the suitability between model architecture and data characteristics. In

this study, Random Forest showed more stable and robust performance compared to GRU and Transformer under the given data configuration, including dataset size, input variables, and temporal window settings.

The results of this study demonstrate that integrating global reanalysis data with local meteorological observations can support reliable soil moisture prediction in tropical regions. This finding offers a valuable foundation for developing data-driven irrigation systems and agricultural management applications. Future studies should focus on expanding the dataset, incorporating additional environmental variables, and exploring more advanced or hybrid predictive architectures. Practical implementation through real-time or web-based systems also remains an important direction for future development.

ACKNOWLEDGEMENTS

The authors would like to sincerely thank Universitas Islam Madura and the Faculty of Engineering for their support throughout the completion of this research. The authors also gratefully acknowledge the Indonesian Agency for Meteorology, Climatology, and Geophysics (BMKG) and NASA POWER for providing the datasets used in this study.

CREDIT AUTHORSHIP CONTRIBUTION STATEMENT

Defilia Fatikasari: Data curation, data preprocessing, model implementation and evaluation, data analysis, original draft preparation, visualization, and manuscript. **Miftahul Walid:** Conceptualization, methodology, supervision, validation, manuscript editing. **Muhsi and Moh.Aminollah Hamzah:** Methodology, validation, and manuscript review.

DECLARATION OF COMPETING INTERESTS

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

DATA AVAILABILITY

The dataset supporting the findings of this study is publicly available in the Mendeley Data repository and can be accessed at: https://data.mendeley.com/datasets/4c7tj3nx32/2.

REFERENCES

[1] J. Wang, Y. Zhang, P. Song, and J. Tian, "Estimating sub-daily resolution soil moisture using Fengyun satellite data and machine learning," *J. Hydrol.*, vol. 632, 2024, doi: https://doi.org/10.1016/j.jhydrol.2024.130814.
[2] K. P. R. Indonesia, *Agricultural Statistics 2024*. Pusat Data dan Sistem Informasi Pertanian, 2024.
[3] FAO, *The State of Food and Agriculture 2023 - Revealing the True Cost of Food to Transform Agrifood Systems. In The State of Food and Agriculture 2023*. Food and Agriculture Organization of the United Nations (FAO), 2023.
[4] BMKG, *Catatan Iklim dan Kualitas Udara Indonesia*. Deputi Bidang Klimatologi Badan Meteorologi, Klimatologi, dan Geofisika (BMKG) Jakarta, 2024.
[5] X. Zou *et al.*, "Precipitation Sensitivity to Soil Moisture Changes in Multiple Global Climate Models," *Atmos.*, vol. 14, 2023.
[6] C. Zhao *et al.*, "Temperature increase reduces global yields of major crops in four independent estimates," *PNAS*, vol. 114, no. 35, pp. 9326–9331, 2017, doi: 10.1073/pnas.1701762114.
[7] T. Alahmad, M. Neményi, A. Széles, N. Ali, O. Hijazia, and A. Nyéki, "Spatiotemporal prediction of soil moisture content at various depths in three soil types using machine learning algorithms," *Front. Soil Sci.*, vol. 5, no. September, pp. 1–17, 2025.
[8] M. Li and Y. Yan, "Comparative Analysis of Machine-Learning Models for Soil Moisture Estimation Using High-Resolution Remote-Sensing Data," *L.*, vol. 13, no. 8, pp. 1–24, 2024.
[9] F. Azzahrotunnisa, M. I. Mahfud, N. S. Ishak, and R. Kurniawan, "Perbandingan Metode Supervised Machine Learning untuk Prediksi Kelembapan Tanah di Jakarta," *Semin. Nas. Sains Data 2024 (SENADA 2024)*, pp. 77–87, 2024.
[10] H. Y. Taihuttu, I. S. Sitanggang, and L. Syaufina, "Soil Moisture Prediction Model in Peatland Using Random Forest Regressor," *BAREKENG J. Math. Its Appl.*, vol. 18, no. 4, pp. 2505–2516, 2024.
[11] Y. Wang and Y. Zha, "Comparison of transformer, LSTM and coupled algorithms for soil moisture prediction in shallow-groundwater-level areas with interpretability analysis," *Agric. Water Manag.*, vol. 305, pp. 1–17, 2024.
[12] L. Zhao, T. Luo, X. Jiang, and B. Zhang, "Prediction of soil moisture using BiGRU-LSTM model with STL decomposition in Qinghai – Tibet Plateau," *PeerJ*, vol. 11, pp. 1–20, 2023, doi: 10.7717/peerj.15851.
[13] W. Zheng *et al.*, "GRU – Transformer : A Novel Hybrid Model for Predicting Soil Moisture Content in Root Zones," *Agron.*, vol. 14, no. 3, pp. 1–23, 2024.
[14] G. Custódio and R. C. Prati, "Smart Agricultural Technology Comparing modern and traditional modeling methods for predicting soil moisture in IoT-based irrigation systems," *Smart Agric. Technol.*, vol. 7, pp. 1–12, 2024, doi: 10.1016/j.atech.2024.100397.
[15] S. M. Shawon, N. I. Neha, A. N. Jui, N. Dey, and H. T. Zubair, "Advances in soil moisture measurement techniques and prediction using artificial intelligence: An extensive and systematic review," *Smart Agric. Technol.*, vol. 12, pp. 1–34, 2025.
[16] M. Taheri, M. Bigdeli, H. Imanian, and A. Mohammadian, "An Overview of Machine-Learning Methods for Soil Moisture Estimation," *In Water (MDPI)*, vol. 17, no. 11, pp. 1–35, 2025.
[17] M. Lamichhane, S. Mehan, and K. R. Mankin, "Soil Moisture Prediction Using Remote Sensing and Machine Learning Algorithms :

A Review on Progress , Challenges , and Opportunities," *Remote Sens.*, vol. 17, no. 14, pp. 1–36, 2025.
[18] J. Wang, "Data interpolation methods with the UNet-based model for weather forecast," *Int. J. Data Sci. Anal.*, vol. 20, no. 3, pp. 2525–2538, 2025, doi: 10.1007/s41060-024-00611-z.
[19] A. D. Calin, A. M. Coroiu, and H. B. Muresan, "Analysis of Preprocessing Techniques for Missing Data in the Prediction of Sunflower Yield in Response to the Effects of Climate Change," *Appl. Sci.*, vol. 13, pp. 1–21, 2023.
[20] T. Hırca and G. E. Türkkan, "Assessment of Different Methods for Estimation of Missing Rainfall Data," *Water Resour. Manag.*, vol. 38, no. 15, pp. 5945–5972, 2024, doi: 10.1007/s11269-024-03936-3.
[21] V. Kramar and V. Alchakov, "Time-Series Forecasting of Seasonal Data Using Machine Learning Methods," *In Algorithms*, vol. 16, pp. 1–16, 2023.
[22] R. Ndungi and I. S. Blekanov, "Improving Time Series Forecasting by Applying the Sliding Window Approach," *Artif. Intell. Mach. Learn.*, vol. 12, no. 2, pp. 11–18, 2025, doi: 10.33693/2313-223X-2025-12-2-11-18.
[23] X. Song, L. Deng, H. Wang, Y. Zhang, Y. He, and W. Cao, "Deep learning-based time series forecasting," *Artif. Intell. Rev.*, vol. 58, no. 23, pp. 1–67, 2025.
[24] J. Kim, H. Kim, H. Kim, D. Lee, and S. Yoon, "A comprehensive survey of deep learning for time series forecasting : architectural diversity and open challenges," *Artif. Intell. Rev.*, vol. 58, no. 216, pp. 1–95, 2025.
[25] R. P. Masini, M. C. Medeiros, and E. F. Mendes, "Machine Learning Advances for Time Series Forecasting," *arXiv*, pp. 1–44, 2021.
[26] H. Shokati *et al.*, "RandomForest-Based Soil Moisture Estimation Using Sentinel-2, Landsat-8/9, and UAV-Based Hyperspectral Data," *Remote Sens.*, vol. 16, pp. 1–17, 2024.
[27] H. Li, R. S. Miller, D. T. Nguyen, and C. J. Roberts, "Spatiotemporal Modeling of Soil Moisture in Humid Areas by Integrating Transformer Architecture and Remote Sensing Data," in *Science, Engineering and Technology Proceedings*, 2025, pp. 96–102.
[28] R. P. dos Santos, J. P. Matos-Carvalho, and V. R. Q. Leithardt, "Deep learning in time series forecasting with transformer models and RNNs," *PerrJ Comput. Sci.*, pp. 1–30, 2025, doi: 10.7717/peerj-cs.3001.
[29] P. Rajak, T. Afreen, A. S. Raghubanshi, and H. Singh, "The impact of rainfall variability on selected soil properties and ecophysiological traits in Prosopis juliflora invaded plots," *Sci. Rep.*, vol. 15, pp. 1–16, 2025.
[30] J. D. Galvincio, R. D. Q. Miranda, and G. G. da Luz, "Use of Soil Moisture as an Indicator of Climate Change in the SUPer System," *Hydrol.*, vol. 11, no. 65, pp. 1–16, 2024.
[31] H. Liu *et al.*, "Spatiotemporal variability of soil moisture and its influencing factors in a forested catchment with complex terrain," *CATENA*, vol. 256, 2025, doi: https://doi.org/10.1016/j.catena.2025.109079.