Advance Sustainable Science, Engineering and Technology (ASSET)
Vol. 8, No.3, July 2026, pp. 02603030-01 ~ 02603030-012
ISSN: 2715-4211 DOI: https://doi.org/10.26877/asset.v8i3.3061

Metaheuristic Optimization Stacking Application for Rainfall Classification: A Comparative Study

Rachmat Bintang Yudhianto*, Agus M. Soleh, Anang Kurnia

Program in Statistics and Data Science, School of Data Science, Mathematics, and Informatics, IPB University. Jl. Raya Darmaga Kampus IPB, Babakan, Bogor Regency, 16680, West Java, Indonesia

*ydth_2000_rachmat@university.ac.id

Abstract. Accurate rainfall classification plays a vital role in effective meteorological forecasting, agricultural planning, and also to avoid natural disasters. However, standard classification models often exhibit unstable performance. This study evaluates the effectiveness of an Ensemble Stacking framework enhanced with Optimized Swarm Based and Metaheuristic method such as Artificial Bee Colony (ABC) and Cuckoo Search (CS) optimization algorithms to improve prediction reliability. The proposed approach was tested using a detailed rainfall dataset by combining basic classification models, such as Decision Tree, SVM, Naive Bayes, and kNN. The results show that Stacking Ensemble generally outperform individual basic models in Accuracy and F1 Score (reaching a median > 0.80), while unoptimized Stacking method show low variances but provides a less better result in terms of Accuracy and F1 Score. In contrast, the Stacking model optimized with ABC emerged as a better method, demonstrating the highest stability and significantly reducing the performance distribution range compared to the non-optimization and Cuckoo Search scenarios. These findings conclude that the application of Artificial Bee Colony optimization to Stacking ensembles effectively minimizes prediction variance, making it the most reliable strategy for consistent rainfall forecasting using classification modelling technique.

Keywords: Ensemble Machine Learning, Precision Meteorology, Metaheuristic Optimization, Agroclimatic Precision, Rainfall Prediction

(Received 2025-12-01, Revised 2026-05-30, Accepted 2026-06-02, Available Online by 2026-06-17)

1. Introduction

In terms of sustainable development in agriculture and the environment, rainfall is a relevant topic of discussion given the unpredictable climate, especially in Indonesia, which is highly dependent on meeting the needs of its people by relying on agricultural yields. Analysis and prediction of rainfall, as explained in previous research, are crucial in improving agricultural productivity in terms of sufficient food sources and the need for clean and safe water sources by accurately accumulating and estimating the amount of daily rainfall [1]. Sustainable development centered on rainfall not only on focused on secure food and water establishment but also prevention of natural disasters caused by extreme rainfall that continues to occur, one of which is in West Java province, Indonesia. Land topography, soil moistures and other variability of sources happens to effect flood disasters in West Java due to extreme rainfall activity in recent times, analysis of factors that occur rainfall could be related to those variability of sources and helps us to study and prevent the next disaster to come [2] [3]. These variability of rainfall factors can be used to create prediction and analysis using some of many methods in which one of them used machine learning methods through classification technique [4].

Classification modeling using Machine Learning is one of the solutions for estimating or predicting outcomes based on a dataset [5]. The prediction results often depend on the characteristics of the data and the Machine Learning model used. While traditional machine learning architectures such as Decision Trees and Naive Bayes have been foundational in this domain, they frequently demonstrate high variance and limited generalization when faced with the inherent noise and dimensionality of environmental datasets [6]. To address these limitations, an Ensemble method can be applied, which combines several predictions from multiple models [7]. In this study, the Ensemble method used is the Stacking-based Ensemble [8].

Stacking is an Ensemble method that combines predictions or estimations from individual models (base learners). The estimation from each single model is then called meta-features, which are used by a meta-learner in Stacking to determine the final result. Stacking Ensemble frameworks have emerged as a robust alternative, synthesizing the predictive strengths of diverse base learners to achieve strong accuracy [9]. Modeling using the Stacking Ensemble method has been shown to improve prediction accuracy. Previous studies by Karomah [10] and Sunarko [11] demonstrated that the Stacking Ensemble approach provides better prediction accuracy compared to using single models individually. However, to support the effectiveness of these Stacking method is critically dependant on the configuration of high-dimensional hyperparameter spaces, which traditional grid-search methods fail to explore efficiently. These hyperparameters can be optimized using Metaheuristic method, which is Optimization Algorithms [12].

Studies titled “Hybrid stacking ensemble algorithm and simulated annealing optimization for stability evaluation of underground entry-type excavations” [13] and “Improved prediction of slope stability using a hybrid stacking ensemble method based on finite element analysis and field data” [14] both use Ensemble Stacking showed significant performance improvements when using optimization algorithms such as Artificial Bee Colony and Cuckoo Search to find the best hyperparameters for each base model. Based on this background, this research aims to compare and evaluate theoretical modeling and practical deployment by conducting a rigorous comparative evaluation of the Artificial Bee Colony (ABC) and Cuckoo Search (CS) algorithms for the optimization of a stacked ensemble model. Diverging from the prevailing literature that relies solely on domain-agnostic data, this study centers its primary validation on a rainfall classification case study. To ensure the empirical robustness and algorithmic transferability of the proposed optimization framework, this research further employs three diverse public benchmark datasets as secondary validation targets. This multi-tiered evaluative approach allows for a comprehensive assessment of how these metaheuristics influence model stability and predictive reliability across varying imbalance regimes and data distributions, ensuring the model is sufficiently robust for real-world deployment. The primary scientific novelty lies in the systematic benchmarking of these two distinct optimization within a sustainable engineering context, justifying their computational feasibility and robustness for meteorological decision-support systems.

2. Methods

This study is summarized by the following flow design and utilizes two types of datasets: empirical data and three benchmark datasets. The empirical data consists of rainfall dataset recorded from BMKG Indonesia, spesifically Bogor, West Java area used to support the main target of this research that is sustainable agricultural planning and meteorological forecasting, while the benchmark datasets were collected from the UCI and Kaggle platforms to validate the generalizability of the optimization framework. The research flow design and detailed descriptions of these datasets are provided in Figure 1, Table 1, and Table 2 below.

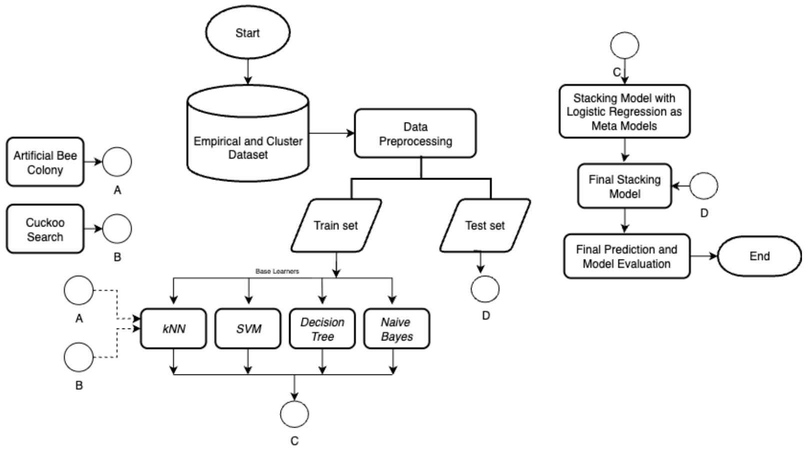

Figure 1. Flow design of research

Table 1. Description of data cluster

No	Dataset	Description	Features	Characteristics
1	*Australian*	This dataset includes observations for predicting credit approval in Australia.	690 observations, 14 predictor variables (categorical and numerical)	A low imbalance proportion with 44% positive observations and 56% negative observations.
2	German Credit Fraud	This dataset includes observations for predicting fraud transcation in classifying credit risk (good or bad).	1000 observations, 20 categorical predictor variables.	A moderate imbalance proportion with 70% good risk and 30% bad risk observations.

No	Dataset	Description	Features	Characteristics
3	*Missing Migrants*	This dataset contains observations about missing migrant population information.	2,420 observations, 10 numerical and categorical predictor variables.	A moderate imbalance proportion with 41% positive observations and 59% negative observations.

Table 2. Description of empirical dataset

Variable	Variable Description	Data Type
Y	Rainfall (RR)	Numeric (mm)
X1	Wind direction at maximum speed (ddd_x)	Categorical
X2	Most frequent wind direction (ddd_car)	Categorical
X3	Maximum wind speed (ff_x)	Numeric (m/s)
X4	Average wind speed (ff_avg)	Numeric (m/s)
X5	Average humidity (RH_avg)	Numeric (m/s)
X6	Duration of sunlight exposure (ss)	Numeric (hours)
X7	Maximum temperature (Tx)	Numeric (°C)
X8	Minimum temperature (Tn)	Numeric (°C)
X9	Average temperature (Tavg)	Numeric (°C)

2.1. Base Models used for Stacking Model

This research mainly used Stacking based ensemble model which include several Models to combine. Stacking model in this research based by the figure 1 above, used datasets with various characteristics and divide those data into train and test set. Train set will be classified in the first phase using base learners likes Decision Tree, Support Vector Machine and many other models to create meta-data predictions from various models with a wide diversity of components, these various components then trained once more using the meta-classifier [15]. In this research, we use 4 machine learning models which had its own characteristics and variety process.

2.2. K-Nearest Neighbor (kNN)

K-Nearest Neighbor (*kNN*) is a machine learning model that works by looking at the similarity between new observations and previous observations then grouping them based by their class [16] [17]. The similarity of the modelling results can be seen from the distance between previous grouped predictors and the new predictions from *kNN*. Model distance calculations uses the following Euclidian formula :

$$\boldsymbol{d(x,y)} = \sqrt{\sum_{i=1}^{n} (\boldsymbol{x_i} - \boldsymbol{y_i})^2} \tag{1}$$

d(x,y) is distance predictors of *x* and *y*
n is total observation

2.3. *Naïve Bayes (NB)*

Naïve Bayes in one of machine learning model that works by all predictors having the same contribution in decision making with all predictors having the same weight and being independent of each other [18] . Naïve Bayes model generally refers to the concept of *Bayesian* probability with the following equation :

$$\boldsymbol{P(A|B) = \frac{P(A)\,P(B|A)}{P(B)}} \tag{2}$$

P(A|B) is the probability of A (class) occurring after event B (feature)
P(A) initial probability of class (A) after seeing feature (B)
P(B|A) is the probability of feature (B) appearing in class (A)
P(B) is the overall probability of feature (B) occurring in all classes (A)

2.4. *Decision Tree*

Decision Tree is a model that represent tree form that creates roots contains node of probabilities. Each node is a decision contains predictor variable that will be tested and will eventually form branches that led us to final node of probabilities and in the end making a final decision [19]. Based by these explanation we can conclude the form of this model following this entropy equation formula :

$$\boldsymbol{Entropy(S) = \sum_{i=1}^{c} P_i log_2 P_i} \tag{3}$$

S represent dataset used
c represent proportion of class counts
P_i represent class i probabilities overall from datasets

2.5. *Support Vector Machine*

Support Vector Machine (*SVM*) is a machine learning model aims to classify data into two categories using linear function in high dimensional features. These models aim to find a separating function (*hyperplane*) in N-dimensional space that indicates the number of predictor variables in the data [20]. This model uses an equation to find separating function in linear hyperplane feature space with the following :

$$\boldsymbol{f(x) = w^T x + b} \tag{4}$$

w represent the weight vector perpendicular to the hyperplane
x represent the input feature vector
$\boldsymbol{b}$ represent the bias term

2.6. *Stacking Ensemble*

Stacking Ensemble is a machine learning model utilize by combining prediction probability from two or more different low-level learners, these model then use a high-level base meta learner algorithm to obtain the best final prediction combined [21] [22]. In this study, there are 4 low-level learners that we would combine that is K-Nearest Neighbor (kNN), Decision Tree, Naïve Bayes, and Support Vector Machine (SVM), and for the meta learner algorithm used is Logistic Regression with this following equation :

$$\boldsymbol{log}\frac{\boldsymbol{p}(\boldsymbol{y}=\mathbf{1})}{\mathbf{1}-(\mathbf{1}-\boldsymbol{p})}= \beta_0+\beta_1X_1+\beta_2X_2+..+\beta_nX_n \tag{5}$$

Where: β_0, β_n, weigh factor, x a prediction from different models and p is reference to total of low-level models used

2.7. Metaheuristic Optmization Algorithm

This research will use two different method of optimization method in purpose to gain the best accuracy in each different low-level models, these method tends to combine the best hyparameter tuning function to gain the maximum and better accuracy. Artificial Bee Colony (ABC) and Cuckoo Search (CS) are the two optimization method that used in this research, Artificial Bee Colony is a prominent Swarm Intelligence based optimizer that mimics the decentralized, self-organized foraging behavior of honey bees, in contrast Cuckoo Search algorithm is a metaphor-based metaheuristic that integrates the biological principles of obligate brood parasitism with the mathematical efficiency of Lévy flights [23] [24]. Both algorithm although were population-based, ABC relies on a specific role-based swarm hierarchy to navigate search spaces, whereas CS utilizes a dual-search strategy combining local random walks with heavy-tailed global trajectories to maintain high population diversity and ensure robust convergence toward the global optimum [25].

2.8. Analysis Procedure

Analysis procedure for this research was divided into three stages: data preparation, modeling, and evaluation. The same procedure was applied to both the empirical data and the cluster datasets.

- The collected datasets were imported into RStudio. After loading, missing values, variable correlations, and standardization for categorical and numerical features were checked to ensure readiness for modelling.
- Rainfall datasets used in this research, additional data preprocessing was carried out, such as data transformation and consideration of extreme and non-extreme rainfall labeling based on information determined by BMKG [26].
- The cleaned data were split into training and testing sets with a 4:1 ratio. In this study we also use SMOTE method applied only on training datasets to perform balance majority and minority class within it.
- Stacking ensemble classification modelling was performed using four different low-level models and Logistic Regression models as the meta learner algorithm which were detailed in section 2.6. In addition, these four low-level models used these tuning hyperparameters detailed in this Table 3 below:

Table 3. Hyperparameter tuning set for each low-level models

Cod.	Algorithm	Algorithm Parameter Tuning Set
kNN	k-Nearest Neighbor	{n_neighbors: (3, 15), weights: ['uniform', 'distance'], metric: ['euclidean', 'manhattan', 'minkowski']}
DT	Decision Tree	{max_depth: (3, 20), min_samples_split: (2, 20), min_samples_leaf: (1, 10), criterion: ['gini', 'entropy']}
NB	Naïve Bayes	{var_smoothing': (1e-10, 1e-8)}
SVM	Support Vector Machine	{C: (0.1, 100.0), gamma: (0.001, 1.0), kernel: ['rbf', 'poly', 'sigmoid']}

- This research contain three scenarios, where two of them used the optimization method. First scenario contain Stacking Ensemble with four low-level models without optimizing its parameter tuning sets, each low-level models use 3-fold validation.
- In second scenario, base models were optimized using the Artificial Bee Colony (ABC) algorithm, which finds the best solution through iterative random search and selection [27]. Finally in third scenario, the Cuckoo Search (CS) algorithm was applied, which generates new solutions using parameter combinations and discards old ones if they are less effective [28] [29]. Each scenarios applied the optimization algorithm to obtain the optimum parameter tuning sets on low-level models detailed in Table 3 and with the same configuration that each low-level model used 3-fold validation.
- Overall, the process of this study was run into three different scenarios, whereas on the second and third scenario applied with total of 30 iteration for each optimized low-level models and for every scenarios from the second until fourth process were repeated for ten times in a row. For both optimization method the designated objective fitness function with 30 iteration was configured to maximize the cross-validated balanced accuracy.

The performance was evaluated and compared among the three scenarios (without optimization, with ABC, and with CS) using three metrics of evaluation. Three metrics used in this research are Accuracy, Balanced Accuracy, and F1-Score.

3. Results and Discussion

This section details the classification modeling outcomes for the primary and cluster datasets. Model performance was evaluated based on accuracy, balanced accuracy, and F1-score. To compare the results, an ANOVA test was applied to Schemes 1 through 3 to assess the variance across these three metrics. This comparative analysis allows us to formulate a final recommendation based on the modeling results.

3.1. Cluster Dataset Modelling Result

This first section mainly discuss about how the six datasets from Kaggle and UCI platform are model using the Stacking Ensemble with 3 schemes, which is using none Optimization Algorithm for the first scheme, Artificial Bee Colony Optimization and Cuckoo Search Optimization for the second and third scheme, these results using benchmark dataset were compared using only one metric, namely accuracy. Table 4 below show the result of all modelling schemes.

Table 4. Modeling results from all data clusters with accuracy metrics value

Setup Scheme	Dataset	Mean Accuracy Results				
		DT	*SVM*	*NB*	*kNN*	*Stacking*
Scheme 1	*Australian*	0.82	0.86	0.81	0.84	**0.86**
Scheme 1	*German Credit Fraud*	0.66	**0.71**	0.64	0.63	0.66
Scheme 1	*Missing Migrants*	0.75	0.76	0.71	0.76	**0.78**
Scheme 2	*Australian*	0.84	0.85	0.81	0.85	**0.86**
Scheme 2	*German Credit Fraud*	0.66	0.70	0.64	0.66	**0.70**
Scheme 2	*Missing Migrants*	0.76	0.76	0.71	0.77	**0.78**
Scheme 3	*Australian*	0.83	0.85	0.81	0.85	**0.86**
Scheme 3	*German Credit Fraud*	0.65	0.69	0.64	0.66	**0.70**
Scheme 3	*Missing Migrants*	0.76	0.76	0.71	0.77	**0.78**

Overall, results shown from Table 4 indicate that the Stacking model consistently produced more stable and competitive classification performance compared to the individual classifiers, namely Desicion Tree, Support Vector Machine, Naïve Bayes, and K-Nearest Neighbor, across all experimental schemes and cluster datasets. The integration of optimization methods into the stacking model generally improved it performance by producing better parameter configuration, maintaining its classication stability. Among all cluster datasets, Australian achieved the highest accuracy values, indicating that the dataset information were easier to identify by the models rather than German Credit Fraud whom showed lower performance due to its imbalanced data distribution. In constrast, Missing Migrants demonstrated relatively stable results across all models, where the optimized Stacking were able to maintain better performance than other individual models. Furthermore based by the results, both Artificial Bee Colony and Cuckoo Search improved the performance of the conventional Stacking model. However, the ABC-Stacking approach achieved slightly better classification performance by producing higher accuracy values in several experimental scenarios. Meanwhile, CSO-Stacking results demonstrated more stable and consistent results across different datasets, indacting good robustness and generalization capability.

3.2. Empirical Dataset Analysis

Empirical dataset namely Rainfall Classification datasets in the Bogor area in 2024 contained 319 valid observation after removing 47 instances of missing or unidentified data from an initial 366 records at the beginning, these 47 unindentified datasets was written as "8888" at the datasets. Overall, the analysis showed that average wind speed were heavily concentrated in the low to moderate range, predominantly 1-2 m/s, which conclude that extreme wind are exceptionally rare in the Bogor area in 2024. Directionally, the observation overwhelmingly dominated by the calm category around 94% from the valid observation, suggest that the local wind dynamics are highly stable and weak, compared to other climate factors and merely as a supporting meteorological variable only. Meanwhile as we can observe from **Figure** *2*, primary weather drivers such as maximum temperature (TX), average temperature (TAVG), sunshine duration (SS), and average relative humidity (RH_AVG) exhibited strong correlation between each other.

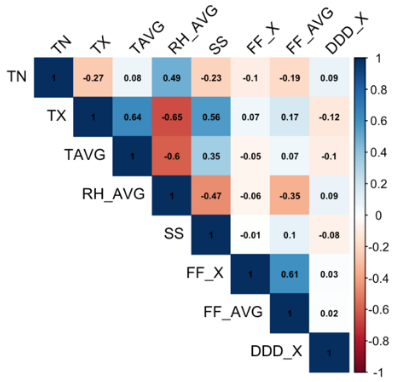

Figure 2. Heatmap correlation between independent variables

In the next section of these analysis, the dataset indicates a significant class imbalance in the target variable when identifying the category of extreme and non-extreme rainfall. Non-extreme rainfall is highly dominant compared to the extreme rainfall category with the proportion of 87:13 total observation as shown on Figure 3, therefore the SMOTE method was applied to balance the class categories in this dataset.

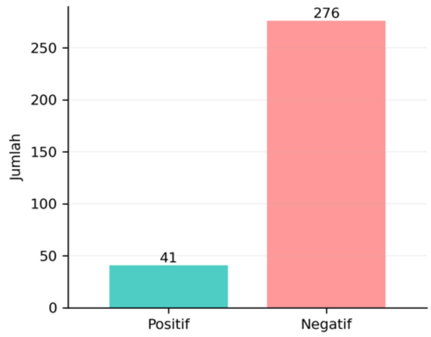

Figure 3. Class distribution of rainfall categories

3.3. Empirical Dataset Modelling Result and Best Model Selection

In this next section, we will use datasets that has been analyze and preprocessed in previous procedure, in this case the data were splited as data train set and test set with 80:20 proportion each. All scenarios were evaluated using Stacking Ensemble Classifier as the main method with two method of Optimization Algorithm, each of them are Artificial Bee Colony and Cuckoo Search.

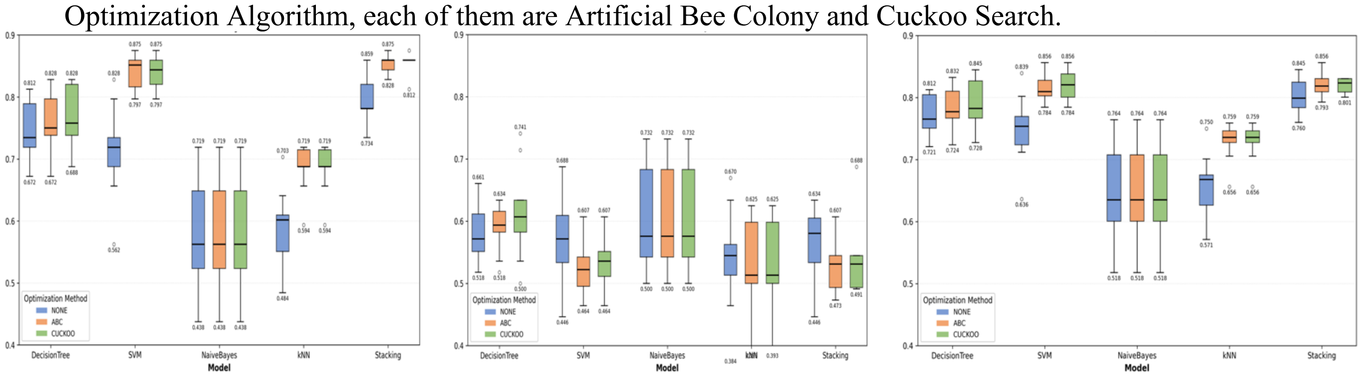

Figure 4. Performance distribution using stacking classifier on each scenarios

As we can see from **Figure *4*** results above, Support Vector Machine exhibits a greater weight contribution than the other base models and it is highly responsive to both algorithm optimzation in achieving higher accuracy. The evaluation also confirms that Artificial Bee Colony and Cuckoo Search optimization combined with Stacking Classifier method capable of producing significantly more robust rainfall category predictions with high performance variance and instability seen in none optimized models are drastically reduced. These scenarios reliably elevate overall correctness, pushing both Accuracy and F1-Score from Stacking scenarios as the leading model performance with approximately 0.875 accuracy. Ultimately, both Artificial Bee Colony and Cuckoo Search with Stacking Classifier function as powerful consistent feature to maximize prediction results for rainfall category classification. However, the deployment of these optimized architectures necessitates careful consideration. Balanced Accuracy metrics in overall analysis results reveals that these optimization processes may inadvertently induce a predictive bias toward the majority class, this degradation in class-imbalanced prediction occurs because the optimization algorithms aggressively exploit the SMOTE training sets. While the SMOTE technique effectively balances the majority nor minority class, the optimization method uses may overtune the hyperparamaters to fit this perfectly synthesized distribution. This interaction highlights that deep hyperparameter optimization can sometimes distort the benefits of synthetic sampling.

4. Conclusion

In conclusion, this research demonstrates that integrating Artificial Bee Colony (ABC) and Cuckoo Search (CS) metaheuristics into Stacking Classifier significantly enhances predictive reliability over none optimized Stacking Classifier based. While ABC occasionally higher predictions accuracy and CS provides more consistent generalization, overall performance remains fundamentally constrained by severe class imbalances. Nevertheless, both algorithms prove highly robust for complex empirical task like Bogor rainfall classification case. To translate these findings into operational deployment, it is recommend to use a tiered approach. User should first established rapid, computationally efficient baselines using unoptimized models. Once those items are set, ABC or CS optimization can be deployed for scenarios where maximizing absolute accuracy is the primary goal. Finally, when the critical objective is detecting rare meteorological events, cost-sensitive learning should be prioritized over further hyperparameter tuning to effectively mitigate majority class bias.

Declaration of AI and AI assisted technologies in the writing process

During the preparation of this work the author(s) used Gemini and ChatGPT in order to find resources and Coding only. After using this tool/service, the author(s) reviewed and edited the content as needed and take(s) full responsibility for the content of the publication

Declaration of Competing Interest

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.

References

[1] Liyew CM, Melese HA. Machine learning techniques to predict daily rainfall amount. J Big Data 2021;8. https://doi.org/10.1186/s40537-021-00545-4.

[2] Ramadhan R, Marzuki M, Suryanto W, Sholihun S, Yusnaini H, Muharsyah R. Rainfall variability in Indonesia new capital associated with the Madden-Julian Oscillation and its contribution to flood events. Quaternary Science Advances 2024;13. https://doi.org/10.1016/j.qsa.2024.100163.

[3] Rahayu R, Mathias SA, Reaney S, Vesuviano G, Suwarman R, Ramdhan AM. Impact of land cover, rainfall and topography on flood risk in West Java. Natural Hazards 2023;116:1735–58. https://doi.org/10.1007/s11069-022-05737-6.

[4] Putra M, Rosid MS, Handoko D. A Review of Rainfall Estimation in Indonesia: Data Sources, Techniques, and Methods. Signals 2024;5:542–61. https://doi.org/10.3390/signals5030030.

[5] Wijoyo A, Saputra AY, Ristanti S, Sya'ban R, Amalia M, Febriansyah R. Pembelajaran Machine Learning. OKTAL : Jurnal Ilmu Komputer dan Science 2024.

[6] Adadi A. A survey on data-efficient algorithms in big data era. J Big Data 2021;8. https://doi.org/10.1186/s40537-021-00419-9.

[7] Nurizki A, Fitrianto A, Mohamad Soleh A. Performance of Ensemble Learning in Diabetic Retinopathy Disease Classification. Scientific Journal of Informatics 2024;11:375–86. https://doi.org/10.15294/sji.v11i2.4725.

[8] Islam MA, Uddin MA, Aryal S, Stea G. An ensemble learning approach for anomaly detection in credit card data with imbalanced and overlapped classes. Journal of Information Security and Applications 2023;78. https://doi.org/10.1016/j.jisa.2023.103618.

[9] Hoxha J, Çodur MY, Mustafaraj E, Kanj H, El Masri A. Prediction of transportation energy demand in Türkiye using stacking ensemble models: Methodology and comparative analysis. Appl Energy 2023;350. https://doi.org/10.1016/j.apenergy.2023.121765.

[10] Karomah BM. PENERAPAN METODE STACKING DALAM MENGKLASIFIKASIKAN PENDERITA PENYAKIT DIABETES. JUPIKOM 2022;1.
[11] Sunarko B, Hasanah U, Hidayat S, Muhammad N, Irfan Ardiansyah M, Khikam Hakiki M, et al. Edu Komputika Journal Penerapan Stacking Ensemble Learning untuk Klasifikasi Efek Kesehatan Akibat Pencemaran Udara. Edu Komputika 2023;10.
[12] Yunianto I, Fadhillah Soleman Y, Sersan Aswan No J, Timur B, Bekasi K, Jawa Barat P, et al. Review: Algoritma Optimasi. DES 2023 Journal of Informatics and Communications Technology 2023;5:111–25. https://doi.org/10.52661.
[13] Liu L, Zhao G, Liang W, Jian Z. Hybrid stacking ensemble algorithm and simulated annealing optimization for stability evaluation of underground entry-type excavations. Underground Space (New) 2024;17:25–44. https://doi.org/10.1016/j.undsp.2023.11.002.
[14] Kardani N, Zhou A, Nazem M, Shen SL. Improved prediction of slope stability using a hybrid stacking ensemble method based on finite element analysis and field data. Journal of Rock Mechanics and Geotechnical Engineering 2021;13:188–201. https://doi.org/10.1016/j.jrmge.2020.05.011.
[15] Kumar M, Singhal S, Shekhar S, Sharma B, Srivastava G. Optimized Stacking Ensemble Learning Model for Breast Cancer Detection and Classification Using Machine Learning. Sustainability (Switzerland) 2022;14. https://doi.org/10.3390/su142113998.
[16] Yin X, Liu Q, Pan Y, Huang X, Wu J, Wang X. Strength of Stacking Technique of Ensemble Learning in Rockburst Prediction with Imbalanced Data: Comparison of Eight Single and Ensemble Models. Natural Resources Research 2021;30:1795–815. https://doi.org/10.1007/s11053-020-09787-0.
[17] Santoso ZR, Hamim Wigena A, Kurnia A. Mendeteksi Unsur Depresi pada Unggahan Media Sosial Menggunakan Metode Machine Learning dengan Optimasi Berbasis Inspirasi Alam. ESTIMASI: Journal of Statistics and Its Application 2025;6:2721–379. https://doi.org/10.20956/ejsa.v6i2.45516.
[18] Peretz O, Koren M, Koren O. Naive Bayes classifier – An ensemble procedure for recall and precision enrichment. Eng Appl Artif Intell 2024;136. https://doi.org/10.1016/j.engappai.2024.108972.
[19] Blockeel H, Devos L, Frénay B, Nanfack G, Nijssen S. Decision trees: from efficient prediction to responsible AI. Front Artif Intell 2023;6. https://doi.org/10.3389/frai.2023.1124553.
[20] Guido R, Ferrisi S, Lofaro D, Conforti D. An Overview on the Advancements of Support Vector Machine Models in Healthcare Applications: A Review. Information (Switzerland) 2024;15. https://doi.org/10.3390/info15040235.
[21] Jayapermana R, Aradea A, Kurniati NI. Implementation of Stacking Ensemble Classifier for Multi-class Classification of COVID-19 Vaccines Topics on Twitter. Scientific Journal of Informatics 2022;9:8–15. https://doi.org/10.15294/sji.v9i1.31648.
[22] Daza A, Ponce Sánchez CF, Apaza-Perez G, Pinto J, Zavaleta Ramos K. Stacking ensemble approach to diagnosing the disease of diabetes. Inform Med Unlocked 2024;44. https://doi.org/10.1016/j.imu.2023.101427.
[23] Alsarhan A, AlJamal M, Harfoushi O, Aljaidi M, Barhoush MM, Mansour N, et al. Optimizing Cyber Threat Detection in IoT: A Study of Artificial Bee Colony (ABC)-Based Hyperparameter Tuning for Machine Learning. Technologies (Basel) 2024;12. https://doi.org/10.3390/technologies12100181.

[24] Joshi AA, Aziz RM. A two-phase cuckoo search based approach for gene selection and deep learning classification of cancer disease using gene expression data with a novel fitness function. Multimed Tools Appl 2024;83:71721–52. https://doi.org/10.1007/s11042-024-18327-4.
[25] Riyanto IMZ, Suprapti L, Fayiza S, Rahmawati EF, Suwandi FY, Oktarina SD, et al. IMPLEMENTATION OF CUCKOO SEARCH-BASED ENSEMBLE VARIABLE IMPORTANCE IN THE CLASSIFICATION OF NON-CASH FOOD ASSISTANCE (BPNT) RECIPIENTS IN WEST JAVA. Barekeng 2026;20:2601–12. https://doi.org/10.30598/barekengvol20iss3pp2601-2612.
[26] Hasnataeni Y, Saefuddin A, Soleh AM. Comparison of Ensemble Forest-Based Methods Performance for Imbalanced Data Classification. Scientific Journal of Informatics 2025;12:183–98. https://doi.org/10.15294/sji.v12i2.24269.
[27] Shunmugapriya P, Kanmani S. Optimization of stacking ensemble configurations through Artificial Bee Colony algorithm. Swarm Evol Comput 2013;12:24–32. https://doi.org/10.1016/j.swevo.2013.04.004.
[28] Mareli M, Twala B. An adaptive Cuckoo search algorithm for optimisation. Applied Computing and Informatics 2018;14:107–15. https://doi.org/10.1016/j.aci.2017.09.001.
[29] Pasaribu AA, Kurnia A. MODEL APPROACH OF AGGREGATE RETURN VOLATILITY: GARCH(1,1)-COPULA VS GARCH(1,1)-BIVARIATE NORMAL. Barekeng 2025;19:2069–82. https://doi.org/10.30598/barekengvol19iss3pp2069-2082.