International Journal of Advances in Data and Information Systems
Vol. 6, No. 3, October 2025, pp. 653~662
ISSN: 2721-3056, DOI: 10.59395/ijadis.v6i3.1428

Enhancing GERD Disease Prediction using Extra Tree Classifier Tuned by Komodo Mlipir Algorithm

Rida Indah Fariani[1], Diya Namira Purba[2]
[1,2]Informatics Department, Astra Polytechnic, Indonesia

Article Info

Article history:
Received Jun 28, 2025
Revised Jul 30, 2025
Accepted Oct 25, 2025

Keywords:
GERD
KMA
Extra Tree Classifier
Disease Detection

ABSTRACT

Gastroesophageal reflux disease (GERD) is a prevalent gastrointestinal disorder characterized by the backward flow of gastric contents into the esophagus, often causing heartburn and regurgitation, with a global prevalence of approximately 13.98%. Early detection is essential to prevent severe complications such as esophagitis, esophageal strictures, and esophageal cancer. However, conventional diagnostic methods are often limited by inadequate healthcare resources and high cost, particularly in developing countries. On the other hand, machine learning can be implemented as a promising alternative method for disease detection, improving accuracy through data pattern identification. Machine learning has been used for several disease detection tasks, such as Breast Cancer, Diabetes, etc. This study proposed an enhanced GERD prediction model by implementing the Extra Tree classifier optimized by the Komodo Mlipir Algorithm (KMA) for hyperparameter optimization. This study used a GERD dataset from the Harvard Dataverse, which consists of 1200 rows with 69 features. The result shows that the Extra Tree Algorithm that KMA tuned achieved a high-performance evaluation with an F1-score of 0.97. This highlights the effectiveness of KMA in enhancing model performance. Compared to the previous study, the proposed Extra Tree Models optimized by KMA performed improved performance, demonstrating the effectiveness of metaheuristic optimization in GERD prediction.

This is an open access article under the CC BY-SA license.

Corresponding Author:
Rida Indah Fariani
Informatics Department,
Astra Polytechnic,
Bekasi, Indonesia.
Email: rida.fariani@polytechnic.astra.ac.id

1. INTRODUCTION

The main Gastroesophageal reflux disease (GERD) denotes a pathological condition characterized by the retrograde flow of gastric contents into eshophagus. This phenomenon can elicit various symptoms, notably heartburn and regurgitation. The prevalence of GERD is significantly high on a global scale with estimated occurrence rate of 13.98% [1]. It is crucial to emphasize the importance of early diagnosis to reduce the incidence of undiagnosed cases and avert more severe complications, such as esophagitis, esophageal strictures, Barrett's eshophagus and esophageal[1].

The current alternative method for disease identification has a limitation. This limitation such as insufficient the total of health personnel and health facilities in developing countries. Additionally, the main challenges in identifying specific ailments, such as Fatty Liver Disease, are primarily due to the high cost. [1].

Recently, machine learning has been implemented as an alternative for disease detection. It shows a good performance in analyzing medical data and also can identify the hidden pattern from

the data. Several studies has been implement machine learning algorithm as the method for disease identifiction. In 2025, Delfina et al. proposed a study about GERD early detection using XGBOOST [1]. This study used a dataset from Harvard Dataverse repository [1]. The study shows that XGBoost has a good performance with the value of accuracy 98% [1].

Another study that shows the performance of machine learning in disease detection proposed by Yen et al in 2022 [2]. This study implement machine learning and deep learning. Dataset for this study used the NBI endoscopic image with the total 671 images after processing [2]. This study shows an evaluation improvement from 78% to 92% from the accuracy [2]. The success of these studies highlight the opportunity to implement machine learning for disease detection specifically for GERD detection [1], [2], [3].

Moreover, studies about an implementation of Extra Tree Classifier in disease detection. In 2023, a study about Heart disease risk assesment with extra tree classifier proposed by Talukder et al [4]. This study implemented Extra tree Classifier with the dataset retrieve from Kaggle platform using 2021 annual BRFSS [4]. This shows the good performance in Extra Tree Classification with the value of accuracy 95%[4].

This study shows the good performance with the evaluation of accuracy 96% of Extra tree classifier in disease detection [5]. This dataset used a Breast Cancer Wisconsin dataset from the UCI machine learning [5]. Its proposed by Sharma et al [5]. Detection of coronary artery disease using extra tree catgorization proposed by Wahid et al in 2023 [6]. The dataset used in this study is Z-Alizadeh Sani CAD dataset. The study shows the performane with the evaluation score 95% [6]. The Extra Tree algorithm offers several advantages in classification and regression tasks, including reducing overfitting and handling large and high-dimensional datasets [7]. However, the algorithm's performance still depends on the selection of optimal hyperparameters [8]. Therefore, it is important to implement an appropriate parameter tuning method. Parameter tuning is the process of identifying the optimal combination of hyperparameter values of a given algorithm to enhance its overall performance [9]. In the case of the Extra Tree algorithm, key hyperparameters such as the number of estimators, maximum tree depth, and the minimum number of samples required to split a node must be selected appropriately [5]. Proper tuning of these parameters can significantly improve the model's accuracy, generalization capability, and robustness [8]. Without a proper parameter tuning method, it may lead to issues such as underfitting or overfitting [9].

To handle the issues, the integration of metaheuristic optimization methods has emerged as a promising solution. One such approach is the Komodo Mlipir Algorithm (KMA), a nature-inspired metaheuristic algorithm designed to balance exploration and exploitation within high-dimensional search spaces [10]. By incorporating KMA into the hyperparameter optimization process, it becomes possible to intelligently and efficiently navigate the parameter space, thereby identifying configurations that yield superior model performance [10], [11]. The combination of the Extra Tree and KMA represents a robust framework for developing highly accurate and generalizable predictive models [12], [10] .

This study explores the enhancement of GERD prediction by integrating the Extra Trees Classifier with the Komodo Mlipir Algorithm for hyperparameter optimisation. The primary objective is to improve the accuracy and reliability of early GERD detection, enabling more responsive and personalised healthcare interventions. By leveraging the strengths of ensemble learning and evolutionary optimization, this approach is expected to address current limitations in prediction performance [11]. Accurate early detection facilitates timely medical treatment and supports preventive strategies such as dietary regulation and lifestyle changes, ultimately reducing the risk of long term complications. This study also contributes to the broader application of machine learning in medical diagnostics, particularly in symptom-based disease prediction, and lays the groundwork for more refined predictive systems in the future.

2. RESEARCH METHOD

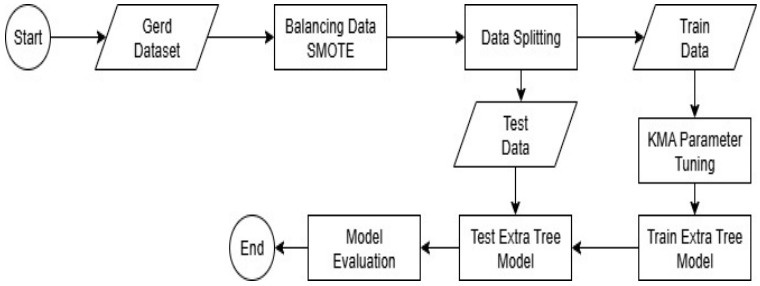


Figure 1. Process Design

Figure 1 illustrates the workflow for predicting GERD using a machine learning model. The process begins with the acquisition of a GERD dataset, followed by the data pre-processing to clean and prepare the data. To handle the imbalance class, the SMOTE (Synthetic Minority Oversampling Technique) method is applied. The balanced dataset is then split into training and testing subsets. The train data is used to build the Extra Trees Classifier model, while the test data is used to evaluate its performance [1]. The trained model is tested on the test data, and the results are assessed through model evaluation metrics. This systematic approach ensures that the model is trained effectively, and its predictive performance is validated rigorously.

2.1 Data Collection

In this study, the data was utilized from the Harvard Dataverse repository. This dataset has 1200 rows of data with 69 features [1]. The target column name is" GORDgrade", which contains label 1 (non-probable GERD), indicating that the patient doesn't have GERD, and label 2 (probable GERD), indicating that the patient has GERD. This dataset has 68 features, including 67 integer features and one float feature. There are no missing values or duplicates in the dataset. Based on Figure 2, there are 755 non-probable GERD and 445 probable GERD. This significant difference in the total of the positive and negative labels that affect the data becomes imbalanced data.

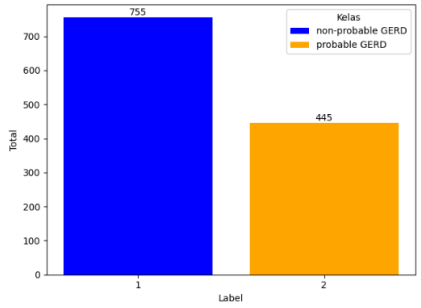


Figure 2. Data Label Distribution

2.2 Splitting Data

In machine learning algorithm implementation, it is necessary to split the data into train and test data. The train data is used for the algorithm to learn about the data pattern and test data is used for the data to evaluate the prediction performance [13]. Therefore, the model can learn and identify the pattern in the data. For this study, the dataset is split into 40 percent of test and 60 of train data.

2.3 Balancing Data

```
Imbalance Ratio (Majority / Minority): 1.7
```

Figure 3. The Imbalance Ratio of GERD Dataset

The main challenge in the GERD dataset is its imbalance of positive and negative labels based on the target column" GORDgrade".Based on the figure, the output shows an imbalance ratio (IR) of 1.7, which shows a high proportion comparison between the majority and minority classes in a classification [14]. The IR is calculated by dividing the number of labels in the majority and minority classes. According to Fernanesz et al in 2018, a value greater than 1 indicates that the data labels are imbalanced; this imbalance condition can affect the classification algorithms' learning process, potentially leading to biased predictions [14]. The imbalance ratio formula is presented in Equation 1.

$$IR = \frac{N_{maj}}{N_{min}} \qquad (1)$$

Where N_{maj} is the size of majority class and N_{min} is the size of minority class [14]. Several balancing methods, such as under-sampling, over-sampling, and hybrid sampling imbalanced data, can handle the imbalance issue [15]. Under-sampling is a balancing method used to handle by reducing the number of majority labels in the dataset [16]. While this approach simplifies the dataset and reduces training time, it risks discarding important information from the data[16]. Oversampling is a balancing method that increases the number of minority classes by duplicating or generating synthetic labels [17]. This method allows for improved class balance without losing potential information from the data [17]. Hybrid sampling combines oversampling and under-sampling, enhancing the minority class while cleaning the noise or redundant majority labels [15]. From these methods, oversampling has an advantage in maintaining important information from the dataset while improving the machine learning algorithm's sensitivity to the minority class [18]. Therefore, this study implements the balancing method, namely the Synthetic Majority Oversampling Technique (SMOTE) [17].

SMOTE is one of the sampling methods that synthesizes the minority class to balance it with the majority class. This method initializes the random sample from the minority class and determines it based on k-nearest neighbors [17]. The use of SMOTE in this study is particularly appropriate because the minority class represents the GERD-positive cases, which are critical for the predictive performance of the model. By increasing the number of GERD cases, SMOTE allows the classifier to learn the patterns and characteristics of the minority class more effectively [17]. This eventually improves the model’s ability to detect GERD-positive instances, enhances the overall classification balance, and leads to more reliability.

2.4 Komodo Mlipir Algortihm Parameter Tuning

The Komodo Mlipir Algorithm (KMA) represents a nature-inspired metaheuristic optimization technique that draws upon the distinctive behavior of Komodo dragons alongside the Javanese notion of mlipir, which translates to stealthily navigating along the roadside to evade threats [18]. KMA categorizes the population into three distinct groups: large males (high-quality solutions), females (medium-quality solutions), and petite males (low-quality solutions), where each group employs varied movement strategies to achieve an equilibrium between exploration and exploitation [10], [19]. Large males adopt a high exploration (HLE) approach, females engage in either mating or pathogenesis to facilitate medium exploration, and petite males implement an innovative mlipir movement to navigate the search space with increased diversity through low exploration [10].

Building upon this theoretical framework, the Komodo Mlipir Algorithm can be effectively adapted as a parameter tuning strategy for machine learning models, where each individual in the population represents a unique combination of hyperparameters. In this framework, the algorithm initiates by establishing a heterogeneous population of candidate parameter configurations, subsequently refining them through iterative performance assessment [19]. This algorithm implements the fundamental evolutionary principle that persists, which candidates exhibiting performance are supplanted by those demonstrating superior efficacy [20], which embodies the dual processes of exploiting advantageous solutions and exploring novel alternatives. Through this

iterative mechanism, KMA adeptly navigates the hyperparameter space to identify optimal configurations. Although the specific instance employed for tuning the machine learning algorithm [10]

Although the specific instance employed for the parameter tuning method may vary, the process is still based on the KMA workflow, as illustrated in figure 4. The process begins with initializing a population, where each individual represents a set of hyperparameters [20]. The algorithm proceeds with the initial fitness of each individual. Based on the fitness evaluations, the algorithm implements a movement strategy; the individual moves to the promising region in the search space [20]. The process then generates a new generation of solutions, often incorporating stochastic variation and recombination [20]. Finally, the fitness of the updated population is re-evaluated to determine progression toward optimality [10]. In this study, the fitness evaluation of the algorithm was conducted using k-fold cross-validation with the F1-score as the primary evaluation metric.

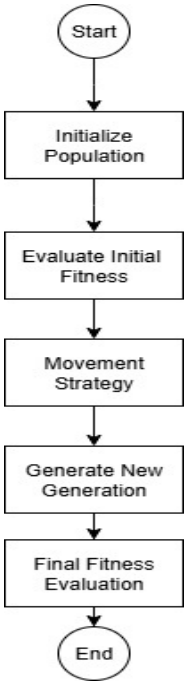


Figure 4 Komodo Mlipir Algorithm Process adapted from [21].

In this study, the algorithm is implemented in a simplified version of KMA to optimize hyperparameters for the Extra Tree Classifier. The original is inspired by the movement of the Komodo, which divides the population into subgroups such as large males, females, and petite males, each employing distinct movement strategies to balance exploration and exploitation. The process initiates with the generation of a heterogeneous population, wherein each individual represents a distinct configuration of hyperparameters, including the number of estimators, maximum tree depth, minimum samples split, and minimum leaf node. The KMA is implemented in an iterative search process by the two parameters, population size and the number of iterations. In this implementation, the population size is 6, indicating that six different hyperparameter configurations are used, and the iterations are 10, where each generation introduces a new solution, generated randomly from the same search space.

The convergence criterion for this implementation is implicit in the number of iterations, which is 10, rather than an adaptive or performance-based stopping rule. Moreover, there is no explicit check for stagnation. The algorithm concludes by selecting the best-performing individual from the final population based on the high accuracy score. Although the implementation simplifies KMA by omitting its explicit subgroup strategies, it successfully preserves the core principles of evolutionary optimization, diversifying the search space while progressively intensifying the focus on promising regions. This adaptation illustrates the algorithm's efficacy and adaptability in navigating the hyperparameter space, ultimately enhancing the predictive performance and computational efficiency of the Extra Tree model.

2.5 Extra Tree Classifier

The Extra Trees classifier, frequently referred to as Extremely Randomized Trees, represents a tree-based ensemble methodology employed for supervised classification and regression endeavors. It is characterized by its high degree of randomization, which involves randomizing both the attribute and cut-point choice when splitting a tree node [5], [7]. This approach results in the construction of totally randomized trees whose structures are independent of the learning sample, thereby enhancing computational efficiency and reducing the risk of overfitting [22].

This algorithm extends traditional decision trees by introducing a high degree of randomization in feature choice and split-point selection. Extra Trees randomly selects candidate split points within the feature range and then chooses the one yielding the most considerable impurity reduction, resulting in an entirely randomized set [7] [23]. An advancement of Extra Trees Classifier is its ability to perform fast, parallelized training with high accuracy, and its effectiveness is further enhanced when combined with hyperparameter tuning, such as metaheuristic algorithms like Komodo Mlipir Algorithm (KMA), which optimizes the tree structure and improves overall predictive performance [24].

2.6 Model Evaluation

$$Q = \frac{TP + TN}{TP + FP + TN + FN} \quad (2)$$

$$PR = \frac{TP}{TP+FP} \quad (3)$$

$$RC = \frac{TP}{TP + FN} \quad (4)$$

$$F1 = 2 \times \frac{PR \times RC}{PR + RC} \quad (5)$$

The Extra Tree models are evaluated using the machine learning evaluation metrics, i.e., Accuracy (Q), Precision (PR), Recall (RC), F1-score and Area Under the Curve [25], [26]. Accuracy measures the proportion of correctly predicted instances, including true negatives and true positives, divided by the total number of instances [25]. Precision is an evaluation metric that measures the ratio of accurate positive predictions to the total number of correct optimistic predictions made by the model [25], [26]. Recall is a formula to calculate the ratio of true positives to the actual positives in the dataset [17], [26]. Lastly, the F1-score is the formula to evaluate the balanced performance between precision and recall. It considers the value of false positives and false negatives [2]. The Area Under the Curve (AUC) is a performance metric that summarizes the effectiveness of a classification model in predicting the positive and negative classes [27]**.**

3. RESULTS AND DISCUSSION

3.1 Balancing Data

Based on Table 1, the dataset significantly differs in the number of positive and negative labels. The total number of non-probable GERD cases is 755, and the probable number is 445 [1]. This significant difference can lead to an imbalance in data [17]. These issues can reduce the performance of the data. To handle this issue, we implement SMOTE balancing method [18]. The method will synthesize the minority class until its balance with the majority class [28]. The balanced result from the balancing method is presented in Table 1.

Table 1. Balancing Data Result

Total Labels	Before SMOTE	After SMOTE
Non-probable GERD	755	755
Probable GERD	445	755

3.2 Splitting Data

In the implementation of machine learning algorithm, the data is split into 50% of train and 50% of test data. The train data is used for algorithm input to train the model. Therefore, the train data can learn and identify the pattern within the data. For test data is used for the algorithm to predict the data and evaluate their performance. From Figure 5, there are 376 non-probable GERD with 234 probable GERD in the train data, and for the test data, there are 379 non-probable GERD and 221 probable GERD.

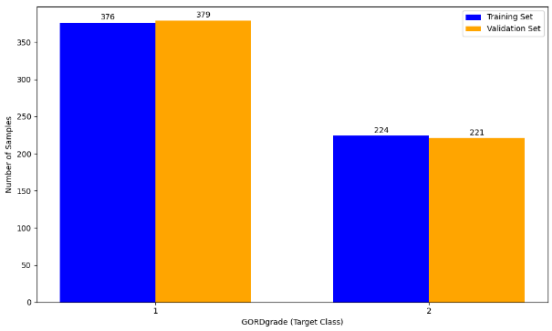


Figure 5. Labels Distribution in Train and Test Data

3.3 KMA Parameter Tuning

Table 2 Tuning Parameter Ranges

Parameter Name	Ranges
n_estimators	10-200
Max_depth	2-30
Min_samples_split	2-10
Min_samples_leaf	1-5

The parameter tuning is implemented to determine the best parameter value for the Extra Tree Algorithm. This model uses the KMA to find the optimal parameter value. The process implements the evolutionary principle to find the best solution in the search space. The parameter ranges for the tuning process are n_estimators 100 and 200, max_depth 2 and 30, min_samples_split 2 and 10, and min_samples_leaf 1 and 5. The best parameter results are n_estimators 78, max_depth 17, min_samples_split 2, and

3.2 Extra Tree Classification Model

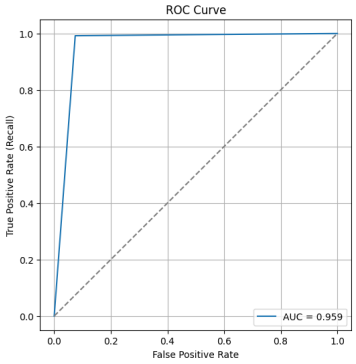


Figure 6 The AUC Evaluation Performance

Table 3. Evaluation Result

Model	Q	F1	PR	RC
Baseline Model	0.95	0.96	1.00	1.00
Prediction Model	**0.97**	**0.96**	**1.00**	**1.00**

Table 4. Evaluation Comparison with Previous Study

Author	F1
Wisesty et al [1].	0.95
Proposed Model	**0.96**

This study considers the F1-score as an overall measurement for all the models. The model evaluation result is presented in Table 3. We compared the performance between the tuned and non-tuned models. Based on Table 3, In the non-tuned model, the performance achieves a value of accuracy of 0.95 with a value of F1-score of 0.96. The tuned model achieves a higher evaluation result with a accuracy value of 0.97, and the F1-score also achieves a similar score of 0.97**.** Figure 6 shows a good performance in AUC equal to 0.95, indicating high performance. It highlights the predicted performance of the Extra Tree Algorithm and KMA. The observed improvement, although moderate, is statistically and practically significant, as it demonstrates the value of integrating metaheuristic optimization methods in the model development process [29]. Overall, the results confirm that KMA-based tuning contributes to improving model robustness, generalization capability, and predictive accuracy, thus reinforcing it applicability in enhancing classification tasks within the domain of disease prediction.

One of the advancements of the Komodo Mlipir Algorithm is its hybrid evolutionary approach, which combines local search and global exploration strategies [21]. Unlike conventional optimization algorithms that often converge prematurely in local optima, KMA is designed to maintain a dynamic balance in exploration. This allows the algorithm to effectively search for optimal solutions and adapt its search behavior based on feedback from fitness evaluation [21].

In summary, the implementation of the Komodo Mlipir algorithm as a tuning method significantly contributes to the overall improvement in the model accuracy, and F1-score the enhanced performance of the tuned model confirms the importance of implementing optimization techniques in machine learning pipelines [21]. We also conducted a comparative analysis between the proposed prediction model and a previously published model for GERD disease classification using machine learning, as shown in Table 4. As presented in the table, the previous study proposed by Wisesty et al. achieves an F1 Score of 0.95 [1]. In contrast, our proposed model, which integrates the Komodo Mlipir Algorithm (KMA) for hyperparameter tuning, achieves a higher F1 Score of 0.96. It shows better performance and highlights the effectiveness of the Extra Tree Algorithm with KMA tuning.

4. CONCLUSION

This study shows that implementing the Komodo Mlipir Algorithm (KMA) for hyperparameter tuning substantially improves the performance of GERD disease prediction using an Extra Tree Classifier. The tuned model demonstrated higher accuracy, with F1-score values of 0.97, compared to the non-tuned model's accuracy of 0.95 and F-score of 0.96. These findings highlight the value of incorporating metaheuristic optimization models to achieve more accurate and reliable predictions [12].

In addition, KMA's hybrid nature effectively combines global exploration and local strategies [30]. The comparative results with other machine learning models further support the proposed method's superiority, as it achieved the highest F1-score among the models evaluated. Overall, the implementation of KMA as a tuning method confirms its potential for broader use in disease predictions.

Based on the Komodo Mlipir Algorithm (KMA) performance in optimizing the Extra Tree Classifier for GERD disease prediction, future research is encouraged to explore the application of KMA in other disease domains and medical datasets to evaluate its generalizability and robustness. Researchers may also investigate the integration of KMA with deep learning models, such as CNNs or LSTMs, to assess its effectiveness in handling more complex, high-dimensional data types, including medical imaging or longitudinal health records. Additionally, exploring adaptive or dynamic variants of KMA could improve tuning efficiency and allow a better balance between global search and local refinement.

REFERENCES

[1] U. N. Wisesty, H. A. Delfina, and I. Kurniawan, "Gastroesophageal Reflux Disease Early Detection using XGBoost Method Classifier," *J. Tek. Inform. Jutif*, vol. 6, no. 2, pp. 855–870, Apr. 2025, doi: 10.52436/1.jutif.2025.6.2.4143.

[2] H.-H. Yen, H.-Y. Tsai, C.-C. Wang, M.-C. Tsai, and M.-H. Tseng, "An Improved Endoscopic Automatic Classification Model for Gastroesophageal Reflux Disease Using Deep Learning Integrated Machine Learning," *Diagnostics*, vol. 12, no. 11, p. 2827, Nov. 2022, doi: 10.3390/diagnostics12112827.

[3] S.-W. Liu *et al.*, "Recent advances in machine learning for precision diagnosis and treatment of esophageal disorders," *World J. Gastroenterol.*, vol. 31, no. 23, Jun. 2025, doi: 10.3748/wjg.v31.i23.105076.

[4] S. H. Talukder, S. K. Mondal, M. Aljaidi, R. B. Sulaiman, and T. Islam, "Heart Disease Risk Assessment and Prediction: A Robust Ensemble Approach with Extra Tree Classifier," in *2023 2nd International Engineering Conference on Electrical, Energy, and Artificial Intelligence (EICEEAI)*, Zarqa, Jordan: IEEE, Dec. 2023, pp. 1–6. doi: 10.1109/EICEEAI60672.2023.10590147.

[5] G. Alfian *et al.*, "Predicting Breast Cancer from Risk Factors Using SVM and Extra-Trees-Based Feature Selection Method," *Computers*, vol. 11, no. 9, p. 136, Sep. 2022, doi: 10.3390/computers11090136.

[6] N. Wahid, A. Zaidi, and G. Dhiman, "Detection of Coronary Artery Disease Using Extra Tree Categorization".

[7] C. Krishna and V. Paul, "Entropy-Based Feature Selection using Extra Tree Classifier for IoT Security," *Iraqi J. Sci.*, pp. 2466–2480, May 2023, doi: 10.24996/ijs.2023.64.5.31.

[8] N. Hussein and S. R. M. Zeebaree, "Performance Evaluation of Extra Trees Classifier by using CPU Parallel and Non-Parallel Processing," *Indones. J. Comput. Sci.*, vol. 13, no. 2, Apr. 2024, doi: 10.33022/ijcs.v13i2.3802.

[9] E. Bartz, T. Bartz-Beielstein, M. Zaefferer, and O. Mersmann, Eds., *Hyperparameter Tuning for Machine and Deep Learning with R: A Practical Guide*. Singapore: Springer Nature Singapore, 2023. doi: 10.1007/978-981-19-5170-1.

[10] S. Suyanto, A. A. Ariyanto, and A. F. Ariyanto, "Komodo Mlipir Algorithm," *Appl. Soft Comput.*, vol. 114, p. 108043, Jan. 2022, doi: 10.1016/j.asoc.2021.108043.

[11] D. Arumuganainar, "Extra Tree Classifier Predicts an Interactome Hub Gene as HSPB1 in Oral Cancer: A Bioinformatics Analysis," *Cureus*, May 2024, doi: 10.7759/cureus.59863.

[12] Y. Li and M. He, "Elevating Student Performance Prediction using Extra-Trees Classifier and Meta-Heuristic Optimization Algorithms," *Int. J. Adv. Comput. Sci. Appl.*, vol. 15, no. 2, 2024, doi: 10.14569/IJACSA.2024.0150247.

[13] B. Lindahl and N. L. Mills, "A new clinical classification of acute myocardial infarction," *Nat. Med.*, vol. 29, no. 9, pp. 2200–2205, Sep. 2023, doi: 10.1038/s41591-023-02513-2.

[14] R. Zhu, Y. Guo, and J.-H. Xue, "Adjusting the imbalance ratio by the dimensionality of imbalanced data," *Pattern Recognit. Lett.*, vol. 133, pp. 217–223, May 2020, doi: 10.1016/j.patrec.2020.03.004.

[15] M. Khushi *et al.*, "A Comparative Performance Analysis of Data Resampling Methods on Imbalance Medical Data," *IEEE Access*, vol. 9, pp. 109960–109975, 2021, doi: 10.1109/access.2021.3102399.

[16] T. Wongvorachan, S. He, and O. Bulut, "A Comparison of Undersampling, Oversampling, and SMOTE Methods for Dealing with Imbalanced Classification in Educational Data Mining," *Information*, vol. 14, no. 1, p. 54, Jan. 2023, doi: 10.3390/info14010054.

[17] E. H. Hermaliani and M. Ernawati, "Penerapan Metode SMOTE Untuk Mengatasi Imbalanced Data Pada Klasifikasi Ujaran Kebencian," *Comput. Sci.*, vol. 4, no. 1, 2024.

[18] D. Dablain, B. Krawczyk, and N. V. Chawla, "DeepSMOTE: Fusing Deep Learning and SMOTE for Imbalanced Data," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 34, no. 9, pp. 6390–6404, Sep. 2023, doi: 10.1109/TNNLS.2021.3136503.

[19] W. A. Syafei, A. Wibowo, and I. A. Ashari, "A Novel SW-KMA-Bi-LSTM Approach for Improving Traffic Flow Prediction," *Int. J. Intell. Eng. Syst.*, 2025.

[20] "Stochastic Komodo Algorithm," *Int. J. Intell. Eng. Syst.*, vol. 15, no. 4, Aug. 2022, doi: 10.22266/ijies2022.0831.15.
[21] Q. Liu and X. Zhang, "Improved Adaptive Komodo Mlipir Algorithm," *IEEE Access*, vol. 10, pp. 67883–67897, 2022, doi: 10.1109/ACCESS.2022.3186308.
[22] A. A. Soomro *et al.*, "Insights into modern machine learning approaches for bearing fault classification: A systematic literature review," *Results Eng.*, vol. 23, p. 102700, Sep. 2024, doi: 10.1016/j.rineng.2024.102700.
[23] M. K. Senapaty, A. Ray, and N. Padhy, "A Decision Support System for Crop Recommendation Using Machine Learning Classification Algorithms," *Agriculture*, vol. 14, no. 8, p. 1256, Jul. 2024, doi: 10.3390/agriculture14081256.
[24] A. M. Patel and A. Suthar, "AdaBoosted Extra Trees Classifier for Object-Based Multispectral Image Classification of Urban Fringe Area," *Int. J. Image Graph.*, vol. 22, no. 03, p. 2140006, May 2022, doi: 10.1142/S0219467821400064.
[25] D. N. Purba, A. R. Sasabella, and A. T. Misky, "A comparative Study of Boosting Algorithms Optimized by Cuckoo Search on Heart Disease Datasets.".
[26] F. D. Adhinata, Wahyono, and R. Sumiharto, "A comprehensive survey on weed and crop classification using machine learning and deep learning," *Artif. Intell. Agric.*, vol. 13, pp. 45–63, Sep. 2024, doi: 10.1016/j.aiia.2024.06.005.
[27] M. R. Miller, "Multiclass AUC for Comparison of Identification Effectiveness Across Classification Set Sizes," in *2024 IEEE 25th International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, Perth, Australia: IEEE, Jun. 2024, pp. 97–98. doi: 10.1109/WoWMoM60985.2024.00028.
[28] M. Zhu, Y. Zhang, Y. Gong, C. Xu, and Y. Xiang, "Enhancing Credit Card Fraud Detection: A Neural Network and SMOTE Integrated Approach," *J. Theory Pract. Eng. Sci.*, vol. 4, no. 02, pp. 23–30, Feb. 2024, doi: 10.53469/jtpes.2024.04(02).04.
[29] V. Tomar, M. Bansal, and P. Singh, "Metaheuristic Algorithms for Optimization: A Brief Review," in *RAiSE-2023*, MDPI, Mar. 2024, p. 238. doi: 10.3390/engproc2023059238.
[30] L. Li and M. Zhao, "A Novel Komodo Mlipir Algorithm and Its Application in PM2.5 Detection," *Atmosphere*, vol. 13, no. 12, p. 2051, Dec. 2022, doi: 10.3390/atmos13122051.