SCIENCE AND INFORMATION TECHNOLOGY
SINTECH
JOURNAL

SINTECH JOURNAL | ISSN 2598-7305 | E-ISSN 2598-9642
Vol. 8 No 1 – April 2025 | https://s.id/sintechjournal
DOI: https://doi.org/10.31598
Publishing: Prahasta Publisher

A comparative Study of Boosting Algorithms Optimized by Cuckoo Search on Heart Disease Datasets.

Diya Namira Purba[1], Anggi Rodesa Sasabella[2], Afni Tazkiyatul Misky[3]

[1,3] Informatics Major, Astra Polytechnic
Cikarang Selatan, Bekasi Regency, Indonesia
[2]School of Computing, Telkom University
Bandung, Indonesia

e-mail: diya.namira@polytechnic.astra.ac.id[1], anggirodesasasabell@student.telkomuniversity.ac.id[2], afni.tazkiyatul@polytechnic.astra.ac.id[3]

Received : April, 2025	Accepted : April, 2025	Published : April, 2025

Abstract

Heart disease is one of the leading causes of death worldwide. The main factors of heart disease are smoking, alcohol consumption, and obesity. These diseases can affect overall health. Therefore, early detection is important to prevent severe complications. Early diagnosis is often challenging due to asymptomatic nature of heart disease in its initial stages, which leads to higher mortality. As an alternative, machine learning can be implemented for early detection. The purpose of this study is to implement three boosting algorithms: Adaptive Boosting (AB), Extreme Gradient Boosting (XGB), and Gradient Boost (GB). An optimization algorithm, such as the Cuckoo-search Algorithm (CSA), was performed to improve the algorithm's performance. The dataset used in this study are the Cleveland dataset which consists of 303 samples with 13 selected features and IEEE Dataport dataset, which contains 918 samples and 11 features.The evaluation results show that AdaBoost achieved a 0.81 F1 score for the IEEE Dataport dataset, while XGBoost achieved a 0.90 F1 score for the Cleveland dataset. These results indicate that XGBoost performs best for Cleveland dataset, while AdaBoost is more suitable for the IEEE dataport dataset. The Boosting algorithm method optimized by CSA improved accuracy on the IEEE Dataport dataset and maintained stability on the Cleveland dataset. This highlights the effectiveness of Cuckoo search in enhancing model performance. Compared to previous studies, the proposed Boosting Models optimized with CSA exhibited enhanced performance, demonstrating the effectiveness of metaheuristic optimization in heart disease prediction.

Keywords: *heart disease, machine learning, optimization algorithm*

Abstrak

Penyakit jantung adalah salah satu penyebab utama kematian di seluruh dunia. Faktor utama penyakit jantung adalah merokok, konsumsi alkohol, dan obesitas. Penyakit-penyakit ini dapat memengaruhi kesehatan secara keseluruhan. Oleh karena itu, deteksi dini penting dilakukan untuk mencegah komplikasi yang parah. Diagnosis dini seringkali menjadi tantangan karena sifat penyakit jantung yang tidak menunjukkan gejala pada tahap awal, yang dapat menyebabkan kematian lebih tinggi. Sebagai alternatif, pembelajaran mesin dapat diimplementasikan untuk deteksi dini. Tujuan dari penelitian ini adalah untuk *mengimplementasikan tiga algoritma boosting: Adaptive Boosting (AB), Extreme Gradient Boosting (XGB), dan Gradient Boost (GB). Sebuah algoritma optimasi, seperti Cuckoo-search Algorithm (CSA), dilakukan untuk meningkatkan kinerja algoritma. Dataset yang digunakan dalam penelitian in adalah Cleveland dataset terdiri dari 303 sampel dengan 13 fitur terpilih dan IEEE Dataport yang terdiri dari 918 sampel dna 11 fitur. Hasil evaluasi menunjukkan bahwa AdaBoost mencapai skor F1 0,81 untuk dataset IEEE Dataport, sementara XGBoost mencapai skor F1 0,90 untuk dataset Cleveland. Hasil ini menunjukkan bahwa XGBoost merupakan algoritma paling efektive untuk dataset Cleveland, sementara AdaBoost memberikan performansi terbaik pada dataset IEEE dataport. Metode algoritma Boosting yang dioptimalkan oleh CSA meningkatkan akurasi pada dataset IEEE Dataport dan mempertahankan stabilitas pada dataset Cleveland. Temuan ini menunjukkan bahwa algoritma Cuckoo search efektif dalam meningkatkan performa model,. Dibandingkan dengan penelitian sebelumnya, Model Boosting yang diusulkan yang dioptimalkan dengan CSA menunjukkan peningkatan kinerja, yang menunjukkan keefektifan pengoptimalan metaheuristik dalam prediksi penyakit jantung.*

Kata kunci: penyakit jantung, pembelajaran mesin, algoritma Optimasi

1. INTRODUCTION

The heart is responsible for pumping blood to the human body [1]. This organ is crucial and affects the life and health of other organs. If the heart does not function properly, it may cause the other organs to malfunction [2]. Moreover, heart disease is one of the leading causes of death worldwide. The causes of heart disease are smoking, drinking, obesity, etc [3]. Therefore, heart disease has many risk factors and symptoms that need prediction and prevention [3].

There are two types of causes of heart diseases are heart structure and heart function. Heart structure relates to previous heart attacks, and heart function relates to high blood pressure [4]. Furthermore, the symptoms of heart disease are shortness of breath, fatigue, and inflammation in the legs and ankles [5]. Early detection of heart disease can improve a [2]person's survival [6]. The primary objective is to preserve lives by detecting irregularities in cardiac conditions, which will be achieved through the identification and analysis from the heart diseases condition [5]. Additionally, manual detection by pathologist can lead to inaccurate classification [7].

In recent studies, machine learning has been implemented for early disease detection. In 2024, Ahmed et al [6]. proposed a study about heart disease prediction using hybrid machine learning [6]. This study used two datasets: the Cleveland heart disease dataset and the IEEE data port heart diseases [6]. The prediction model was implemented using Hybrid machine learning. The performance of the model shows that the model performs better in disease prediction [6].

Improving heart disease prediction was proposed by Ahmed et al [8]. This model used the UCI machine learning dataset for the dataset [8]. The proposed models have used three methods such as KNN, SVM, and hybrid model. The hybrid model achieves better performance with a value of accuracy of 0.81 [8]. In the same year, Asif et al. proposed enhancing heart disease prediction through ensemble learning techniques [9]. The data used in the model is the Kaggle heart disease dataset [9]. Moreover, the prediction model was performed using the Extra Tree Classifier, XGBoost, and CatBoost [9]. This study shows better performance with a value of accuracy of 0.98 [9].

However, most of the studies mentioned above only compared standard classifiers or hybrid approaches without applying a structured hyperparameter optimization algorithm. In addition, while many used ensemble models such as XGBoost or Random Forest, only a few evaluated the performance improvement with a metaheuristic optimization algorithm.The disease prediction accuracy can be enhanced with machine learning techniques and optimization, as proposed by Chandrasekhar in 2023 [10]. The two datasets used in this study are Cleveland and IEEE dataport datasets [10]. The methods used in the model are random forest, K-nearest neighbour, logistic regression, naive Bayes, gradient boosting, and AdaBoost [10]. This study shows good performance for both datasets, with 0.93 accuracy for the Cleveland dataset and 0.95 accuracy for the Cleveland dataset [10].

The boosting algorithm is commonly used in disease prediction. This algorithm can improve model performance by reducing bias and variance. Additionally, the Cuckoo-search algorithm can be used to optimize the parameters of the model [11]. However, the combination of Cuckoo Search Algorithm and boosting models remains underexplored in the contect of heart disease prediction. Combining the CSA and boosting algorithm can perform better in heart disease prediction [11].

Therefore, this study aims to address the research gap by implementing and evaluating three boosting algorithms: Adaptive Boosting, Gradient Boosting, and Extreme Gradient Boosting, optimized using the Cuckoo Search Algorithm to improve heart disease prediction.

2. RESEARCH METHOD

2.1 Dataset

In this study, two datasets were utilized, Cleveland Heart Disease Dataset from UCI Machine Learning Repository and Heart Disease Dataset from IEEE Dataport [3]. The Cleveland Heart Disease Dataset consists 303 samples with 14 attributes, including the patient demographic such as age, sex, chest pain type, maximum heart rate during exercise and other relevant features such as exercise-induced angina, ST depression, ST segment slope, number of visible major vessels, and thalassemia [3]. It also indicates the presence of heart disease [3]. This dataset also includes binary target attribute indicating the presence or absence of heart disease. The label distribution is presented in Figure 1.

Similarly, the IEEE Dataport heart disease dataset includes 1,190 samples and 12 clinical features commonly used in heart diseases diagnosis, such as age, sex, chest pain, resting blood pressure, cholesterol, fasting blood sugar, resting ECG, maximum heart rate. Additional features include exercise-induced angina, oldpeak, slope, and target (heart disease) [8]. In this dataset, the label ‘0’ represents individuals without heart disease, with the label ‘1’ indicates the presence of heart disease. The total distribution of heart diseases is presented inFigure 2. Both datasets were divided into 80% for training and 20% for testing to evaluate the model’s performance effectively.

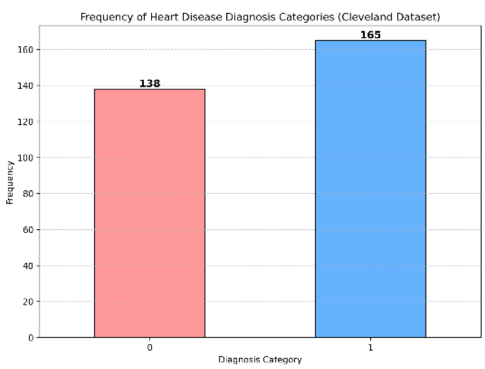

Figure.1. The Distribution of Heart Diseases on Cleveland Dataset

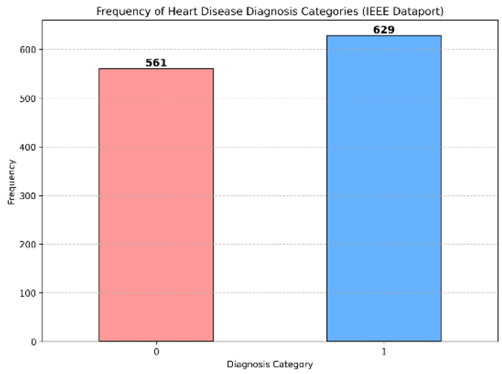

Figure.2. The Distribution of Heart Diseases on IEEE Dataport

2.2 Optimization Methods

The Cuckoo Search (CSA) algorithm, introduced by Christian, is an optimization method inspired by the brood parasitism behavior of cuckoo birds [12]. The main purpose of this algorithm is to determine the optimal solution within the search space $S \subseteq \mathbb{R}^d$ by minimizing the objective function $f(x)$ [12]. There are four steps in this algorithm:

First step is initialization:

$$x_{t=0}^{i,k} = \upsilon(lb^k, ub^k) \qquad (1)$$

In this equation (1),t signifies the iteration count, υ represents a random uniform distribution, $lb^k$and ub^k are the lower and upper boundaries for dimension k, respectively. Additionally, d denotes the total number of dimensions in the problem [12]. For i = 1,…,μ and k = 1,…,d,

Second step is perturbation formulated in equation (2).

$$\underset{x\ t}{\rightarrow}^{i\prime} = \underset{x\ t}{\rightarrow}^{i} + a \underset{r\ t}{\rightarrow}^{i} \qquad (2)$$

A disrupted solution $\underset{x\ t}{\rightarrow}^{i\prime}$ is influenced by a random vector $\underset{r\ t}{\rightarrow}^{i}$ whose components are sampled from a Lévy distributions $\mathcal{L}_y$ with a specified scale parameter. The parameter α determines the extent of the perturbation. For i = 1,…,n [12]

Third step selection is formulated in equation (3).

$$\underset{x\ t\prime}{\rightarrow}^{i} = \begin{cases} \underset{x\ t}{\rightarrow}^{i\prime}, f\left(\underset{x\ t}{\rightarrow}^{i\prime}\right) \ better\ than\ f\left(\underset{x\ t}{\rightarrow}^{i}\right) \\ \underset{x\ t}{\rightarrow}^{i}, otherwise \end{cases} \qquad (3)$$

perform comparisons on each pair $\left(\underset{x\ t}{\rightarrow}^{i}, \underset{x\ t}{\rightarrow}^{i\prime}\right)$, choose the one that has higher quality [12].

Four step recombination formula is presented in equation (4).

$$x_{t+1}^{i,k} = \begin{cases} x_{t'}^{i,k} + \upsilon\llbracket 0,1\rrbracket \cdot \left(x_{t'}^{l^i,k} - x_{t'}^{m^i,k}\right), \\ \quad if\ \upsilon[0,1] \geq P_a\ \forall k, \forall i, \\ \quad x_{t'}^{i,k}, otherwise \end{cases} \quad (4)$$

With probability $1 - P_a$, recombination is performed on the k^{th} component of vector $\vec{x}^{\,i}_{t'}$ by selecting two randomly chosen solution $\vec{x}^{\,l^i}_{k} \in L_t$ and $\vec{x}^{\,m^i}_{t'} \in m_t$, the sets L_t and m_t contain a copy of the population after the third step. The solution $\vec{x}^{\,l^i}_{t'}$ and $\vec{x}^{\,m^i}_{t'}$ are selected from L_t and m_t without replacement, ensuring solution is used once as $\vec{x}^{\,l}_{t'}$ and once as $\vec{x}^{\,m}_{t'}$ if $\vec{x}^{\,l^i}_{t} = \vec{x}^{\,m}_{t} = \vec{x}^{\,i}_{t}$ then the vector $\vec{x}^{\,i}_{t}$ remains unchanged. Once the recombination process is completed, the updated solutions proceed to the next step [12].

The first step is parameter initialization for the algorithm. During the iteration process, the algorithm will try to find the best nest. If it finds the worst nest, the process will continue until it achieves maximum iteration.

In this study, the Cuckoo-search algorithm is implemented to optimize the hyperparameters of boosting algorithms. The main objective is to identify the best parameters to improve prediction. This algorithm's advantage is its ability to escape local optima due to its adaptive size and exploration capabilities. The process of the Cuckoo-search algorithm is presented in the Figure 3.

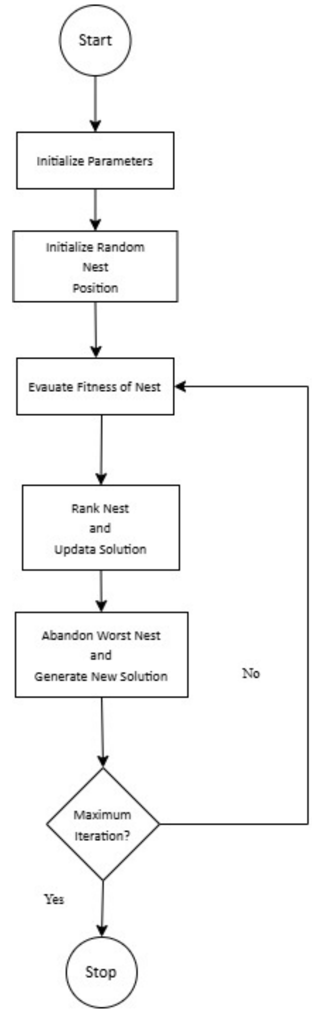

Figure 3. The Process of Cuckoo-search Algorithm adapted from [13]

2.3 Prediction Methods

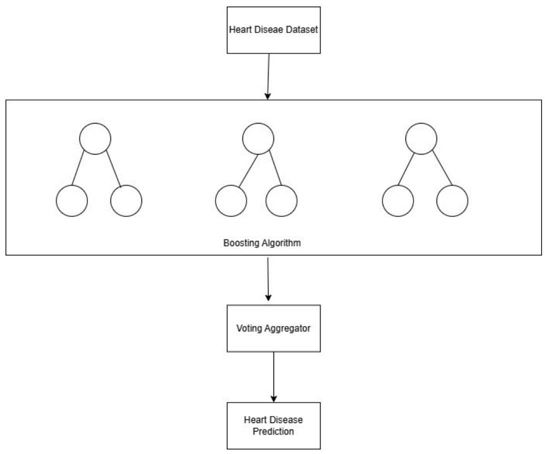

Figure 4. The Boosting Algorithm Architecture Adapted from [14]

Three algorithms were utilized in the prediction model, namely Gradient Boosting (GB), Extreme Gradient Boosting (XGB), and Adaptive Boosting (AB). The prediction method architecture is presented in Figure 4.
Gradient boosting is one of the initial approaches that accurately estimating extreme quantiles, even in intricate predictor spaces and with the presence of high-dimensional noise variables [13]. The gradient boosting method was chosen because in various case studies it has been proven to be able to handle multi-class data effectively [15]. One of the advantages of the Gradient Boosting algorithm is its ability to be used on various types of databases, its robustness in handling outlier data, and its ability to produce accurate predictions [15].
XGBoost is an implementation of Gradient Boosting that applies better regulation to control overfitting [16]. This technique was developed as one of the applications of gradient boosting machines, particularly in regression and classification trees [16]. The basic concept is boosting, which combines predictions from weak learners using additive training methods to form stronger models [17]. XGBoost excels in classification with high performance, speed, and scalability [17]. It optimizes tree learning, handles sparse data, and utilizes parallel computing for efficiency. Additionally, regularization into the loss function helps mitigate overfitting and manage model complexity [16].

The fundamental concept of the AdaBoost assembly technique involves integrating multiple weaker learning models to construct a predictive model, achieved by sequentially training the predictor features [18]. AdaBoost has several advantages that make it effective in many cases [18]. It often provides high prediction accuracy, especially for categorical data, and is easier to use as it requires only a few parameters compared to other algorithms such as SVM [18]. AdaBoost is also flexible with diverse hyperparameter tuning options and can be trained using GPUs to increase processing speed [18]. In addition, AdaBoost can improve the accuracy of weak classifiers, does not require complex preprocessing, and can handle missing data [18]. Now, the algorithm has been extended to binary, text, and image classification, demonstrating its broad flexibility [18].
Boosting algorithm is chosen for heart diseases prediction due to its ability to enhance prediction performance, handle imbalanced data and incorporate built-in regularization to prevent overfitting. To improve their effectiveness, these algorithms will be optimized using the cuckoo search algorithm.
Figure 5 illustrates the workflow of heart disease prediction. The process begins with collecting the heart disease dataset, which is split into a train and test set. The training data is used to train the boosting algorithm by implementing CSA. Finally, accuracy, precision, recall, and F1-score evaluate the model's performance.

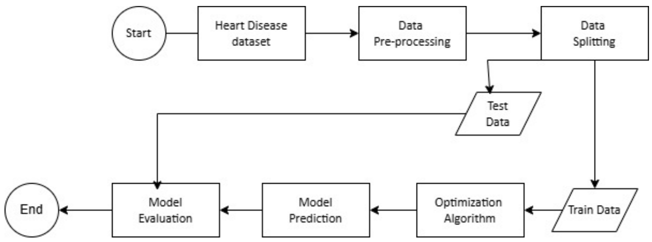

Figure 5. The system design

2.4 Model Evaluation

$$Q = \frac{TP + TN}{TP + FP + TN + FN} \quad (5)$$

$$PR = \frac{TP}{TP + FP} \quad (6)$$

$$RC = \frac{TP}{TP + FN} \quad (7)$$

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (8)$$

The models evaluated by calculated several evaluation metrics. This metric is used to measure the performance of each model. Those parameters are Accuracy (Q), Precision (PR), Recall (RC) and F1-score [19]. Those formulation presented in Equation (5) – (8).

3. RESULT AND DISCUSSION

3.1 Model Development

Table 1: The Parameter Ranges for Optimization Process

METHOD	PARAMETER	RANGES
AB	N_estimators	[50, 100, 150, 200]
	Learning_rate	[0.5, 1.0, 1.5, 2.0]
XGB	N_estimators	[80, 100, 120, 150]

	Learning_rate	[0.2, 0.3, 0.4]
GB	N_estimators	[80, 100, 120, 150]
	Learning_rate	[0.05, 0.1, 0.15. 0.2]

Hyperparameter optimization is used to determine the best parameters for all methods. In this study, the cuckoo-search algorithm is implemented to identify the optimal parameter selection. This algorithm selection process is based on the survival-fittest solution. The ranges of the tuned parameters are presented in Table 1. The AdaBoost Model's best learning rate parameter is 0.5, and n_estimators is 50. A higher learning rate enables the algorithm convergence process to be faster, while the optimal number ensures the model does not overfit and underfit
The XGBoost best parameter is n_estimators is 100, and learning_rate is 0.3. The chosen parameter, a higher number of estimators, enhances prediction performance, while the learning rate ensures effective weight updates without excessive fluctuations.
Lastly, the best parameters for Gradient Boost are a learning rate of 0.05 and n_estimators 80. A moderate number of estimators ensures that the model learns effectively and prevents.

3.2 Model Prediction

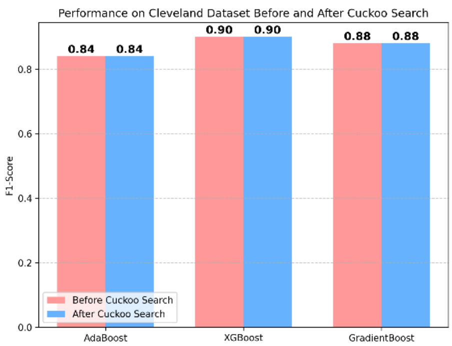

Figure 6. The Performance Improvement Before and After Applying Cuckoo-search (Cleveland Dataset).

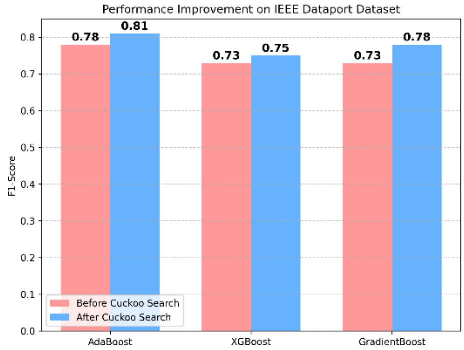

Figure 7. The Performance Improvement Before and After Applying Cuckoo-search (IEEE Dataport Dataset).

Table 2. Performance of the Models Before Applying Cuckoo-search

Dataset	Method	Q	F1	PR	RC
Cleveland	AB	0.84	0.84	0.84	0.84
	XGB	0.90	0.90	0.90	0.90
	GB	0.88	0.88	0.88	0.88
	AB with CSA	0.84	0.84	0.84	0.84
	XGB with CSA	0.90	0.90	0.90	0.90
	GB with CSA	0.88	0.88	0.88	0.88
Dataport	AB	0.78	0.77	0.81	0.78
	XGB	0.73	0.72	0.77	0.73
	GB	0.73	0.73	0.74	0.73
	AB with CSA	0.81	0.81	0.83	0.81
	XGB with CSA	0.75	0.74	0.78	0.75
	GB with CSA	0.78	0.77	0.81	0.78

Table3 Performance Comparison with Previous Study

Author	Method	Q
Bhatt, 2023 [5]	XGB-GridSearch CV	0.86
Proposed Model	XGB-Cuckoo Search	0.90

From Table 2 shows that XGBoost (XGB) achieved the highest performance on the Cleveland dataset with an accuracy, f1-score, precision, and recall of 0.90.. The GB model achieved 0.88 from all evaluation metrics, which indicates good predictive performance. Meanwhile, AdaBoost showed the lowest performance with a score of 0.84 from all evaluation metrics. The performance improvement is presented in Figure 6.

In the IEEE dataport dataset, the model achieved different performance. The AdaBoost model achieved higher performance than the other model with an accuracy of 0.78, while both XGBoost and Gradient Boosting achieved a lower accuracy of 0.73. This indicates that the AdaBoost model can perform the prediction in the dataset than the other model, whereas XGBoost and Gradient Boosting Struggled to perform the prediction. The lower performance can occur because the data distribution is more complex than the Cleveland dataset.

In the Table 2, the performance of AdaBoost, XGBoost, and Gradient Boosting did not improve. This indicates that the model has already reached the optimal performance. XGBoost can perform better because of its ability to regularize and parallel computing. Similarly, the Gradient Boosting ability learning mechanism and AdaBoost adaptive weight adjustment can improve performance [20].

For the IEEE data port dataset, the model showed improvements after cuckoo-search optimization. The AdaBoost model's accuracy improved from 0.78 to 0.81, indicating that the dataset is complex enough for the classification process. XGBoost and Gradient Boosting also improved after the cuckoo-search algorithm indicating that parameter tuning helped the models to adapt to the dataset complexity. The performance improvement is presented in Figure 7.

The cuckoo-search algorithm has an important impact in the parameter optimization process. This method is not similar like the other method such as grid search and random search [12]. The algorithm is designed based on the nesting and reproductive behavior of cuckoo birds. Therefore, Cuckoo search shows the ability as a better optimization method [12].

After comparing with the base models, we also compare the proposed model with previous studies that also predict heart disease with hyperparameter optimization method in Table 3. Based on the result, the proposed model performs better than the previous study with the value 0.90 of F1-score. It can be occurred because the Cuckoo-search ability to iteratively refine parameter selection based on the survival of the fittest principle. In contrast, the compared study used GridSearchCV, which selects hyperparameter based on predefined scoring metric, which may lead to sub optimal selection.

This study's results highlight the importance of dataset characteristics in model performance and the effectiveness of hyperparameter optimization. XGBoost was the best model. Additionally, the application of the cuckoo search algorithm significantly improved model performance.

4. CONCLUSION

We have developed three boosting algorithms, i.e., AdaBoost, Gradient Boosting, and XGBoost, to predict heart disease using Cleveland and IEEE dataport datasets. The hyperparameter tuning process was implemented using Cuckoo Search Optimization, which explores the solution space based on the survival of the fittest principle, allowing for improved parameter selection. The Cuckoo search did not impact the Cleveland dataset, as the models had already reached optimal performance. XGBoost achieved the highest F1-score, 0.90, demonstrating its ability to built-in regularization Although the results remained consistent before and after optimization, the assumption that the model had reached optimal performance was not supported by statistical evaluation. Therefore, further analysis is recommended to confirm this observation.

In contrast, the IEEE data port dataset improved from 0.78 to 0.81. Therefore, Cuckoo-search can optimize the boosting algorithm for heart disease prediction. This result indicates that Cuckoo-search adapts well to the dataset complexities by selecting optimized learning rates and estimators. To further enhance model performance, additional optimization techniques, such as hybrid metaheuristic methods, could be explored in future work.

STATEMENT OF APPRECIATION

In this part, the author can say thank you or appreciation to the interviewees or research funder and/or other parties that have significant role in the article’s writing.

REFERENCES

[1] S. H. Bani Hani and M. M. Ahmad, “Machine-learning Algorithms for Ischemic Heart Disease Prediction:A Systematic Review,” *CCR*, vol. 19, no. 1, p. e090622205797, Jan. 2023, doi: 10.2174/1573403X18666220609123053.

[2] M. Ahmed and I. Husien, “Heart Disease Prediction Using Hybrid Machine Learning: A Brief Review”.

[3] D. Shrestha, “Advanced Machine Learning Techniques for Predicting Heart Disease: A Comparative Analysis Using the Cleveland Heart Disease Dataset”.

[4] “Hybrid Machine Learning Approach for Accurate Heart Disease Prediction,” *IJIES*, vol. 17, no. 4, pp. 728–737, Aug. 2024, doi: 10.22266/ijies2024.0831.55.

[5] C. M. Bhatt, P. Patel, T. Ghetia, and P. L. Mazzeo, “Effective Heart Disease Prediction Using Machine Learning Techniques,” *Algorithms*, vol. 16, no. 2, p. 88, Feb. 2023, doi: 10.3390/a16020088.

[6] K. Eldora, E. Fernando, and W. Winanti, “Comparative Analysis of KNN and Decision Tree Classification Algorithms for Early Stroke Prediction: A Machine Learning Approach,” *J. Inf. Syst. Informatics*, vol. 6, no. 1, pp. 313–338, Mar. 2024, doi: 10.51519/journalisi.v6i1.664.

[7] M. Ahmed and I. Husien, “Heart Disease Prediction Using Hybrid Machine Learning: A Brief Review”.

[8] R. Ahmed, M. Bibi, and S. Syed, “Improving Heart Disease Prediction Accuracy Using a Hybrid Machine Learning Approach: A Comparative study of SVM and KNN Algorithms,” *Int. J. CIM*, vol. 3, no. 1, pp. 49–54, Jun. 2023, doi: 10.54489/ijcim.v3i1.223.

[9] D. Asif, M. Bibi, M. S. Arif, and A. Mukheimer, “Enhancing Heart Disease Prediction through Ensemble Learning Techniques with Hyperparameter Optimization,” *Algorithms*, vol. 16, no. 6, p. 308, Jun. 2023, doi: 10.3390/a16060308.

[10] N. Chandrasekhar and S. Peddakrishna, “Enhancing Heart Disease Prediction Accuracy through Machine Learning Techniques and Optimization,” *Processes*, vol. 11, no. 4, p. 1210, Apr. 2023, doi: 10.3390/pr11041210.

[11] Y. Xiong, Z. Zou, and J. Cheng, “Cuckoo search algorithm based on cloud model and its application,” *Sci Rep*, vol. 13, no. 1, p. 10098, Jun. 2023, doi: 10.1038/s41598-023-37326-3.

[12] C. L. Camacho-Villalón, M. Dorigo, and T. Stützle, “An analysis of why cuckoo search does not bring any novel ideas to optimization,” *Computers & Operations Research*, vol. 142, p. 105747, Jun. 2022, doi: 10.1016/j.cor.2022.105747.

[13] K. Jawad, R. Mahto, A. Das, S. U. Ahmed, R. M. Aziz, and P. Kumar, “Novel Cuckoo Search-Based Metaheuristic Approach for Deep Learning Prediction of Depression,” *Applied Sciences*, vol. 13, no. 9, p. 5322, Apr. 2023, doi: 10.3390/app13095322.

[14] D. Parasar, S. R. Vadalia, S. S. Chavan, K. R. Bhere, F. Nabi, and A. Z. Patel, “Waste Detection and Water Quality Assessment in Aquatic Environments: A Comprehensive Study Using YoloV8 and XGBoost,” Jan. 04, 2024. doi: 10.21203/rs.3.rs-3829799/v1.

[15] Sri Diantika, Hiya Nalatissifa, Riki Supriyadi, Nurlaelatul Maulidah, and Ahmad Fauzi, “IMPLEMENTATION OF MULTI-CLASS GRADIENT BOOSTING TO CLASSIFY ANIMAL SPECIES IN ZOOS: IMPLEMENTASI MULTI-CLASS GRADIENT BOOSTING UNTUK MENGKLASIFIKASIKAN JENIS HEWAN PADA KEBUN BINATANG,” *antivirus*, vol. 17, no. 1, pp. 33–40, Jun. 2023, doi: 10.35457/antivirus.v17i1.2812.

[16] R. Sikander, A. Ghulam, and F. Ali, “XGB-DrugPred: computational prediction of druggable proteins using eXtreme gradient boosting and optimized features set,” *Sci Rep*, vol. 12, no. 1, p. 5505, Apr. 2022, doi: 10.1038/s41598-022-09484-3.

[17] O. Alshboul, A. Shehadeh, G. Almasabha, and A. S. Almuflih, “Extreme Gradient Boosting-Based Machine Learning Approach for Green Building Cost Prediction,” *Sustainability*, vol. 14, no. 11, p. 6651, May 2022, doi: 10.3390/su14116651.

[18] M. Adnan, A. A. S. Alarood, M. I. Uddin, and I. Ur Rehman, “Utilizing grid search cross-validation with adaptive boosting for augmenting performance of machine learning models,” *PeerJ Computer Science*, vol. 8, p. e803, Feb. 2022, doi: 10.7717/peerj-cs.803.

[19] D. Namira Purba, G. Khoerunnisa, and I. Kurniawan, “Firefly Algorithm Under-Sampling (FAUS) in Handling Imbalance Data on Drug Target Interaction Prediction,” in *2024 12th International Conference on Information and Communication Technology (ICoICT)*, Bandung, Indonesia: IEEE, Aug. 2024, pp. 73–78. doi: 10.1109/ICoICT61617.2024.10698356.

[20] J. Velthoen, C. Dombry, J.-J. Cai, and S. Engelke, “Gradient boosting for extreme quantile regression,” *Extremes*, vol. 26, no. 4, pp. 639–667, Dec. 2023, doi: 10.1007/s10687-023-00473-x.