Journal of Electronics, Electromedical Engineering, and Medical Informatics
Multidisciplinary: Rapid Review: Open Access Journal **Vol. 5, No. 4, October 2023, pp: 314-323; eISSN: 2656-8632**

RESEARCH ARTICLE **OPEN ACCESS**

Manuscript received September 27, 2023; revised October 5, 2023; accepted October 5, 2023; date of publication October 20, 2023
Digital Object Identifier (**DOI**): https://doi.org/10.35882/jeemi.v5i4.331

How to cite: Siti Napi'ah, Triando Hamonangan Saragih, Dodon Turianto Nugrahadi, Dwi Kartini, and Friska Abadi, Implementation of Monarch Butterfly Optimization for Feature Selection in Coronary Artery Disease Classification Using Gradient Boosting Decision Tree, Journal of Electronics, Electromedical Engineering, and Medical Informatics, vol. 5, no. 4, pp. 314-323, October 2023.

Implementation of Monarch Butterfly Optimization for Feature Selection in Coronary Artery Disease Classification Using Gradient Boosting Decision Tree

Siti Napi'ah, Triando Hamonangan Saragih, Dodon Turianto Nugrahadi, Dwi Kartini, and Friska Abadi
Computer Science Department, Lambung Mangkurat University, Banjarbaru, South Kalimantan, Indonesia
Corresponding author: Triando Hamonangan Saragih (e-mail: triando.saragih@ulm.ac.id).
This research was supported by Lambung Mangkurat University for providing valuable resources and support.

ABSTRACT Coronary artery disease, a prevalent type of cardiovascular disease, is a significant contributor to premature mortality globally. Employing the classification of coronary artery disease as an early detection measure can have a substantial impact on reducing death rates caused by this ailment. To investigate this, the Z-Alizadeh dataset, consisting of clinical data from patients afflicted with coronary artery disease, was utilized, encompassing a total of 303 data points that comprise 55 predictive attribute features and 1 target attribute feature. For classification, the Gradient Boosting Decision Tree (GBDT) algorithm was chosen, and in addition, a metaheuristic algorithm called monarch butterfly optimization (MBO) was implemented to diminish the number of features. The objective of this study is to compare the performance of GBDT before and after the application of MBO for feature selection. The evaluation of the study's findings involved the utilization of a confusion matrix and the calculation of the area under the curve (AUC). The outcomes demonstrated that GBDT initially attained an accuracy rate of 87.46%, a precision of 83.85%, a recall of 70.37%, and an AUC of 82.09%. After the implementation of MBO, the performance of GBDT improved to an accuracy of 90.26%, a precision of 86.82%, a recall of 80.79%, and an AUC of 87.33% with the selection of 31 features. This improvement in performance leads to the conclusion that MBO effectively addresses the feature selection issue within this particular context.

INDEX TERMS Coronary Artery Disease, Classification, GBDT, Feature Selection, MBO.

I. INTRODUCTION

Coronary artery disease (CAD) is the prevailing form of cardiovascular disease, as per the World Health Organization's (WHO) assessment. It is noteworthy that cardiovascular disease is the primary cause of untimely deaths globally, accounting for 19.9 million fatalities every year, or roughly 31% of total global mortalities. In Indonesia, coronary artery disease stands as the foremost cause of death, ranking second only to stroke at a rate of 26.9% [1]. Moreover, by the year 2030, it is anticipated that coronary artery disease will contribute to over 23 million deaths, constituting approximately 30.5% of worldwide cases [2]. The timely identification of this condition holds great significance in curbing the mortality rate associated with it. Hence, one of the key strategies in achieving early detection is through the implementation of classification techniques.

Classification is a widely utilized technique in the field of data mining. It encompasses various methods, one of which is the Gradient Boosting Decision Tree, abbreviated as GBDT. GBDT is an algorithm that employs boosting and is based on the decision tree proposed by [3]. The selection of the GBDT algorithm was based on its inherent stability and ability to handle class imbalance, rendering it an appropriate option for this particular dataset on coronary artery disease[4].

Furthermore, this particular algorithm utilizes a gradient approach, thereby mitigating the issue of overfitting commonly encountered in conventional decision trees. Consequently, it yields more precise and accurate classifications [5]. According to a study conducted by[6], a comparison was made among various machine learning models, including GBDT, Random Forest, Logic Regression, Support Vector Machine (SVM), Decision Tree, K-Nearest Neighbor (KNN), Neural Network, and XGBoost. The results indicated that GBDT demonstrated superior performance compared to the other models, achieving an AUC value of 0.946 and a precision of 0.778. Moreover, another study conducted by [5] compared GBDT with SVM in the context of recognizing Electroencephalography (EEG) epilepsy. The performance results demonstrated that GBDT surpassed SVM with an accuracy difference of 8%.

The performance of a model is affected by various factors, with data quality being one of the primary determinants. In the field of data mining, the datasets typically collected are characterized by high dimensionality or a large volume of data. However, not all features within these datasets have a significant impact on the classification outcomes. As stated in [7], the selection of relevant features can contribute to the enhancement of classification algorithms' performance. For this study, the Z-Alizadeh Sani dataset obtained from the UCI Machine Learning Repository was utilized, which comprises 56 features. Consequently, it becomes imperative to undertake feature selection in order to eliminate irrelevant features.

Feature selection is a crucial stage in the preprocessing of data in order to identify a suitable subset of features. As stated by [8], the selection of such a subset can alleviate the computational burden and enhance overall performance. [9] has researched the Z-Alizadeh Sani dataset, wherein feature selection was also performed. The Genetic Algorithm (GA) was employed for feature selection, alongside the Genetic Support Vector Machine and Analysis of Variance (GSVMA) algorithm for classification. The most noteworthy outcome achieved was an accuracy of 89.45% utilizing a 10-fold cross-validation technique, with the selection of 31 features.

The Monarch Butterfly Optimization (MBO) algorithm is an additional metaheuristic algorithm that can be employed for feature selection. In a study conducted by [10], the implementation of MBO for feature selection demonstrated a considerably high level of classification accuracy when compared to other metaheuristic algorithms, namely the whale optimization algorithm with simulated annealing (WOASAT), the ant lion optimizer (ALO), the genetic algorithm (GA), and the particle swarm optimization (PSO). The average classification accuracy achieved with MBO was 93% across 18 benchmark datasets. These datasets comprised 7 medical datasets and 11 non-medical datasets. Furthermore, research conducted by [11] also employed MBO for feature selection in conjunction with the Deep Belief Network (DBN) as a classifier, aiming to develop a movie recommendation system. The datasets used in this particular study were sourced from Facebook and Movielens. The evaluation of dataset features was conducted to ascertain the suitability of data with diverse attributes in generating appropriate recommendations. The proposed model yielded Mean Absolute Error (MAE) and Root-Mean-Square Error (RMSE) values of 0.716 and 0.915, respectively. Additionally, precision and recall were determined to be 97.35% and 96.60%, correspondingly.

The authors of this investigation will undertake a research endeavor employing the Gradient Boosting Decision Tree (GBDT) algorithm in order to categorize coronary artery disease, while simultaneously employing the Monarch Butterfly Optimization (MBO) algorithm to diminish extraneous characteristics. The primary objective of this study is to compare the efficacy of the GBDT model both before and after the implementation of MBO feature selection. It is anticipated that the utilization of MBO will enhance the performance of the GBDT model. The contribution of this paper is

1. Introduce the concept and application of GBDT classification technique and feature selection with MBO on medical datasets, especially in the context of coronary artery disease.
2. Provide information on the level of accuracy achieved using the GBDT and MBO algorithms.
3. Assist medical professionals in optimizing decision-making based on data analysis.

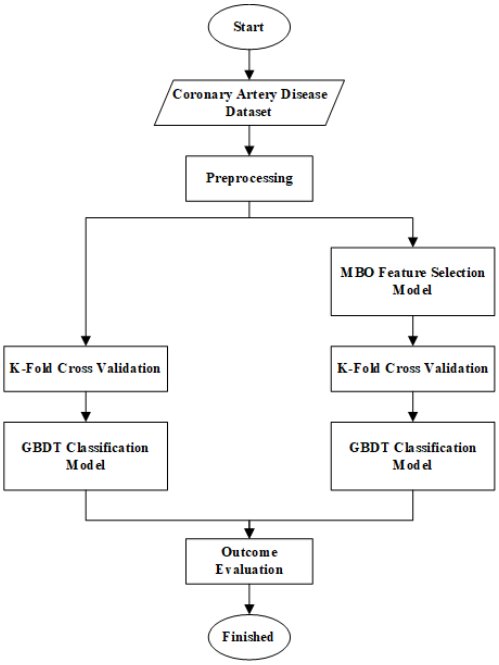

FIGURE 1. Research flowchart

II. METHOD

This section presents the methodology utilized in the study, outlining the data employed, the cross-validation technique employed for data sharing, the label encoding and min-max normalization methods employed for data preprocessing, the gradient boosting decision tree algorithm, the monarch butterfly optimization algorithm, and the evaluation of model performance through the utilization of the Confusion

Matrix and Area Under the Curve (AUC). The schematic representation of this research procedure can be visualized in FIGURE 1.

A. DATA COLLECTION

The Z-Alizadeh sani dataset, which is accessible at https://archive.ics.uci.edu/dataset/412/z+alizadeh+sani, was employed in this investigation. This particular dataset encompasses crucial clinical data associated with patients grappling with coronary artery disease. Precisely, the dataset encompasses 303 rows of information, with a total of 216 patients diagnosed with coronary artery disease (CAD) and 86 patients in a normal physiological state. The dataset boasts 55 characteristics utilized for prediction purposes while possessing one feature designated as the target attribute. The target attribute denotes the patient's diagnostic outcome, divided into two distinct categories: CAD and normal.

The attributes of this dataset are segmented into four distinct attribute categories: demographics, symptoms and examination, laboratory and echocardiography, and electrocardiogram (ECG). Supplementary details regarding the attributes of the Z-Alizadeh Sani dataset can be observed in TABLE 1.

TABLE 1
Features of Alizadeh Sani dataset

Category	Feature Name	Range
Demographics	Age	30 – 86
	Sex	Male, Female
	Weight	48 – 120
	BMI (Body Mass Index kg/m^2)	18.12 –41.90
	DM (Diabetes Mellitus)	0, 1
	HTN (Hypertension)	0, 1
	Current Smoker	0, 1
	Ex-Smoker	0, 1
	FH (Family History)	0, 1
	Obesity	Y (BMI > 25), N (BM I< 25)
	CRF (Chronic Renal Failure)	Y, N
	CVA (cerebrovascular Accident)	Y, N
	Thyroid disease	Y, N
	Airway disease	Y, N
	CHF (Congestive Heart Failure)	Y, N
	DLP (Dyslipidemia)	Y, N
Symptoms and Examination	BP (Blood Pressure)	90 – 190
	PR (Pulse Rate)	50 – 110
	Edema	0, 1
	Weak peripheral pulse	Y, N
	Lung rales	Y, N
	Systolic murmur	Y, N
	Diastolic murmur	Y, N
	Typical chest pain	0, 1
	Dyspnea	Y, N
	Function class	0 – 3
	Atypical	Y, N
	Nonanginal CP (Chest Pain)	Y, N
	Exertional CP (Chest Pain)	N
	lowTH Ang (low-Threshold Angina)	Y, N
Laboratory and Echo	FBS (Fasting Blood Sugar mg/dL)	62 – 400
	CR (Creatine mg/dL)	0.5 – 2.2
	TG (Triglyceride mg/dL)	37 – 1050
	LDL (Low-density lipoprotein mg/dL)	18 – 232
	HDL (High-density lipoprotein)	15 – 111
	BUN (Blood Urea Nitrogen mg/dL)	6 – 52
	ESR (Erythrocyte Sedimentation Rate mm/h)	1 – 90
	HB (Hemoglobin g/dL)	8.9 – 17.6
	K (Potassium mEq/lit)	3.0 – 6.6
	Na (Sodium mEq/lit)	128 – 156
	WBC (White Blood Cell cells/mL)	3700-18000
	Lymph (Lymphocyte %)	7 - 69
	Neut (Neutrophil %)	32 – 89
	PLT (Platelet 1000/mL)	25 – 742
	EF (Ejection Fraction %)	15 – 60
	Region RWMA (Regional Wall Motion Abnormality)	0, 1, 2, 3, 4
	VHD (Vulvar Heart Disease)	Normal, Mild, Moderate, Severe
Electrocardiogram	Q Wave	0, 1
	St elevation	0, 1
	St depression	0, 1
	T inversion	0, 1
	LVH (Left Ventricular Hypertrophy)	Y, N
	Poor R Progression	Y, N
	BBB (Bundle Branch Block)	N, LBBB, RBBB

B. DATA SHARING

Data sharing in this study employs cross-validation techniques with a value of k = 10. The application of cross-validation as a performance evaluation method ensures the dependability of the prediction outcomes. This procedure entails randomly dividing the dataset into K sections. One of these sections serves as validation data to assess the model, while the remaining sections function as training data to educate the classifier. This iterative process is conducted K times, with a distinct validation subset chosen for each iteration [12].

Cross-validation is a crucial process in data analysis, as noted by [13]. It involves the division of the original dataset into two distinct parts: training data and testing data. The term "ten-fold" denotes the value of K, where K is equal to 10 in this particular case. The initial dataset is then divided into ten equal subsets, each serving as either testing or training data in an alternating fashion. This sequential process is repeated for every subset. The visual representation of this data division can be observed in FIGURE 2, which showcases the implementation of 10-fold cross-validation.

FIGURE 2. Data sharing with 10-fold cross-validation

C. PRE-PROCESS DATA

1. LABEL ENCODER

The SciKit-learn library in Python possesses a component known as the label encoder. This encoder serves the purpose of transforming text or categorical data into numerical data within a single column of data automatically [14][15]. An example of using encoder labels on the Z-Alizadeh Sani dataset, on the Sex and BBB features, each data set has categorical variables with set values {"Male", "Fmale"} and {"N", "LBBB", "RBBB"}, then after the encoder label process, it becomes {0, 1} and {0, 1, 2}.

2. MIN-MAX NORMALIZATION

Data normalization is a crucial concern in numerous datasets, including the Z-Alizadeh Sani dataset, due to the presence of variations in numerical feature measurements across different units. Thus, it becomes imperative to conduct data normalization as a preliminary step in data preparation, particularly for tabular data, in order to facilitate the comparison of measurements during model development. Data normalization involves the rescaling of feature values to conform to a standard normal distribution, thereby ensuring uniformity in the input data[16]. Min-max normalization is a frequently employed technique for the normalization of data. This method involves applying a linear transformation to the data to be normalized, as described by Eq. (*1*) [17].

$$x' = \frac{x - \min value}{\max value - \min value} \quad (1)$$

where x' indicates the value of data that has been normalized, x indicates the true value of the data, max value indicates the maximum value of the data, and min value indicates minimum value of the data.

D. GRADIENT BOOSTING DECISION TREE

The Gradient Boosting Decision Tree (GBDT) technique employs a weak classifier known as the Classification and Regression Tree (CART) during each iteration. The core concept behind GBDT is to train a fresh learning machine in a direction that gradually reduces the error rate of the previous learning machine [18]. This iterative process generates a new learning machine that builds upon the knowledge of the previous one. The GBDT algorithm follows a set of steps [19]. Step 1 the learning machine is initialized using Eq. (*2*).

$$F_0(x) = \arg\ min_\rho + \sum_{i=1}^{N} L(\gamma_i, \rho) \quad (2)$$

Step 2 computes the appropriate objective of the regression tree during each iteration. Presented below is the mathematical expression employed for this calculation.

$$\gamma_{m,i} = \left[\frac{\partial L(\gamma_i,\ F(x_i))}{\partial F(x_i)}\right] \quad (3)$$

$$F(x_i) = F_{m-1}(x_i) \quad (4)$$

Step 3 after the initial iteration, one can acquire the optimal base classification by employing the subsequent calculation formula in Eq. (*5*).

$$\alpha_m = \arg\ min + \sum_{i=1}^{N} \left(r_{m,i} - \beta h(x_i; \alpha_m)\right) \quad (5)$$

The calculation of the optimal learning law (ρ_m) is performed through the utilization of the linear optimization technique, and the subsequent updating of the base learning is accomplished by applying Eq. (*6*) that follows.

$$F_m(x) = F_{m-1}(x) + \rho_m h(x_i; \alpha_m) \quad (6)$$

Step 4 involves constructing the most potent learning apparatus.

E. MONARCH BUTTERFLY OPTIMIZATION

The Monarch Butterfly Optimization (MBO) algorithm is a type of population-based algorithm that falls under the category of swarm intelligence algorithms. The inspiration for this algorithm comes from the behavior of certain species, such as bees, butterflies, and similar organisms, which tend to gather together [20]. As stated in [21], the MBO algorithm is characterized by its simplicity and ease of implementation.

The Monarch Butterfly Optimization (MBO) Algorithm starts with a random and uniform population that is called the monarch butterflies population. This population includes the solution candidates of the problem. MBO divides the population into two groups: Land 1 and Land 2. Therefore, the number of monarch butterfly individuals in subpopulations Land 1 and Land 2 are as follows:

$$\text{Land } 1 = NP_1 \times \text{ceil}(p \times NP) \quad (7)$$

$$\text{Land } 2 = \text{NP} - \text{NP}_1 \times (\text{NP}_2) \quad (8)$$

NP represents the total count of populations, while ceil(x) refers to the process of rounding x to the nearest integer that is greater than or equal to x. Additionally, p determines the ratio of monarch butterflies in Land 1 [22]. The migration operator and the butterfly adjustment operator are the two position update operators utilized in the monarch butterfly optimization algorithm [23].

1. MIGRATION OPERATOR

The algorithm generates a new child population by considering the monarch butterfly parents from both Lands. In cases where the parent holds a better value than the generated child, the parent is replaced with the child in order to maintain a constant population count. This ensures the preservation of efficient patents for the subsequent generation. The aforementioned concept can be formulated in the following manner.

$$\chi_{i,k}^{t+1} = \chi_{r1,k}^{t} \quad (9)$$

the notation $\chi_{i,k}^{t+1}$ is used to represent the kth element of the position (χ_i) of monarch butterfly I at generation t+1, Similarly, $\chi_{r1,k}^{t}$ denotes the kth updated element of χ_{r1} for the individual r_1, and the variable t represents the number of current iterations. The individual r_1 is selected randomly from Land 1. If r has a value less than or equal to p, it can be obtained using the following equation:

$$r = p \times \tau \quad (10)$$

whereas τ denotes the duration of migration and r represents an evenly distributed and stochastic variable, conversely, when the value of p is smaller than r, the element k for the newly emerged butterfly is obtained subsequently:

$$\chi_{i,k}^{t+1} = \chi_{r2,k}^{t} \quad (11)$$

the kth updated element of x_{r1}, denoted as $\chi_{r2,k}^{t}$, represents an individual r_1 randomly chosen from Land 2. An interesting advantage of the algorithm lies in its ability to balance the utilization of Land 1 and Land 2. When the value of p is larger, a greater number of populations are selected from Land 1. Conversely, when p has a smaller value, the majority of the population is chosen from Land 2.

2. BUTTERFLY ADJUSTMENT OPERATOR

If the generated child for the monarch butterfly i has a value that is smaller or equal to p, the position is updated according to the following procedure.

$$\chi_{i,k}^{t+1} = \chi_{best,k}^{t} \quad (12)$$

where, $\chi_{best,k}^{t}$ describes the kth individual of x_{best} that gives the best result in the population.

If the value of p is smaller than ρ, the position has been modified according to the following procedure.

$$\chi_{i,k}^{t+1} = \chi_{r3,k}^{t} \quad (13)$$

$$r_3 \in [1,2,\dots,\text{NP}_2] \quad (14)$$

where, $\chi_{r3,k}^{t}$ describes the kth randomly selected member of x_{r3} from Land 2.

During the algorithm, if the rate of adjustment of the butterfly (R_{ba}) proves to be lesser in value than ρ, the position has been effectively modified in the following manner:

$$\chi_{i,k}^{t+1} = \chi_{i,k}^{t} + \alpha \times (dx_k - 0{,}5) \quad (15)$$

The walking step of individual i, represented by dx, can be obtained in the following manner:

$$dx = Levy(x_i^t) \quad (16)$$

Let α denote the weighting coefficient, where it points to the aforementioned value.

$$\alpha = \frac{sm}{t^2} \quad (17)$$

where sm represents the maximum walk step that is passed by a butterfly in one step.

By selecting a large value for parameter α, a lengthy exploration step has been obtained, which amplifies the influence of dx on the application of $\chi_{i,k}^{t+1}$ to the exploration term. Conversely, if α has a small value, α brief exploration step will be taken for $\chi_{i,k}^{t+1}$ leading to an exploitation mechanism[22].

E. EVALUATION

1. CONFUSION MATRIX

In the realm of machine learning, the assessment of classifier performance is typically accomplished through the utilization of a confusion matrix. The confusion matrix, also referred to as a contingency table, possesses the ability to be of arbitrary size. The main diagonal of the matrix indicates the number of instances that have been classified correctly, while the remaining entries correspond to cases that have been misclassified. This matrix encapsulates information regarding both the actual classification outcomes and the predictions generated by the classification system. The evaluation of system performance is typically conducted by leveraging the data encapsulated within this matrix [24]. When considering binary classification, the matrix takes the form of a 2x2 square, as depicted in TABLE 2, in which the columns represent the predictions made by the classifier, and the rows represent the actual values of the class labels. In the presence of imbalanced data, it is customary to assign a positive label to the minority class, while the majority class is designated as negative [25].

TABLE 2
Confusion matrix

	Predicted Positive	**Predicted Negative**
Actual Positive	TP (True Positive)	FN (False Negative)
Actual Negative	FP (False Positive)	TP (True Negative)

TP (True Positive) denotes that the "positive prediction" aligns with the "true condition is positive'. FP (False Positive) signifies that the "positive prediction" does not align with the "true condition is positive." TN (True Negative) signifies that a "negative prediction" aligns with a "negative true state." FN (False Negative) denotes that the "negative prediction" does not coincide with the "true condition is positive" [26].

Presented here are several equations executed on the confusion matrix:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (18)$$

$$PPV = \frac{TP}{TP + FP} \quad (19)$$

$$TPR = \frac{TP}{TP + FN} \quad (20)$$

Equation (19) represents the mathematical expression utilized to derive the Positive Predictive Value (PPV), a metric that is synonymous with recall or sensitivity. Conversely, the determination of the True Positive Rate (TPR), which is also referred to as precision, can be accomplished through the application of equation (20) [27], [28].

2. AREA UNDER CURVE (AUC)

The Area Under Curve (AUC) is a technique employed for the determination of the area beneath the Receiver Operating Characteristic (ROC) curve. The AUC serves as a metric to assess the likelihood that, upon selection of one positive and one negative instance, the classification approach will assign a higher score to the positive instance. Consequently, a higher AUC value corresponds to an improved classification method [29]. The classification quality accuracy, as determined by the AUC value, is presented in TABLE 3.

TABLE 3
Accuracy of classification results based on AUC value

AUC Value	Category
0.90 – 1.00	Excellent Classification
0.80 – 0.90	Good Classification
0.70 – 0.80	Fair Classification
0.60 – 0.70	Poor Classification
0.50 – 0.60	Failure

The calculation of the Area Under the Curve (AUC) is derived from the mean value of the trapezium plane approximations for the graphical representations formed by the True Positive (TP) rate and False Positive (FP) rate [30]. The AUC measure is determined by utilizing equation (*21*) for this calculation.

$$\text{AUC} = \frac{1}{2} + \left(\frac{\text{TP}}{\text{TP} + \text{FN}} + \frac{\text{TN}}{\text{TP} + \text{FP}}\right) \quad (21)$$

III. RESULTS

A. THE RESULTS OF GRADIENT BOOSTING DECISION TREE RESEARCH METHOD

The findings of this investigation demonstrate the outcome of the trials carried out employing the gradient-boosting decision tree approach. The initial trial was executed using the predetermined parameters, whereas the subsequent trial entailed determining the optimal values for the max depth, learning rate, and n-estimator parameters. The outcomes attained with the predetermined parameters (max depth=3, learning rate=0.1, and n-estimator=100) are observable in TABLE 4.

TABLE 4
Results with Default Parameters

Accuracy	Precision	Recall	AUC
84,48 %	77,58 %	66,74 %	78,82 %

1. PARAMETER MAX DEPTH

When examining the max depth parameter, the default values for the learning rate and n-estimator parameters are employed, specifically a learning rate of 0.1 and an n-estimator of 100. The outcomes of the analysis of the max depth parameter can be observed in TABLE 5 provided below.

TABLE 5
Accuracy Results of Max Depth Parameter

Nilai Max Depth	Accuracy	Precision	Recall	AUC
1	85,47 %	80,14 %	68,78 %	80,55 %
2	86,79 %	82,38 %	71,13 %	81,98 %
3	84,48 %	77,58 %	66,74 %	78,82 %
4	86,13 %	80,64 %	70,40 %	81,14 %
5	84,48 %	77,18 %	69,16 %	79,79 %
6	82,48 %	71,88 %	70,83 %	78,97 %
7	78,18 %	65,72 %	55,38 %	71,28 %
8	78,89 %	65,55 %	64,18 %	74,78 %
9	79,55 %	68,10 %	63,76 %	75,07 %
10	77,57 %	63,21 %	63,09 %	73,32 %

Based on the table above, the best test results are in max depth with a value of 2.

2. PARAMETER LEARNING RATE

The optimal value for the max depth parameter in the previous test is employed as the value for testing the learning rate parameter. The default value of 100 is utilized for the n-estimator parameter. The outcomes of the test for the max depth parameter can be observed in TABLE 6 provided below.

TABLE 6
Accuracy Results of Learning Rate Parameter

Nilai Learning Rate	Accuracy	Precision	Recall	AUC
0,1	86,79 %	82,38 %	71,13 %	81,98 %
0,2	85,81 %	79,8 %	69,59 %	80,73 %
0,3	86,46 %	81,7 %	68,8 %	80,8 %
0,4	84,79 %	78,75 %	69,91 %	80,12 %
0,5	84,15 %	77,65 %	67,14 %	78,97 %
0,6	84,78 %	78,91 %	69,05 %	79,91 %
0,7	85,15 %	78,1 %	67,49 %	79,69 %
0,8	86,12 %	81,48 %	70,05 %	81,16 %
0,9	85,46 %	80,18 %	66,97 %	79,65 %
1	83,81 %	75,07 %	68,18 %	78,77 %

Based on the table above, the best test results are on the learning rate with a value of 0,1.

3. PARAMETER N-ESTIMATOR

The n-estimator parameters are examined in order to determine the most optimal values for the max depth and learning rate parameters. These values are derived from the outcomes of the preceding test, representing the best possible options. The test results for the max depth parameter are presented in TABLE 7 for reference.

TABLE 7
Accuracy results of n-estimator parameter

Nilai N-Estimator	Accuracy	Precision	Recall	AUC
50	86,46 %	82,16 %	70,02%	81,41 %
100	86,79 %	82,38 %	71,13 %	81,98 %
150	87,46 %	83,85 %	70,37 %	82,09 %
200	86,12 %	81,4 %	69,35 %	80,84 %
250	86,12 %	82,24 %	67,69 %	80,21 %
300	85,47 %	79,51 %	67,69 %	79,76 %
350	85,46 %	78,18 %	68,94 %	80,14 %
400	84,46 %	74,66 %	68,94 %	79,46 %
450	84,13 %	74,08 %	68,94 %	79,20 %
500	84,13 %	74,33 %	67,83 %	78,91 %

Based on the table above, the best test results are at max depth with a value of 150.

B. THE RESULTS OF GRADIENT BOOSTING DECISION TREE RESEARCH METHOD USING MONARCH BUTTERFLY OPTIMIZATION

To select features, Monarch Butterfly Optimization was used on the Z-Alizadeh Sani dataset. The GBDT results with the best max depth, learning rate, and n-estimator parameter values from the previous experiment were applied to the feature selection process using Monarch Butterfly Optimization. Furthermore, the experiment was conducted 10 times for each parameter and the average was taken. The results of this experiment are shown in TABLE 8 below.

TABLE 8
Gradient boosting decision tree results with monarch butterfly optimization

Pop Size	Iteration	Rata-rata			
		Accuracy	Precision	Recall	AUC
50	100	86,99 %	80,65 %	73,89 %	82,96%
100	200	87,63 %	81,52 %	75,06 %	83,88 %
150	300	87,62 %	81,19 %	74,91 %	83,74 %
200	400	88,94 %	84,29 %	77,58 %	85,04 %
250	500	90,26 %	86,82 %	80,79 %	87,33 %

According to the data presented in TABLE 8, it can be observed that as the values of Pop Size and Iteration increase, there is a corresponding improvement in the performance of the model. The most optimal outcomes are achieved when employing a population size of 250 and conducting 500 iterations. Further information regarding the experiment can be found in the table provided below.

table 9
10 trials using a pop size of 250 and 500 iterations

Run	Selected Feature	Accuracy	Precision	Recall	AUC
1	32	90,74	88,48	82,21	88,29
2	24	89,45	86,08	80,96	86,95
3	30	90,75	88,35	82,34	88,25
4	31	90,85	88,28	82,18	88,34
5	28	89,48	84,99	77,28	85,35
6	31	90,84	88,28	82,18	88,14
7	29	89,46	83,17	77,98	86,41
8	28	89,47	85,56	78,21	85,29
9	33	90,75	86,72	82,35	88,16
10	31	90,81	88,28	82,17	88,15
Average	**29,7**	**90,26%**	**86,82%**	**80,79%**	**87,33%**

In table 9, the optimal outcome is observed in the 6th experimental trial, where 31 distinct characteristics have been chosen. These distinctive attributes are displayed in the subsequent TABLE 10.

TABLE 10
Selected feature

No	Selected Feature	No	Selected Feature
1	Age	17	Function class
2	Sex	18	Atypical
3	BMI	19	Exertional CP
4	HTN	20	St Elevation
5	FH	21	Tinversion
6	Obesity	22	TG
7	CRF	23	LDL
8	CVA	24	HB
9	Airway disease	25	K
10	Thyroid disease	26	WBC
11	CHF	27	Lymph
12	DLP	28	Neut
13	Weak peripheral pulse	29	EF-TTE
14	Lung Rales	30	Region RWMA
15	Typical chest pain	31	VHD
16	Dyspnea		

IV. DISCUSSION

Classification of coronary artery disease encompasses four primary stages: preprocessing, data sharing, classification, and feature selection. The preprocessing stage comprises two processes, specifically label encoder and min-max normalization. The label encoder serves the purpose of converting categorical attributes within the Z-Alizadeh Sani dataset into numerical representations, thereby permitting algorithmic processing. Given the disparate ranges of values within this dataset's features, min-max normalization is employed to rescale the data within the range of 0-1. To segregate the data into training and testing subsets, the cross-validation method is implemented, with consideration given to the relatively modest size of the dataset. By utilizing a k value of 10, the cross-validation method partitions the data into ten distinct subsets.

The classification stage involves the use of the Gradient Boosting Decision Tree algorithm. First, classification was performed using the default parameter values provided by the Scikit-learn library in Python. With the default parameters, the results obtained are shown in TABLE 4. Second, classification is performed by finding the best parameter values for max depth, learning rate, and n-estimator. Max depth controls the depth of the trees, the learning rate regulates the contribution of each tree to the ensemble model, and the n-estimator indicates the number of trees used. By reducing the max depth value and increasing the n-estimator, the model performance improves as shown in TABLE 7. With an AUC value of 82.09% based on TABLE 3, the model performance is categorized as good classification. The Gradient Boosting Decision Tree with the best parameter values will then be used to evaluate the feature selection process.

Butterfly Optimization was chosen for feature selection. Experiments were conducted 10 times on each combination of population size and iteration parameters, then the average was taken. Based on TABLE 8, the best performance is at a population size of 250 and iteration of 500. The selected features can be seen in TABLE 10. Monarch butterfly

optimization can reduce the features of the Z-Alizadeh Sani dataset by about 40% - 56% features. The model comparison before and after applying feature selection based on TABLE 7 and table 9 is presented in FIGURE 3.

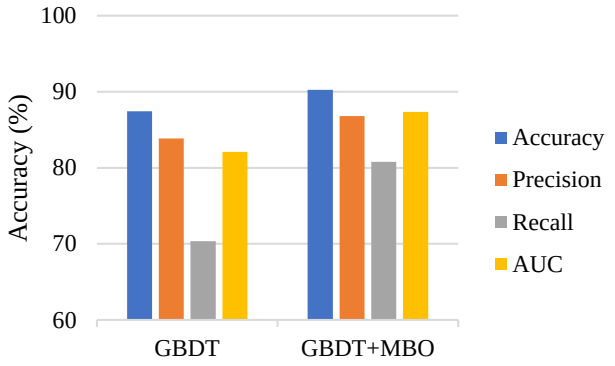

FIGURE 3. Comparison of models without feature selection and with feature selection

The utilization of Monarch Butterfly Optimization in the realm of feature selection has contributed to the enhancement of accuracy when employing the Gradient Boosting Decision Tree algorithm to classify coronary artery disease. The visual representation depicted in FIGURE 3, elucidates that the accuracy, precision, recall, and AUC all exhibited improvement after the execution of feature selection. In comparison with previous studies, the results of this study show that the Gradient Boosting Decision Tree (GDBT) method for classification with feature selection using Monarch Butterfly Optimization (MBO) produces better performance in classifying coronary artery disease datasets. This is because the model proposed in this study managed to improve the accuracy rate compared to previous studies that used different classification and feature selection methods. A previous study [9] that proposed a hybrid machine learning model known as Genetic Support Vector Machine And Analysis Of Variance (GSVMA) and combined it with feature selection using a genetic algorithm, achieved the highest accuracy rate of 89.45%. Therefore, it can be concluded that in this context, the basic GDBT algorithm is effective in improving the quality of coronary artery disease classification compared to the hybrid model proposed in the previous study.

However, the weakness in this study lies in the use of an unbalanced dataset between CAD and normal classes. In many cases, class imbalance relates to real-world problems where the minority class may be an important or potentially dangerous case. Models trained on data with unbalanced classes are less able to generalize well to more balanced data. Such models risk overfitting on majority data and have difficulty adapting when balanced data is used. Therefore, for future research, it is recommended to use dataset imbalance handling techniques such as oversampling, and undersampling, or synthetic methods such as SMOTE to help improve model performance and ensure more accurate research results. Nonetheless, despite these limitations, this study successfully demonstrated that combining GBDT with MBO can improve the performance of coronary artery disease classification, in terms of accuracy, precision, recall, and AUC.

The results of this study show that using MBO as a feature selection method can improve the performance of the GBDT model in classifying coronary artery disease. This indicates the potential to reduce the dimension of irrelevant data and allow the model to focus more on essential features. The implications of these findings include a significant impact in medical practice and public health. With the model's ability to extract patterns and relationships in data, early diagnosis, and appropriate treatment can improve patient prognosis. Such implications highlight the important role of technology and data analytics in medical and health sciences, with the potential to improve the diagnosis, treatment, and prevention of serious diseases such as coronary artery disease.

V. CONCLUSION

This investigation introduces a Metaheuristic Monarch Butterfly Optimization (MBO) algorithm designed specifically for feature selection. The utilization of MBO has proven to enhance the performance of the Gradient Boosting Decision Tree (GBDT) significantly. The findings demonstrate that GBDT achieved an accuracy level of 87.46%, a precision value of 83.85%, a recall rate of 70.37%, and an AUC (Area Under the Curve) score of 82.09% in the classification of coronary artery disease. Subsequently, following the implementation of feature selection utilizing MBO, there was a noticeable improvement in various metrics. Specifically, there was an increase in accuracy by 2.8%, precision by 2.97%, recall by 10.42%, and AUC by 5.24%. Consequently, GBDT-MBO achieved an accuracy rate of 90.26%, a precision rate of 86.82%, a recall rate of 80.79%, and an AUC score of 87.33% with a selection of 31 features. Based on the evident enhancement in performance, it can be deduced that MBO is indeed an effective technique employed for feature selection. For future research endeavors, it is recommended to consider the utilization of a data balancing method before its combination with MBO. Moreover, further exploration can be conducted by combining MBO with alternative classification algorithms, such as XGBoost.

REFERENCES

[1] M. Wang, *Coronary Artery Disease: Therapeutics and Drug Discovery*, vol. 1177. 2020. doi: 10.1007/978-981-15-2517-9.
[2] M. Sayadi, V. Varadarajan, F. Sadoughi, S. Chopannejad, and M. Langarizadeh, "A Machine Learning Model for Detection of Coronary Artery Disease Using Noninvasive Clinical Parameters," *Life*, vol. 12, no. 11, 2022, doi: 10.3390/life12111933.
[3] J. H. Friedman, "Greedy function approximation: A gradient boosting machine," *Ann. Stat.*, vol. 29, no. 5, pp. 1189–1232, 2001, doi: 10.1214/aos/1013203451.
[4] M. Zheng, F. Wang, X. Hu, Y. Miao, H. Cao, and M. Tang, "A Method for Analyzing the Performance Impact of Imbalanced Binary Data on Machine Learning Models," *Axioms*, vol. 11, no. 11, 2022, doi: 10.3390/axioms11110607.
[5] J. He, L. Yang, D. Liu, and Z. Song, "Automatic Recognition of High-

Density Epileptic EEG Using Support Vector Machine and Gradient-Boosting Decision Tree," *Brain Sci.*, vol. 12, no. 9, 2022, doi: 10.3390/brainsci12091197.
[6] Z. Ye *et al*., "The prediction of in-hospital mortality in chronic kidney disease patients with coronary artery disease using machine learning models," *Eur. J. Med. Res.*, vol. 28, no. 1, pp. 1–13, 2023, doi: 10.1186/s40001-023-00995-x.
[7] H. Liu and R. Setiono, "Feature selection via discretization," *IEEE Trans. Knowl. Data Eng.*, vol. 9, no. 4, pp. 642–645, 1997, doi: 10.1109/69.617056.
[8] G. Forman, "An extensive empirical study of feature selection metrics for text classification," *J. Mach. Learn. Res.*, vol. 3, pp. 1289–1305, 2003.
[9] J. Hassannataj Joloudari *et al.*, "GSVMA: A Genetic Support Vector Machine ANOVA Method for CAD Diagnosis," *Front. Cardiovasc. Med.*, vol. 8, pp. 1–14, 2022, doi: 10.3389/fcvm.2021.760178.
[10] M. Alweshah, S. Al Khalaileh, B. B. Gupta, A. Almomani, A. I. Hammouri, and M. A. Al-Betar, "The monarch butterfly optimization algorithm for solving feature selection problems," *Neural Comput. Appl.*, vol. 34, no. 14, pp. 11267–11281, 2022, doi: 10.1007/s00521-020-05210-0.
[11] S. Sridhar, D. Dhanasekaran, and G. C. P. Latha, "Content-Based Movie Recommendation System Using MBO with DBN," *Intell. Autom. Soft Comput.*, vol. 35, no. 3, pp. 3241–3257, 2023, doi: 10.32604/iasc.2023.030361.
[12] C. Khammassi and S. Krichen, "A GA-LR wrapper approach for feature selection in network intrusion detection," *Comput. Secur.*, vol. 70, no. June, pp. 255–277, 2017, doi: 10.1016/j.cose.2017.06.005.
[13] H. Mo, H. Sun, J. Liu, and S. Wei, "Developing window behavior models for residential buildings using XGBoost algorithm," *Energy Build.*, vol. 205, p. 109564, 2019, doi: 10.1016/j.enbuild.2019.109564.
[14] P. Joshi, *Python Machine Learning Cookbook*. Packt Publishing, 2016. [Online]. Available: https://books.google.co.id/books?id=EwNwDQAAQBAJ
[15] J. T. Hancock and T. M. Khoshgoftaar, "Survey on categorical data for neural networks," *J. Big Data*, vol. 7, no. 1, 2020, doi: 10.1186/s40537-020-00305-w.
[16] T. M. Alam *et al.*, "An investigation of credit card default prediction in the imbalanced datasets," *IEEE Access*, vol. 8, pp. 201173–201198, 2020, doi: 10.1109/ACCESS.2020.3033784.
[17] S. Sinsomboonthong, "Performance Comparison of New Adjusted Min-Max with Decimal Scaling and Statistical Column Normalization Methods for Artificial Neural Network Classification," *Int. J. Math. Math. Sci.*, vol. 2022, 2022, doi: 10.1155/2022/3584406.
[18] X. Zhao, X. Li, S. Sun, and X. Jia, "Secure and Efficient Federated Gradient Boosting Decision Trees," *Appl. Sci.*, vol. 13, no. 7, 2023, doi: 10.3390/app13074283.
[19] L. Ma, H. Xiao, J. Tao, T. Zheng, and H. Zhang, "An intelligent approach for reservoir quality evaluation in tight sandstone reservoir using gradient boosting decision tree algorithm," *Open Geosci.*, vol. 14, no. 1, pp. 629–645, 2022, doi: 10.1515/geo-2022-0354.
[20] Y. Feng, S. Deb, G. G. Wang, and A. H. Alavi, "Monarch butterfly optimization: A comprehensive review," *Expert Syst. Appl.*, vol. 168, p. 114418, 2021, doi: 10.1016/j.eswa.2020.114418.
[21] G. G. Wang, S. Deb, and Z. Cui, "Monarch butterfly optimization," *Neural Comput. Appl.*, vol. 31, no. 7, pp. 1995–2014, 2019, doi: 10.1007/s00521-015-1923-y.
[22] S. Bao *et al.*, "A new method for optimal parameters identification of a PEMFC using an improved version of Monarch Butterfly Optimization Algorithm," *Int. J. Hydrogen Energy*, vol. 45, no. 35, pp. 17882–17892, 2020, doi: 10.1016/j.ijhydene.2020.04.256.
[23] D. L. Namburi and M. Satya Sai Ram, "Speaker Recognition Based on Mutated Monarch Butterfly Optimization Configured Artificial Neural Network," *Int. J. Electr. Comput. Eng. Syst.*, vol. 13, no. 9, pp. 767–775, 2022, doi: 10.32985/ijeces.13.9.5.
[24] a. K. Santra and C. J. Christy, "Genetic Algorithm and Confusion Matrix for Document Clustering," *Int. J. Comput. Sci.*, vol. 9, no. 1, pp. 322–328, 2012, [Online]. Available: http://ijcsi.org/papers/IJCSI-9-1-2-322-328.pdf
[25] M. Bekkar, H. K. Djemaa, and T. A. Alitouche, "Evaluation Measures for Models Assessment over Imbalanced Data Sets," *J. Inf. Eng. Appl.*, vol. 3, no. 10, pp. 27–38, 2013, [Online]. Available: http://www.iiste.org/Journals/index.php/JIEA/article/view/7633
[26] M. Te Wu, "Confusion matrix and minimum cross-entropy metrics based motion recognition system in the classroom," *Sci. Rep.*, vol. 12, no. 1, pp. 1–10, 2022, doi: 10.1038/s41598-022-07137-z.
[27] I. Markoulidakis, I. Rallis, I. Georgoulas, G. Kopsiaftis, A. Doulamis, and N. Doulamis, "Multiclass Confusion Matrix Reduction Method and Its Application on Net Promoter Score Classification Problem," *Technologies*, vol. 9, no. 4, 2021, doi: 10.3390/technologies9040081.
[28] D. Chicco, N. Tötsch, and G. Jurman, "The matthews correlation coefficient (Mcc) is more reliable than balanced accuracy, bookmaker informedness, and markedness in two-class confusion matrix evaluation," *BioData Min.*, vol. 14, pp. 1–22, 2021, doi: 10.1186/s13040-021-00244-z.
[29] B. Robert and E. B. Brown, *Data Mining: Concep, models and techniques*, no. 1. 2004. [Online]. Available: https://link.springer.com/book/10.1007/978-3-642-19721-5
[30] R. Dubey, J. Zhou, Y. Wang, P. M. Thompson, and J. Ye, "Analysis of sampling techniques for imbalanced data: An n=648 ADNI study," *Neuroimage*, vol. 87, pp. 220–241, 2014, doi: 10.1016/j.neuroimage.2013.10.005.

AUTHOR BIOGRAPHY

Siti Napi'ah originated in Kandangan, Hulu Sungai Selatan, South Kalimantan. After graduating from high school, she continued her education to the university level. Since 2018, she has been pursuing her academic world as a student of the Computer Science Department at Lambung Mangkurat University. Her current area of research lies within the realm of data science. Currently, she is completing her final project which requires research centered on the classification of coronary artery disease.

Triando Hamonangan Saragih is a lecturer in Department of Computer Science, Lambung Mangkurat University. His research interest is centered on Data Science and Computer Networkin and Data Science. He completed his bachelor's degree in Informatics at Brawijaya University, Malang in 2016. After that, he pursued a master's degree in Computer Science Brawijaya University, Malang in 2018. The research field he is involved in is Data Science.

Dodon Turianto Nugrahadi is a lecturer in Department of Computer Science, Lambung Mangkurat University. His research interest is centered on Data Science and Computer Networking. He completed his bachelor's degree in Informatics Engineering at UK. Petra, Surabaya in 2004. After that, he pursued a master's degree in Information Engineering at Gajah Mada University, Yogyakarta in 2009. His current area of research revolves around Network, Data Science, IoT, and QoS.

Dwi Kartini received her bachelor's and master's degrees in computer science from the Faculty of Computer Science, Putra Indonesia University "YPTK" Padang, Indonesia. Her research interests include the applications of Artificial Intelligence and Data Mining. She is an assistant professor in the Department of Computer Science, Faculty of Mathematics and Natural Sciences, Lambung Mangkurat University in Banjarbaru, Indonesia.

Friska Abadi finished her bachelor's degree in Computer Science from Universitas Lambung Mangakurat in 2011. Subsequently, in 2016, she obtained her master's degree from the Department of Informatics at STIMIK Amikom, Yogyakarta. Following that, she joined Universitas Lambung Mangakurat as a lecturer in Computer Science. Currently, she holds the position of head of the software engineering laboratory. Her current area of research revolves around software engineering.