

# Developing Heart Attack Prediction Model Based on Hyperparameter Tuning and Machine Learning Approach

**Authors: Omar Shakir Hasan<sup>1</sup>; Ibrahim Ahmed Saleh<sup>2</sup>**

<sup>1</sup>*computer science, College of Computer Science and Mathematics, Mosul University, Mosul,  
Iraq*

<sup>2</sup>*software engineering, College of Computer Science and Mathematics, Mosul University,  
Mosul, Iraq*

*Email: omarshakir06@gmail.com*

## **ABSTRACT.**

Heart attacks pose a severe health threat to human society in the world. If the heart attack is discovered in its early stages, it can be treated and save souls from death and not occurring severe complications. The machine learning approach can predict the precision of the heart attack risk to take the appropriate intervention measures in time and help medical institutions. The heart attack data sets are excessively unbalanced, leading to the bias of machine learning models. To model, an unbalanced dataset needs exceptional methods to reduce model bias. Heart attack constitutes a minority category of clinical data sets. Finding hyperparameters is a critical task in finding the best model. This paper proposes a concept of boosting (Catboost) to develop a heart attack prediction model for this type of data and use the Hyperopt method to find hyperparameters to reduce model bias. The study concluded that Catboost is better at predication the target minority group compared to (decision tree, naive Bayes, and Xgboost)and finding hyperparameters using Hyperopt increases model accuracy and reduces bias

**KEYWORDS:** Machine Learning, Imbalance Data; Hyperparameter Tuning; Catboost, Xgboost,; Naive Bayes, Decision Trees.

## **I. INTRODUCTION**

In recent years, with the rapid development of machine learning, a large number of electronic medical records have been generated in the medical field. Machine learning can change the way clinicians diagnose and manage patients[1]. If developed and used correctly, Machine learning diagnosis is expected to be faster, more convenient, and more accurate than any previous diagnosis technology. However, the development of Machine learning in the medical field also faces performance problems of the traditional algorithms model[2], and the accuracy still needs further improvement to predicate disease. Medical research is the most important field, and the amount of data created is enormous and very useful. It is thus essential to seek the most efficient data processing system for patient safety.

The cause of a heart attack is blood vessels became narrow or blocked when blood and oxygen are supplied to the heart, resulting in sudden death[3]. The persistently high morbidity, disability, and mortality rates have brought heavy economic and psychological burdens to society, families, and patients. If the diagnosis of the heart attack at early, it will save a life[4]. Building a predictive model for predicting heart attacks based on patient medical data has challenges in constructing the final model. These challenges focus on the nature of the data, the problem of imbalance, finding an appropriate machine learning model, finding hyperparameters, and evaluating the final performance of the specified model. A set of measurement tools related to the subject of the paper and the efficiency of algorithms were used to determine the model efficiency.

---

*Corresponding author.*

*E- mail address: omarshakir06@gmail.com*

*Omar Shakir Hasan; Ibrahim Ahmed Saleh, Volume 10 Issue 9, pp 1-12, September 2022*

## 2. RELATED WORKS

Many studies were carried out to predict heart attacks based on statistical methods. Recently, it has become common practice to use intelligent methods to predict heart attacks.

Hisham Khdaif, [5]. Introduced four classifiers: KNN logistic regression, support vectors machine, and Multilayer Perceptron Neural Networks applied to 462 samples and nine attributes from the South African Heart Disease data retrieved on the KEEL repository. The dataset contains 302 records of healthy people and 160 records of patients with coronary artery disease (CAD), which are unbalanced. The K-means Method, combined with the synthetic minority oversampling technique (SMOTE), was applied to solve the imbalanced classification issue. The results showed SVM had the highest performance with an accuracy of 0.781[30].

Chetan Sharma et al, [6]. The paper aims to predict a heart attack. Used two different datasets. The first one was obtained from the "Cleveland Clinic Foundation," and the second dataset is taken from the publicly available platform combining five other datasets named Heart Disease Dataset (Comprehensive). Both datasets contain 76 attributes. Data preprocessing was used to remove noise and mess values and applied Naive Bayes and NB tree on both data sets. The result shows that Naive Bayes achieved higher accuracy, reached 80.58, and got the highest precision of 80.6% on the first dataset. Naive Bayes achieved higher accuracy of 80.70% and higher precision of 83.7% on the second dataset[33].

Pabitra Kumar Bhunia et al,[7]. The author used a heart csv dataset containing 14 attributes and 303 instances. Divided the dataset into the training phase (90 %) and the testing phase (10 %). Random Forest, Support Vector Machine logistic regression, KNN classifier, and decision tree was used to predict heart disease. The findings revealed that the Random Forest and Support Vector Machine had a maximum accuracy of 90.32 %. At the same time, logistic regression, KNN classifier, and decision tree achieved an accuracy of 87.09 %, 70.96 %, and 83.87 %, respectively[34].

Aniruddha Dutta et.al ,[8]. Dopted a two-step first, employ least absolute shrinkage and selection operator (LASSO) based feature weight assessment followed by majority-voting based identification of important features applied on NHANES dataset with high unbalance. Next used is CNN for prediction. Accuracy reached 77% for the presence of CHD and 81.8% for the absence of CHD cases on testing data, and 85.70% for the total dataset. [155]

Nitten S. Rajliwall et all , [9]. Models of Naive Bayes, bagging, decision tree, logistic regression, k nearest neighbors, random forest, support vector machine, and the neural network have been configured and used a high unbalanced dataset called (NHANES).The accuracy reached of 95.7%, 96.5% , 97.6%,96.4% ,80.8%, 98.5%, 95.4%,98.8% respectively[156].

H. Benjamin Fredrick David et al,[10]. StatLog dataset available in the UCI machine learning repository with Balance 1:1 has been used.. Suggested naive Bayes, decision tree, and random forest for classification based on cross-validation, and naive Bayes, decision tree, random forest for classification based on a percentage split precession for NB, decision tree, and random forest. The Precession of NB, DT, and random forest based on Cross-Validation is 0.714, 0.714, and 0.73, respectively. The Precession NB, DT, and random forest based on percentage split are 0.313, DT, 0.788,0.804, respectively [160].

Walaa Adel Mahmoud et al, [11].used the Framingham dataset with an unbalanced 1:3.Suggested an imputation-mean approach to handling missing and outlier data values and different classification algorithms KNN, SVM, DT, LR, and RF algorithms. The accuracy was 83.95, 84.5, 84.82, 84.89, and 85.05 % for the KNN, SVM, DT, LR, and RF algorithms. The RF algorithm successfully predicts 85.05 %[161].

## 3. MATERIALS AND METHODS

### 3.1 Unbalancing Dataset

Imbalanced datasets have a considerable disparity in the number of samples in each dataset category. It is assumed that the number of samples in the positive class is much larger than in the negative class[12]. An imbalanced dataset is a problem for prediction models where the distribution of dataset samples is skewed or biased. The data distribution can be one or a few samples called minority class or hundreds, thousands, or millions of samples called majority class[13]. Imbalanced classifications make machine learning system predictive modeling difficult. As a result, models with low prediction accuracy arise, especially for the minority class. This is because the minority class is larger than the majority class, and hence the problem is more prone to

categorization errors for the minority class[14]

### 3.2. Heart Attack Dataset

NHANES dataset was used in this paper, and the dataset contains 37,080 records of diverse individuals. (1300) people have coronary heart and (35780) non-coronary heart(non-heart attack). Each person has 50 features and one target value. The data set is unbalanced, as people with coronary heart disease( heart attack ) represent 3.5% of the total data.[15]

### 3.3. Data Analysis

The data set was analyzed in terms of data distribution in each feature, finding missing values, variable types, and the importance of each feature for the algorithms used to find the correlation between the statistical features and characteristics of the data set. Table (1) represents the statistics description of the totality of the data.

**Table 1.** Dataset Statistics

Number of Features	51
Number of observations	62837
Missing data	0
Duplicate instance	30383
Duplicate instance (%)	48.40%
Total memory usage	24.4 MiB
Average instance size in memory	408.0 B
<b>Variable Types</b>	
Numeric data	43
Categorical data	8

### 3.4. Training Machine Learning Algorithms and Model Building

After initializing the data and analyzing it, machine learning models are trained to determine the models efficiency as a whole. In this paper, four machine learning algorithms (naive Bayes, decision trees, XGboost, CatBoost) were trained and tested using 80% and 20% of the total data, respectively. Below is a review of the algorithms used.

#### Naive Bayes

It is referred to as the Naive Bayes classifier since it is a supervised learning method for classification problems. It is assumed that the components are independent and that there is no association between the elements in a Naive Bayes model. In actual life, however, this is not always the case. The naïve assumption that the characteristics are useless is why the algorithm is referred to as "naive" in the first place[16][17]. The Bayes theorem is the idea behind the Naive Bayes algorithm. The following is an equation of Naive Bayes :

$$p(A/B) = \frac{P(B/A) \cdot P(A)}{P(B)} \quad (1)$$

$p(A | B)$ : the probability of event A for a given event B

$p(B | A)$ : Given event A, the probability of event B

$p(A)$ : the probability of event A

$p(B)$ : the probability of event B

#### Decision Tree

A classification tree is a kind of supervised learning used in computer science. In most cases, a decision tree comprises a root node, multiple internal nodes, and numerous leaf nodes, among other things[18]. The leaf nodes correspond to decision outcomes, and each other node relates to an attribute. The root node contains all samples in their completeness. One of the important tasks is to extract a series of rules for the knowledge information contained in the data[19]. The feature selection for each node depends on the following measurements:

Information entropy: Information entropy is an index to measure the degree of disorder of features; that is,

to measure the impurity of information intuitively (impurities features).

$$E(S) = \sum_{i=1}^c -P_i \log_2 P_i \quad (2)$$

Where  $S$  is the total number of features,  $c$  is the number of classes, and  $p(i)$  indicates the probability of records belonging to that class.

Information gain: Information gain measures the change in information entropy between independent attributes. It tries to estimate the information contained in each attribute.

$$Gain(S, X) = Entropy(S) - Entropy(S, X) \quad (3)$$

Among them,  $Gain(S, X)$  is the information gain of feature  $X$ .  $Entropy(S)$  is the information entropy of the entire dataset, and the second term  $entropy(S, X)$  is the information entropy of feature  $X$ .

The attribute with the greatest information gain is ranked as the division node by ranking each node information gain. All other characteristics are filtered out when utilizing information entropy as a node selection method.

Classification error:

$$\text{Classification error}(T) = 1 - \left[ p \left( \frac{i}{t} \right) \right] \quad (4)$$

$p(i/t)$  indicates the probability of records belonging to that class

#### XGBoost algorithm

Extreme Gradient Boosting (XGBoost) is an ensemble learning improvement approach based on decision trees that combines weak base classifiers into stronger ones. It is an improved GBDT algorithm and an iterative decision tree algorithm. The algorithm consists of multiple decision trees, and the conclusions of all trees are added together as the final [20]. XGBoost may be used to prevent overfitting and solve a variety of problems. [21].

#### CatBoost Algorithm

CatBoost is a new gradient boosting algorithm proposed by Prokhorenkova et al. (2018) and Dorogush . that can be successfully applied to diverse types and formats of data. It supports both numerical and categorical features. CatBoost uses binary decision trees as the base predictor. Uses the same features to split learning instances into the left and the correct partitions for each tree level, solving the overfitting problem [22]. A perfectly symmetrical tree model is adopted to reduce overfitting and improve the accuracy and generalizability of the algorithm, unlike the original boosting algorithms that train every residual error. The CatBoost algorithm speeds training through feature combinations that combine variables with the same information gain [23]. Furthermore, contrary to the conventional learning classifiers, it is the key motivation behind utilizing its intelligence in the current paper.

The suggested model was evaluated based on several scales: accuracy and precision, The most critical things to consider are precision. Here are some assumptions measuring tools in further detail [24]. Given that there are two classes,  $p$ , and  $n$ , to be classed, let us assume

TP: True Positives.

TN: True Negatives.

FP: False Positives.

FN: False Negatives.

True Positives (TP: True Positives.): Count the number of positive samples that the model correctly classified

as positive.

False Positive (FP: false positive): Counting the number of negative samples that the model incorrectly classified as positive.

True Negative (TN: true negative): Counting the number of negative samples that the model correctly classified

as negative.

False Negative (FN: false negative): Counting the number of positive samples that the model incorrectly classified as negative.

Accuracy

The correct prediction (TP, TN) is divided by all the results. The formula is:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (8)$$

It is impossible to assess the model performance based on its accuracy when the data set is imbalanced. That is when the number of positive and negative samples is considerably different. Precision is a more reliable indication when dealing with imbalanced data sets[24].

#### Precision

When the data set is unbalanced, The number of samples in the positive class is much larger than in the negative class. In that case, the model accuracy alone cannot be used to evaluate the model performance. Precision is a better indicator for a model measuring an unbalanced data set. It used the influence of the prediction degree of accuracy as a positive example in all classes in which the prediction is a positive example[25].

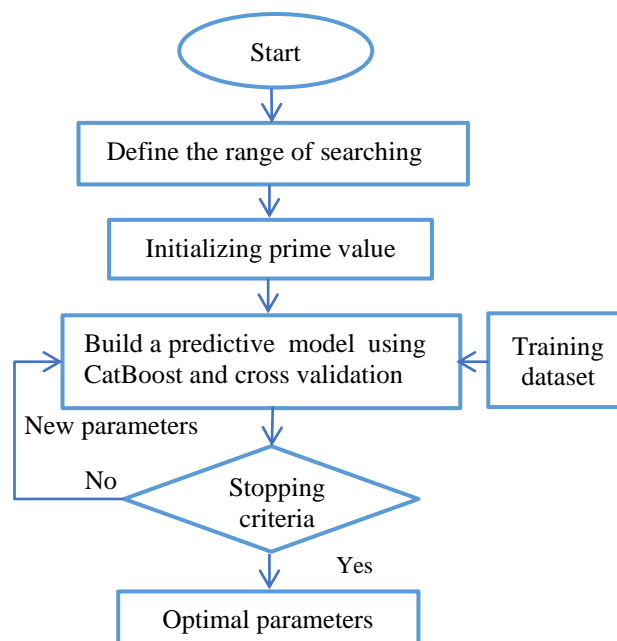
$$Precision = \frac{TP}{TP+FP} \quad (5)$$

### 3.5.Model Tuning

Setting parameters is generally considered one of the complex problems in determining the best parameters for the various machine learning algorithms[26]. Determining the appropriate parameters significantly impacts the prediction model performance and how to adjust those parameters. This section deals with the basic parameters of the CatBoost algorithm and the use of the random search and proposed methodology (Hyperopt) to find the optimal parameters to predict the heart attack for the given dataset.

#### Random Search Algorithm

Random search is a hyperparameter search algorithm that finds the best vector parameters for a machine learning model. Finding a sample randomly picked within the search range is how it calculates the best value for a given situation. The optimal global value or its approximate value can be found with high probability through random sampling for each possible group of per parameter values, then calculates the error and finally chooses the group in which the error is the least. A random search algorithm aims to find the optimal model parameters[27].



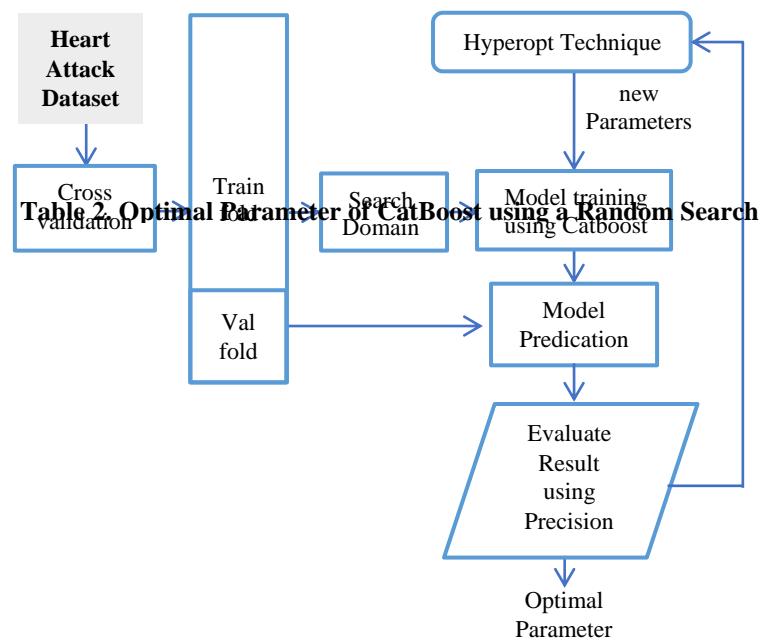
**Figure 1. Flowchart of random search for parameters optimization values of CatBoost**

In this paper, cross-validation is used for each set of parameters, obtaining the highest cross-validation score and the optimal parameter and using the random search algorithm to determine the particular parameters of (CatBoost). Figure (1) shows the flowchart of the random search algorithm. The results of each parameter are presented in Table (2).

Parameter	Range	Optimum value
learning_rate	[0.1, 0.3, 0.5, 0.7, 0.9]	0.5
n_estimators	[50, 100, 150, 200]	100
max_depth	[1, 2, 3, 4, 5]	3
Subsample	0.0-1.0	0.1

### Hperopt Optimization

it is especially important to employ the Hyperopt optimization technique to optimize parameters[27]. This approach finds the ideal parameters of the model quickly and effectively. The data set is divided into five-set using K-fold cross-validation, the declaration of the hyperparameters list name, and a range of values given to the cat boost algorithm based on domain search. And the (precision) measure of the minority class was used as the evaluation indicator once the optimal parameters were acquired. The optimal parameters obtained were then fed into the CatBoost model to increase the model prediction capabilities even more[27]. Table (3) represents the results of the Hyperopt, and the optimal values are created. The specific experimental process is shown in Figure (2).



**Figure 2. Diagram Hyperopt method for parameter optimization values of CatBoost**

**Table 3. Optimal values of hyperparameters using**

Parameter	Range	Optimum value
learning_rate	[0.1, 0.3, 0.5, 0.7, 0.9]	0.11
n_estimators	[50, 100, 150, 200]	101
max_depth	[1, 2, 3, 4, 5]	2
Subsample	0.0-1.0	0.8

Figure (3) shows the workflow of the proposed method. The model was built using the Python platform according to the following steps:

**Input:** Dataset of heart attack

**Output:** Score for accuracy, confusion matrix, and report on the classification of the forecasted value

**Process:**

Step 1: Import needed libraries from sklearn:(pandas,  
numpy)

Step 2: Import the classifier used

Step 3: Import split dataset function: (train\_test\_split)

Step 4: Import sklearn. Metrics.accuracy\_score function

Step 5: Load dataset of heart attack as CSV format using  
read\_csv() function

Step 6: Split the input and target filed

Step 7: Find optimal parameters using random Search

Step 8: Apply cross-validation using cross\_val\_score ()  
function

i. Randomly shuffle the data set

ii. Divide the data set into k sets.

iii. The groups are taken sequentially; one is a test group, and the rest is a training group.

iv. Each model is evaluated using test sets.

v. Repeat step 3\_4 for all groups.

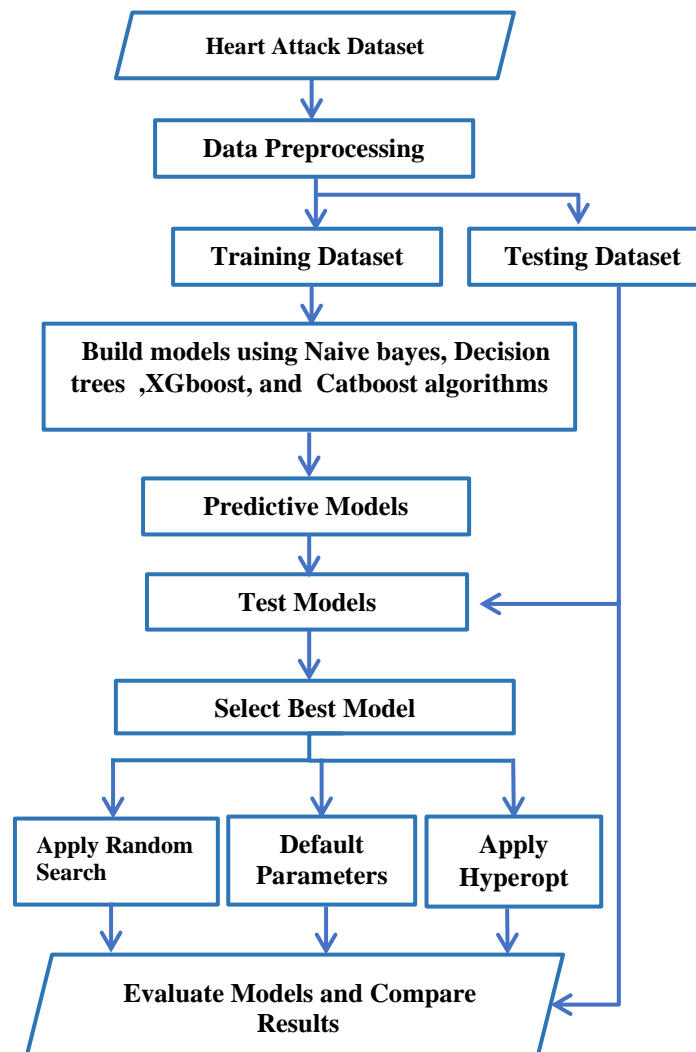
vi. The final result is calculated by calculating the average results of all k models.

Step 9: i) Calculate accuracy using accuracy\_score()  
function based on forecasted value and actual value

ii) Show classification report using classification  
report() function

iii) Find ROC curve using ROC() function

Step 10) Find optimal parameters using hyperopt and  
repeat step 8\_9

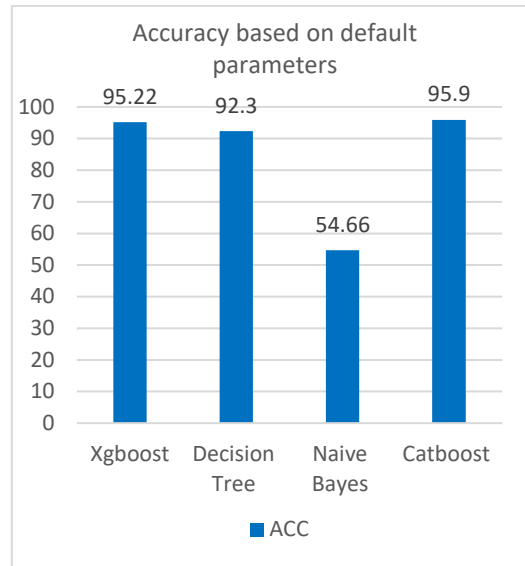


**Figure 3. Work Flow of Proposed Method**

#### **4. Comparing the Performance of Machine learning Models**

##### **4.1. Compare Models Using Accuracy Based on Default Parameters**

The accuracy scale represents a basic pillar in the performance measures for machine learning algorithms. Still, it expresses the model performance as a whole. Still, it does not show whether the model is biased to a certain category or how each target category is verified. figure (4) reviews the accuracy of the performance of the algorithm.

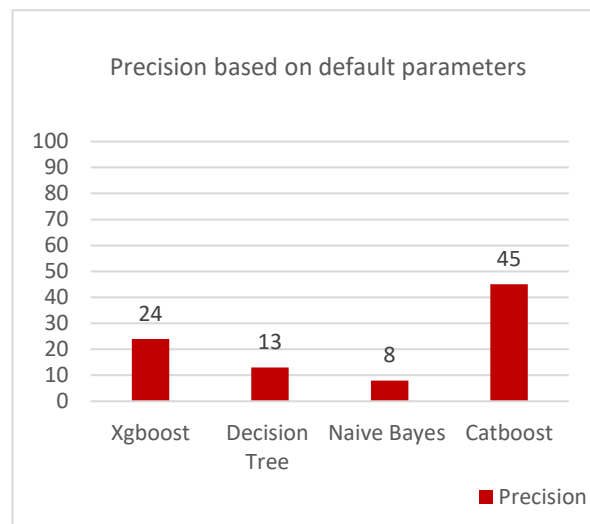


**Figure 4. Comparing the Accuracy of Models**

The accuracy scale does not represent the performance desired of the model, but it can be seen from Figure (4) that the CatBoost achieved the best performance with an accuracy of 95.9%. Still, this result does not express the target value of the system as a whole.

#### 4.2. Compare Models using Precision Based on Default Parameters

The precision scale is adopted in this paper as it expresses the true equilibrium value and shows the actual prediction of both categories. Figure (5) shows the performance of models for the minority category, which is the target value.



**Figure 5. Comparing the Precision of Models**

It can be seen from the table that the performance of the CatBoost algorithm was the best, with a precision of 45%, which indicates the model performance concerning the target value (the person with a heart attack). However, this method has been the best according to the precision scale.

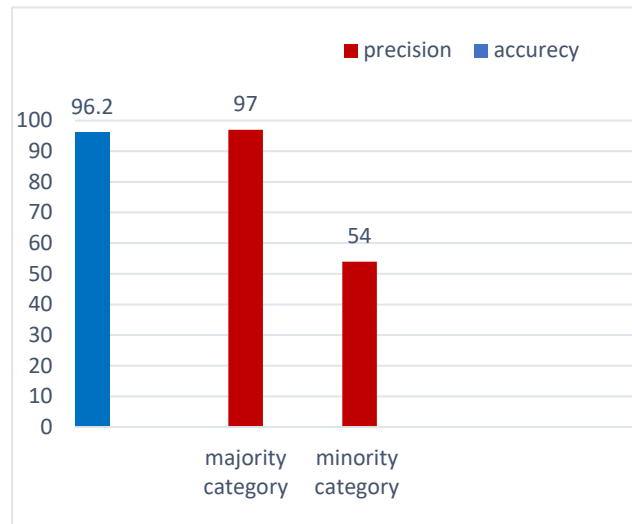
#### 5. Result of CatBoost model within Hyperparameter

After comparing the performance of models to solve the imbalance problem, as a subsequent step to improve the precision of the chosen model, it was proposed to use a Hyperopt method to find hyperparameters

and compare them with the random search and default parameters for finding hyperparameters. This section reviews the results of changing hyperparameters on the proposed model.

### 5.1. Catboost with Random Search

This paper used random search as a comparison tool to find hyperparameters. Parameters affecting the proposed algorithm were identified, and a range was defined for these parameters. After finding random search results and determining the parameter, it was applied to the CatBoost algorithm to know the performance of the model and the use of precision as an optimization function. Figure (6) displays the model performance using the random search parameters.

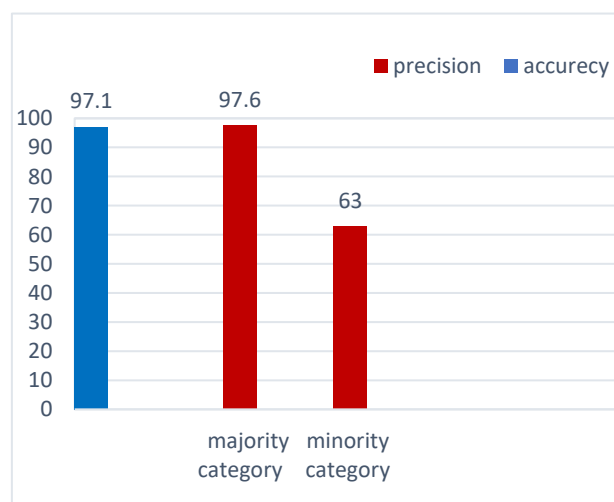


**Figure 6. Catboost with Random Search**

Using random search to find the parameters has increased the model performance compared to using default values of parameters. Figure (6) shows that the model accuracy has been 96.2. % The precision scale found that the minority category was 54%, and the majority category was 97%. This indicates that the random search has been better than default parameters in solving the problem of data imbalance and accuracy in general.

### 5.2. Catboost with Hyperopt Method

To find the hyperparameters, it was proposed to use the Hyperopt method and determine the precision of the minority class to determine the parameters of CatBoost. After finding the optimal values by Hyperopt Method, it was applied to the Catboost model and tested its performance.



**Figure 7. Catboost with Hyperopt Method**

Figure (7) shows the performance of the CatBoost model when applying the Hyperopt method to find hyperparameters. The proposed method achieved high performance on both scales, achieving an accuracy of 97.1% as a general performance for the model. At the same time, the precision for the minority class was 63.4% and 97.6% for the majority class. These results indicate the possibility of increasing the model's accuracy and balancing the results for unbalanced data sets.

## 6. CONCLUSIONS

In this paper, Data balance was determined for the target values, which gives a view to researchers in the field of artificial and medical intelligence. Four types of machine learning algorithms were used (DT and NB, XGboost, and CatBoost), and they were compared using different scales to determine the accuracy and the precision of the models for each category because the data set is unbalanced. The results showed that boosting is better for the accuracy of the results and achieves a better balance for this type of data. Finding the hyperparameters is not feasible because there are relationships between the parameters, and manually determining the best parameters is impossible. To find these parameters, the paper suggests using a Hyperopt method to find the optimal values, applying them and comparing them with a random search, and using the precision measure of the minority category to determine the desired value.

## References

1. Kumar, Yogesh & Koul, Apeksha & Singla, Ruchi & Ijaz, Muhammad Fazal. (2022). "Artificial intelligence in disease diagnosis: a systematic literature review, synthesizing framework and future research agenda". *Journal of Ambient Intelligence and Humanized Computing*. 1-28. 10.1007/s12652-021-03612-
2. Ahuja, Abhimanyu. (2019). "The impact of artificial intelligence in medicine on the future role of the physician". *Journal of PeerJ*. 7. e7702. 10.7717/peerj.7702.
3. Tarbell, John & Mahmoud, Marwa & Corti, Andrea & Cardoso, Luis & Caro, Colin. (2020). "The role of oxygen transport in atherosclerosis and vascular disease". *Journal of The Royal Society Interface*. 17. 20190732. 10.1098/rsif.2019.0732.
4. Hessel FP. (2021)" Overview of the socio-economic consequences of heart failure". *Cardiovasc Diagn Ther*. 11(1):254-262. doi: 10.21037/cdt-20-291. PMID: 33708497; PMCID: PMC7944217.
5. Khair, Hisham & Dasari, Naga. (2021). "Exploring Machine Learning Techniques for Coronary Heart Disease Prediction". *International Journal of Advanced Computer Science and Applications*. 12. 10.14569/IJACSA.2021.0120505.
6. Sharma, Chetan & Shambhu, Shankar & Das, Prasenjit & Jain, Dr. Shaily. (2021). "Features Contributing Towards Heart Disease Prediction Using Machine Learning".
7. Pabitra Kumar Bhunia & Pouami Mondal & Kankan Gitanguly & Arijit Debnath & Monalis D E & Pranati Rakshit .(2021)." Heart Disease Prediction using Machine Learning". *International Journal of engineering research technology*.2278-018.
8. Dutta, Aniruddha & Batabyal, Tamal & Basu, Meheli & Acton, Scott. (2020). "An Efficient Convolutional Neural Network for Coronary Heart Disease Prediction". *Journal of Expert Systems with Applications*. 159. 113408. 10.1016/j.eswa.2020.113408.
9. Rajliwall, Nitten & Davey, Rachel & Chetty, Girija. (2018). *Machine Learning Based Models for Cardiovascular Risk Prediction*. *Proceedings International Conference on Machine Learning and Data Engineering ,iCMLDE 2018*.142-148. 10.1109/iCMLDE.2018.00034.
10. David. H, Benjamin & Belcy, S.(2018). Heart disease prediction using data mining techniques. *ICTACT Journal On Soft Computing*. 10.21917/ijsc.2018.0253.
11. Walaa Adel Mahmoud a , Prof. Dr. Mohamed Aborizka a ,Prof. Dr. Fathy Ahmed Elsayed Amer2b.(2021). "Heart Disease Prediction Using Machine Learning and Data Mining Techniques: Application of Framingham Dataset". *Turkish Journal of Computer and Mathematics Education*. 4864- 4870.
12. Leevy, Joffrey & Khoshgoftaar, Taghi & Bauder, Richard & Seliya, Naeem. (2018). "A survey on addressing high-class imbalance in big data". *Journal of Big Data*. 5. 10.1186/s40537-018-0151-6.
13. Kumar, Pradeep & Bhatnagar, Roheet & Gaur, Kuntal & Bhatnagar, Anurag. (2021). "Classification of Imbalanced Data:Review of Methods and Applications". *IOP Conference Series: Materials Science and Engineering*. 1099. 012077. 10.1088/1757-899X/1099/1/012077.

14. Santiso, Sara & Casillas, Arantza & Pérez, Alicia. (2018). "The class imbalance problem detecting adverse drug reactions in electronic health records". *Health Informatics Journal*. 25. 146045821879947. 10.1177/1460458218799470.
15. <https://catalog.data.gov/dataset/national-health-and-nutrition-examination-survey-nhanes-national-cardiovascular-disease-su-00a88>
16. Kaviani, Pouria & Dhotre, Sunita. (2017). "Short Survey on Naive Bayes Algorithm". *International Journal of Advance Research in Computer Science and Management*. 04.
17. G, Parthasarathy. (2020). "Analysis Of Machine Learning Algorithm For Prediction Of Heart Disease". *International Journal of Advanced Research in Computer Science*. 11. 42-46. 10.26483/ijarcs.v11i3.6532.
18. Kumar, Dharmender & Priyanka, N.A.. (2020). "Decision tree classifier: a detailed survey". *International Journal of Information and Decision Sciences*. 12. 246. 10.1504/IJIDS.2020.10029122.
19. Reddy, V & Meghana, P & Reddy, N V Subba & Rao B, Ashwath. (2022). "Prediction on Cardiovascular disease using Decision tree and Naïve Bayes classifiers". *Journal of Physics: Conference Series*. 2161. 012015. 10.1088/1742-6596/2161/1/012015.
20. Sennan, Sankar & Potti, Anupama & Chandrika, G. & Ramasubbareddy, Somula. (2022). "Thyroid Disease Prediction Using XGBoost Algorithms". *Journal of Mobile Multimedia*. 10.13052/jmm1550-4646.18322.
21. Sigrist, Fabio. (2020). "Gradient and Newton boosting for classification and regression". *Journal of Expert Systems with Applications*. 167. 114080. 10.1016/j.eswa.2020.114080.
22. Liudmila Prokhorenkova & Gleb Gusev & Aleksandr Vorobev & Anna Veronika Dorogush & Andrey Gulin. (2018). "CatBoost: unbiased boosting with categorical features". 32nd Conference on Neural Information Processing Systems (NeurIPS 2018), Montréal, Canada.
23. Ibrahim, Abdullahi & Raheem, Ridwan & Muhammed, Muhammed & Abdulaziz, Rabiya & Ganiyu, Saheed. (2020). "Comparison of the CatBoost Classifier with other Machine Learning Methods". *International Journal of Advanced Computer Science and Applications*. 11. 11.
24. Dwivedi, Ashok. (2018). "Performance evaluation of different machine learning techniques for prediction of heart disease". *Journal of Neural Computing and Applications*. 10.1007/s00521-016-2604-1
25. Vujovic, Zeljko. (2021). "Classification Model Evaluation Metrics". *International Journal of Advanced Computer Science and Applications*. Volume 12. 599-606. 10.14569/IJACSA.2021.0120670.
26. Hossain, Md Riyad & Timmer, Douglas & Moya, Hiram. (2021). "Machine learning model optimization with hyper-parameter tuning approach". *Global Journal of Computer Science and Technology: D Neural & Artificial Intelligence*
27. Shekhar, Shashank & Bansode, Adesh & Salim, Asif. (2021). "A Comparative study of Hyper-Parameter Optimization Tools". 1-6. *IEEE Asia-Pacific Conference on Computer Science and Data Engineering (CSDE)*. 10.1109/CSDE53843.2021.9718485.