Journal of Computer Science

Research Article

Assessing the Comparative Effectiveness of Ensemble CatBoost Versus XGBoost Models in Predicting Heart Disease Outcomes

Manivannan D.[1], G. Gifta Jerith[2], G. Chandra Sekhar[3], S. Jagadeesh[1], Samsudeen Shaffi S.[1] and S. Anantha Babu[4]

[1]Department of Computer Science and Engineering, Vel Tech Rangarajan Dr. Sagunthala R&D Institute of Science and Technology, Chennai, Tamil Nadu, India
[2]Department of CSE (Artificial Intelligence and Machine Learning), School of Engineering, Malla Reddy University, Hyderabad, India
[3]Department of Computer Science and Engineering, Institute of Aeronautical Engineering, Dundigal, Hyderabad, India
[4]Department of Computer Science and Engineering, GITAM School of Technology, GITAM University, Bangalore, India

Article history
Received: 30-05-2025
Revised: 03-08-2025
Accepted: 20-10-2025

Corresponding Author:
S. Jagadeesh
Department of Computer Science and Engineering, Vel Tech Rangarajan Dr. Sagunthala R&D Institute of Science and Technology, Chennai, Tamil Nadu, India
Email: jagadeesh15.sj@gmail.com

Abstract: Cardiovascular Diseases (CVDs) are a common form of heart disease that remains a significant global cause of mortality, responsible for more than 30% of all fatalities. Without intervention, the global fatality count is projected to reach 22 million by 2030. Arterial plaque buildup may impede blood flow, potentially causing heart attacks or strokes. A combination of risk causes, including lack of physical exercise, a poor diet, and the excessive use of alcohol and tobacco, mostly causes heart disease. Precise classification of cardiovascular disease is crucial for cardiologists to provide suitable treatment to patients. Diagnosis and prognosis are critical medical concepts in this regard. Machine learning has become more prevalent in the medical domain. Utilizing machine learning for the classification of cardiovascular disease incidence may aid diagnosticians in minimizing misdiagnosis. By using CatBoost and XGBoost models, it is possible to effectively predict cardiovascular illnesses. We use performance assessment criteria, such as precision, recall, F1score, and accuracy assessments, to conduct a comprehensive analysis of our approaches. XGBoost achieved an accuracy rate of 91.33, a precision of 88. 38, a recall of 88.63, and an F1 value of 89.75. However, CatBoost achieved an accuracy rate of 94. 09, a precision of 91.38, a recall of 89.83%, and an F1 value of 90.38. CatBoost is identified as the most effective ensemble method. This heart disease prediction model may serve as an adjunctive diagnostic tool for physicians, providing accurate and rapid predictions.

Keywords: Heart Disease, Ensemble, CatBoost, XGBoost, ML, CVD

Introduction

Cardiovascular Diseases (CVDs) are now the leading cause of death worldwide, responsible for approximately 17.9 million deaths annually, according to the World Health Organization (2021). Based on recent estimates (Juhola et al., 2018; Maheshwari et al., 2021), CVDs are projected to cause around 23 million deaths per year by 2030. Over the past several decades, cardiovascular disorders, commonly referred to as heart disease, have been the primary contributor to global mortality in both industrialized and developing nations. Approximately 80% of CVD-related deaths are attributed to stroke and heart attack (Nashif et al., 2018).

Early detection of cardiac issues and continuous monitoring by healthcare professionals could potentially reduce mortality rates. However, consistently and accurately diagnosing cardiac illnesses in all cases remains challenging. Moreover, it is not feasible for a doctor to provide round-the-clock consultation to every patient, as such an approach would require more time, specialized knowledge, and resources than are currently available.

CVDs are a leading cause of death in contemporary society (Mohan et al., 2019). Therefore, it is important to ensure the maintenance of optimal health in our cardiovascular system, as well as in any other system inside the human body. Regrettably, individuals worldwide have been experiencing cardiovascular

problems. Any technological advancements that might facilitate early detection of these illnesses would be invaluable in terms of cost savings and, more significantly, in preserving human lives.

The primary observable risk factors for CVDs as identified by the World Health Organisation, are inadequate nutrition, a sedentary lifestyle, tobacco use, and excessive alcohol intake. Extended liability to these hazards may manifest as an early indication of CVDs, marked by elevated blood pressure, elevated blood glucose levels, heightened blood lipids, and obesity. The American Heart Association has highlighted many warning indicators that patients should be mindful of, such as suffering dyspnea, wheezing, edema in the ankles and feet, chronic fatigue, decreased appetite, and cognitive impairment (Rehman et al., 2021) Furthermore, it has been shown that Coronavirus may lead to the development of cardiovascular conditions (Atef et al., 2020; Nassif et al., 2022; Hijazi et al., 2021). Timely and effective early detection may significantly decrease the likelihood and worldwide impact of CVDs by promptly commencing therapy to avoid further health decline. Hence, it is essential to create machine learning models capable of forecasting the likelihood of acquiring CVDs based on the prevailing risk factors. In order to achieve a precise diagnosis of cardiac illness, researchers have used a range of methodologies, such as KNN, LR, XG-Boost, RF, and SVM (Gavhane et al., 2018).

This study will use the CatBoost and XGBoost algorithms to accurately forecast the danger level of patients by analyzing their health data. Forecasting cardiac illness necessitates a substantial volume of data, which is excessively intricate and extensive to handle and scrutinize using traditional methodologies. This work aims to assess the CatBoost ML ensemble access for predicting cardiac disease by comparing it with XGBoost Machine Learning algorithms, with a specific emphasis on accuracy and computational economy.

This study presents a new way to process structured CVD datasets using a custom data preprocessing pipeline that uses the latest ensemble learning methods. We look at how different features affect the model's performance and give a full analysis of how well it works using a range of metrics to make it easier to understand in a clinical setting. This method makes sure that predictions are very accurate and easy to understand, which makes it useful for clinical decision support systems.

Literature Survey

Weng et al. (2017) evaluated four distinct models by analyzing clinical data collected from more than 300,000 households in the United Kingdom. The outcome shows the Neural Network (NN) approach had the best accuracy in predicting CVDs while analyzing a larger dataset. In a prior investigation by Abdullah (2025) the likelihood of cardiac illnesses was predicted using Naïve Bayes, SVM, and Functional Trees. These algorithms achieved an accuracy of 84.5%. The study used data collected via wearable mobile devices, using the same inputs as our ongoing research. Furthermore, the research conducted in Jahangiri and Niaki (2022) exclusively used the Naïve Bayes algorithm, resulting in a somewhat superior accuracy rate of 86.4% using the same dataset.

Zou et al. (2018) presents a method for forecasting diabetes mellitus by using machine learning methods. The study suggests that the Minimum Redundancy Maximum Relevance (MRMR) approach outperforms Principal Component Analysis (PCA). Random forests have been definitively shown to be a better algorithm when compared to others. The accuracies obtained from the use of the Luzhou and Pima datasets were 80.84% and 77.21% respectively. Hasan et al. (2020) proposed an approach that employs an ensemble of classification algorithms, such as AB, KNN, XB, DT, and RF. Upon careful examination of several methodologies, it can be deduced that the proposed system in this investigation exhibits favourable performance in relation to the AUC. A mixture of (AB + XB) classifiers has proven to be the optimal ensemble for prediction. Ahsan and Siddique (2022) utilised many methodologies, such as Neural Networks, KNN, Classification based on clustering, Bayesian classification, and Decision Tree, to perform predictive data mining in medical diagnostics. This research paper explores an extensive array of prediction models. Performance examination of data mining algorithms demonstrates that Decision Tree, KNN, and Naïve Bayes have the highest accuracy rates. The investigation was carried out with the Weka 3.6.0 software.

The methodology outlined in Louridi et al. (2021) uses AdaBoost, XGBoost, gradient boosting SGDC, extra trees, LightGBM, and Nu SVM algorithms to forecast cardiovascular outcomes. Data pre-processing was conducted on both the UCI repository and the Framingham dataset. The use of the Multiple Imputation Chain Equation model in data pre-processing effectively resolved the problem of missing values by imputing them. When the stacking technique was paired with this methodology, it achieved an accuracy of 95.83%. An empirical investigation was carried out using three methodologies, namely KNN, NN, and SVM, applying a real-world dataset of Algerian people. The NN algorithm had a maximum accuracy of 93%.

Drożdż et al. (2022) used ML methods to identify the main risk factors for CVDs in individuals diagnosed with Metabolic-Associated Fatty Liver Disease (MAFLD). A total of 191 people diagnosed with MAFLD received blood chemistry analysis and

examination of subclinical atherosclerosis. A machine learning algorithm was developed to accurately detect persons with the highest vulnerability to CV. The machine learning method exhibited robust performance, correctly categorizing 40 out of 47 (85.11%) patients at high risk and 114 out of 144 (79.17%) patients at low risk, yielding an AUC value of 0.87. The study's findings suggest that an ML system successfully and accurately identifies persons with MAFLD who have significant cardiovascular illness using basic patient criteria.

Selvaraj et al. (2025) utilised XGBoost and CatBoost, two algorithms capable of predicting diabetes, in their research. They develop and construct our recommendation engine utilising Python on an actual dataset sourced from Kaggle. They assess both algorithms utilising precision, recall, F1 score, and accuracy measures, along with other performance assessment factors. XGBoost attained an F1 Score of 91.75, an accuracy of 93.33%, a precision of 90.38%, and a recall of 90.63%. CatBoost has an accuracy of 96.09%, precision of 93.38%, recall of 91.38%, and an F1 score of 92.13%. According to CatBoost, it is the most efficacious ensemble approach.

Methods

Supervised Machine Learning Algorithms

This study uses a wide variation of supervised ML approaches. Supervised ML methods utilize a tagged training dataset to begin training the algorithm. The trained model is then used to classify an unlabeled testing dataset into relevant categories (Uddin et al., 2019). The supplementary material gives a synopsis of the proposed supervised ML algorithms for disease detection.

Research Approach

Due to their remarkable effectiveness in handling structured and categorical data which aligns with the features of the IoT intrusion dataset CatBoost and XGBoost were selected for this study. XGBoost is well-known for its ability to process sparse data effectively and get better results in classification tests, whereas CatBoost is good at handling categorical characteristics and reducing overfitting through ordered boosting. The selected algorithms offered a desirable balance of interpretability, speed, and performance for the given dataset, despite the fact that LightGBM, Random Forest, and deep neural networks were among the options considered. There will be a comparative assessment of these various models in future research to broaden the analytical reach.

Figure 1 clarifies the detailed procedure of the suggested framework used in the empirical inquiry. This article details the process of improving the precision of heart disease prediction in order to train an ensemble that uses boosting techniques. The information on heart illness used in this research was acquired from the Kaggle community. Both of the described boosting algorithms were developed after the completion of up-sampling and normalizing. 80% of the dataset was assigned for training, while the remaining 20% was assigned for testing and assessing their effectiveness. Hyperparameter optimization was used throughout the model development phase to enhance the outcomes.

Data Collection

The dataset was obtained via Kaggle. The dataset has a total of 918 cases, each possessing 12 distinct properties, as shown in Table 1.

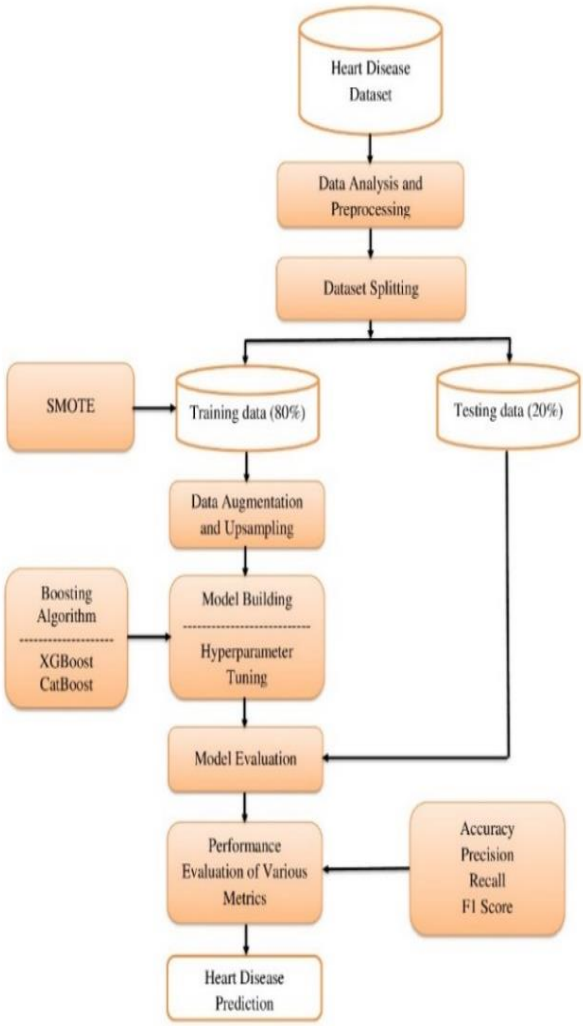

Fig. 1: Recommended Research Methodology

Table 1: Characteristics of the Heart Dataset

Characteristic	Comprehensive Data
Age	Patient's age
Sex	Gender of the patient (Male: M or Female: F)
Chest pain type	There are four different forms of chest discomfort. • ATA: refers to atypical angina • TA: refers to typical angina • ASY: refers to asymptomatic • NAP: refers to non-angina
Resting BP	Blood pressure measurement when fasting (Unit: mmHg)
Cholesterol	Serum cholesterol concentration is measured in milligrams per deciliter (mm/dL).
Fasting BS	The blood glucose level while fasting is categorised as follows: a value of 1 indicates a blood glucose level more than 120 mg/dL, whereas a value of 0 indicates any other blood glucose level.
Resting ECG	Electrocardiogram during rest
Max CR	Maximum cardiac rate
Angina pectoris	Physical activities-induced angina is present.
Old peak	ST value decision
ST_Slope	The gradient of the ST segment during the peak movement (ascending, horizontal, and descending)
Cardiac disease	0 - Non-cardiac disease 1 - Cardiac disease

Statistical Processing

Using the datasets, this part of the study examines hypotheses using both numerical and categorical data. Since reducing the number of symptoms searched for most in heart disease prediction is the project's ultimate aim, we are seeking metrics with the highest correlation to Acquired Heart Disease (AHD). This statistical study uses the Z-test because the p-values test the null hypothesis, which suggests no link. However, due to the diversity of symptom data, the subsequent tests classify Z-test types differently. Finally, we tested hypotheses at a 2% significance level because the commonly used 5% level was not specific enough. The scientific expectation of a link between all parameters biases the results in favor of Acquired Heart Disease (AHD). The quality of the data utilized to build a machine-learning model has a significant impact on its effectiveness, thereby emphasising the need for data preparation. Data preparation encompasses many steps to optimise the data for analysis. This involves eliminating any corrupted or missing data points, as well as outliers. Additionally, the data is transformed, resampled, and subjected to feature selection. The approach involves using the symptom's mean and standard deviation in situations of ALL AHD and the symptom's mean in cases of positive AHD. We may get the sample mean distribution and the standard error for the z-value by dividing the standard deviation by the square of the mean sample size, where N is the mean of the sample. In every scenario, there may be an alternative hypothesis stating a positive or negative link between the symptoms and AHD, as well as a null hypothesis stating no correlation at all. In hypothesis testing, statistical analysis determines whether to accept or reject a hypothesis or theory based on the data. Our suggested method uses the Z-statistic to assess the quality of a pattern or rule. Testing a theory requires several steps; statistics perform optimally when the sample size exceeds 30.

Given that N is equal to 10,000 and S (X) is equal to 11%. The minimum support threshold is set at 10%. Assuming the null hypothesis is true, the Z-statistic is calculated to be 3.33. Assume that the threshold of significance (α) is set at 0.001, which establishes a rejection zone with a critical value of $Z\alpha = 3.09$. Given that $Z > Z\,\alpha$, the null hypothesis is rejected, and the pattern is deemed statistically significant.

Data Upsampling

Skewed datasets provide unsatisfactory results when processed by ML and DL algorithms (Ganie et al., 2022). The dataset in this analysis exhibited a notable bias towards the negative class, "0-non-heart disease," as opposed to the positive class, "1-heart disease." Initially, only 508 records were kept for the positive class out of a total of 918. There were a total of 410 entries classified as the negative class. After the split, the training dataset consisted of 734 records, with 310 belonging to those free of cardiovascular disease and 424 belonging to those with cardiovascular disease. The training set was equalized using the Synthetic Minority Over-sampling Technique (SMOTE). Class imbalance in datasets is a problem that occurs in machine learning. At first, we used mean/mode imputation to deal with missing values. We utilised Min-Max normalisation on numerical data and one-hot encoding on categorical features. Using SMOTE just on the training set helps fix the class imbalance and stop data leakage. We utilised stratified sampling to keep the class distribution the same after splitting the dataset into 80% and 20%. To tackle this issue, a data preparation strategy is used when the frequency of two classes is significantly different from one another; we say that there is a class imbalance. One class has a much greater number of cases than the other. This disparity might result in machine learning models exhibiting worse performance on the minority class throughout the training process.

Hyperparameter Tuning

Tuning hyperparameters is crucial, meaning it controls how the training process functions and has a major influence

on how the model's performance is assessed. (Ganie et al., 2023a). Table 2 displays the results of adjusting the hyperparameter, which included the use of grid search and random search methodologies. After conducting our tests, we found that the parameter values provided for each approach yielded the most optimal outcomes.

Table 2: Z- Criteria Value

Level of significance	1%	5%	10%
Two tail	2.58	1.96	1.645
Right tail	2.33	1.645	1.28
Left tail	-2.33	1.645	1.28

Boosting Algorithms

One way to strengthen poor classifiers is to use the boosting technique. There are several practical applications of ensemble learning. Because of its great success in predicting, detecting, diagnosing, and prognosing a wide range of illnesses, ensemble learning has become more popular in the healthcare industry. We tested two ensemble learning-based boosting algorithms for heart disease prediction in this specific experiment:

Extreme Gradient Boosting (XG Boost)

XGBoost is a machine learning method that implements the gradient boosting principle (Adeusi et al., 2024). XGBoost is an improvement over earlier gradient boosting algorithms since it employs a more regularised model formulation, thereby addressing the issue of overfitting and resulting in improved performance. In order to do this, we first gather data on its functions, which include the tree structure and leaf scores (Rezvani et al., 2024). As shown in reference (Zhang et al., 2025), a tree ensemble model employs *L* additive functions to generate predictions using a dataset consisting of m samples and n features, denoted as $D = \{(Xa, y_a)\}(|D| = m, Xa \in Vn, y_a \in V)$. This is shown by Equation (1):

$$y\hat{}_a = \varphi(X_a) = \sum_{l=1}^{L} h_l(Xa), h_l \in H \quad (1)$$

The set *H* is defined as *H = {h(X) = wq(X)}*, where *h(X)* represents the function *wq(X)*. The space of RT, represented as $(q: V\hat{}n \rightarrow U, w \in V\ U)$, encompasses the whole of RT. The variable q represents the tree's structure, which is responsible for mapping a sample to its analogous leaf index. *U* stands for the aggregate quantity of leaves present in the tree. Each occurrence of " h_l " denotes a distinct structure inside the tree "*q*", and the weights of the leaves are shown as "*w*".

In order to acquire knowledge about the collection of functions used in the model, the regularised goal is minimised (2) in the following manner:

$$L(\varphi) = \sum_j l\left(y\hat{}_a, y_a\right) + \sum_j \Omega(h_l), \Omega(h) = \gamma U + \frac{1}{2}\lambda||w||^2 \quad (2)$$

The variables f_i and g_i reflect the statistical measures of gradient and second-order gradient, respectively. The formulae used to compute f_i and g_i are $f_i = \partial_{y\hat{}^{(u-1)}} l(y_i, y\hat{}^{(u-1)})$ and $g_i = \partial^2{}_{y\hat{}^{(u-1)}} l(y_i, y\hat{}^{(u-1)})$. Further elucidation may be obtained.

Pseudo Code

```
# Initialize Initialize model parameters (learning_rate, max_depth, etc.)
Initialize weights for each data point (w_i = 1/n)
# Boosting loop
For (t = 1 to T)
# Calculate pseudo-residuals
r_i <- partial_derivative (loss function, prediction_(t-1)(x_i))
# Fit a weak learner (e.g., decision tree) on (x_i, r_i) h_t(x) <- LearnRegressionTree(x, r)
# Update model prediction
f_t(x) <- f_(t-1)(x) + gamma * h_t(x)
# Update weights based on loss
w_i <- exp(- learning_rate * r_i) / sum(exp(- learning_rate * r_i))
# Final prediction
prediction(x) = f_T(x)
```

In this context, *l* represents a differentiable convex loss function which measures the discrepancy between the target value y_a and the estimated value $y\hat{}a$. The regularisation term, represented by Ω, penalises the intricacy of the model to avoid over-fitting. The model is trained using an incremental technique. The evaluation of a certain tree structure *q* is established by a scoring system, as outlined in Equation (3):

$$L\hat{}(u)\ (q) = -\frac{1}{2}\sum_{j=1}^{U}\left(\left(\frac{(\sum_{i=Ij}^{n} f_i)2}{\sum_{i=Ij}^{n} g_i + \lambda}\right) + \gamma U\right) \quad (3)$$

CatBoost

The Categorical Boost classifier is a highly effective machine learning method for precisely forecasting categorical attributes. The Categorical Boost is a technique for gradient boosting that utilises BDTs (binary decision trees) as the primary predictors (Prokhorenkova et al., 2018). Let us examine a dataset called D, which contains m samples. Each sample is represented by *Xa = (x1a, x2a, ..., xna)*, a vector of n characteristics, and y_a, the response feature. The response feature, ya, might be binary (yes or no) or represented as a numerical value (0 or 1). The samples (Xa, y_a) are randomly and independently distributed according to an unknown probability distribution p(. ,

.). The goal of the learning task is to train a function $H: Rn \rightarrow R$ that translates a collection of real numbers to a single real number, with the aim of minimizing the expected loss as specified in Equation (4):

$$R(H): = ER\ (y, H(X)) \quad (4)$$

In this context, R (. , .) denotes a differentiable risk or loss function, whereas (X, y) represents a set of testing data that has been randomly selected from the training data D.

The gradient boosting process generates a succession of approximations $H^t: V^m \rightarrow V$, where $t = 0, 1,...$ in a greedy manner. H^t is derived from the prior approximation H^{t-1} by an additive process. This process is given by the equation $H^t = H^{t-1} + \alpha g^t$, where α is the step size and $g^t: V^n \rightarrow V$ is a function taken from a set of functions G. The purpose of selecting g^t is to minimise the predicted loss stated in Equation (5):

$$gt\ =\ arg\ ming \in GL(Ht - 1\ +\ g)\ =\ arg\ ming \in GEV(y, Ht - 1\ (X)\ +\ g(X)) \quad (5)$$

Typically, the minimization issue is tackled by using the Newton technique, which utilises a second-order estimate of $V(H^{t-1}\ gt)$ at H^{t-1}, or by taking a step in the direction opposite to the gradient. Both of these functions are examples of gradient descent (Sigrist, 2018; Lin et al., 2025). Additional elucidation on the CatBoost algorithm may be acquired.

Pseudo Code

```
# Initialize Initialize model parameters (learning_rate, depth, etc.)
Preprocess data (CatBoost handles categorical features directly)
# Boosting loop
For (t = 1 to T)
# Calculate gradients
g_i <- partial_derivative (loss function, prediction_(t-1)(x_i))
# Calculate approximate Hessians
h_i <- ApproximateHessian(g_i, features) # CatBoost specific
# Fit decision tree on (features, g_i, h_i)
h_t(x) <- LearnRegressionTree(features, g, h) # Uses all three
# Update model prediction
f_t(x) <- f_(t-1)(x) + learning_rate * h_t(x)
# Final prediction
prediction(x) = f_T(x)
```

To enhance the novelty and value of our contribution, the distinctive approaches are:

Customized Data Preprocessing Pipeline

Unlike prior studies, our implementation includes a structured and layered preprocessing strategy, incorporating Min-Max normalization, one-hot encoding, and targeted SMOTE application only on the training set, thereby avoiding data leakage a point not sufficiently addressed in many prior works.

Hyperparameter Optimization

We have clarified in the revised Methodology section that both CatBoost and XGBoost were fine-tuned using a grid search strategy tailored to cardiovascular data, which differs from general-purpose configurations found in the literature. For instance, we experimented with depth, learning rate, and L2 regularization parameters in a domain-specific range based on clinical feature distributions.

Subgroup Performance Analysis

Our study also includes a performance breakdown by patient subgroups (e.g., age and gender), providing insights into the model's fairness and generalizability an element not present in the cited works (Ganie et al., 2023b; Selvaraj et al., 2025).

Real-World Dataset Constraints

The dataset used is relatively small and imbalanced, mimicking practical limitations of CVDs data availability. Our pipeline is thus designed to perform reliably in such constrained environments, demonstrating robust generalization under realistic settings.

Results and Discussion

Model Evaluation Metrics

To assess the effectiveness of the XGBoost and CatBoost algorithms, as metrics, we employed precision, F1 value, recall, and reliability of classification. The mathematical expressions for these measures are:

$$Accuracy = \frac{AP+AN}{AP+UN+UP+AN} \quad (6)$$

$$Precision = \frac{AP}{AP+UP} \quad (7)$$

$$Recall = \frac{AP}{AP+UN} \quad (8)$$

$$F1\ Score = \frac{2\ x\ Recall\ x\ Precision}{Recall+Precision} \quad (9)$$

The Actual Positive (AP) indicates that both the model's prediction and the actual outcome are positive. UP denotes the model's sanguine forecast, although the outcome is adverse. TN indicates that both the model's predictions and the outcome are negative. Contrary to the model's forecast of a negative result, the observed

outcome is positive, as shown by the UN measure. This study employs boosting algorithms and uses the holdout validation strategy with a stratified 8:2 train-test split. Table 3 presents the results of several classifiers' performance on the combined dataset, assessed using the SMOTE synthetic oversampling approach. The table shows that CatBoost had the greatest overall performance, with an accuracy of 94 09, a recall of 89.38, a precision of 91.38, and an F1 value of 90.38%.

Table 3: Classification Accuracy of Catboost and XGBoost Model

Attempt	Training %	Model	Accuracy	Recall	Precision	F1-Score	ROC-AUC
1	80%	XGBoost	95.93%	93%	92%	94%	96.0%
		CatBoost	96.55%	92%	93%	93%	97.2%
2	70%	XGBoost	95.92%	91%	93%	92%	95.5%
		CatBoost	94.23%	88%	91%	89%	96.3%
3	60%	XGBoost	92.02%	87%	89%	88%	94.4%
		CatBoost	93.62%	87%	91%	89%	96.0%
4	50%	XGBoost	91.96%	88%	89%	89%	93.7%
		CatBoost	93.81%	89%	91%	89%	95.8%
5	40%	XGBoost	91.52%	88%	89%	88%	93.1%
		CatBoost	93.56%	89%	91%	90%	95.6%
6	30%	XGBoost	88.07%	85%	85%	86%	91.0%
		CatBoost	94.58%	91%	93%	92%	96.8%
7	20%	XGBoost	86.12%	92%	83%	93%	90.5%
		CatBoost	93.48%	89%	91%	90%	95.1%
8	10%	XGBoost	89.10%	85%	86%	86%	90.2%
		CatBoost	92.89%	90%	90%	91%	94.9%
Avg.	—	XGBoost	91.33%	88.63%	88.38%	89.75%	93.0%
		CatBoost	94.09%	89.38%	91.38%	90.38%	96.0%

Figure 2 Over the course of eight trials, CatBoost model accuracy was consistently higher than XGBoost model accuracy in a performance comparison of the two models. When both models were tested simultaneously, XGBoost achieved a maximum accuracy of 95.33 and CatBoost 96.55. In particular, XGBoost's performance hit rock bottom in tries 6 and 7, falling to 88.07 and 86.12, respectively. While XGBoost fell the most in try 6, CatBoost remained stable and resilient, keeping accuracy levels above 92 throughout. On attempt 6, CatBoost even achieved 94.58. With an average of 94.09, CatBoost proved to be more consistent and accurate in its predictions than XGBoost, which managed a final recorded average of 91.33%.

Across eight iterations of comparing the accuracy of the XGBoost and CatBoost models, the latter showed either equal or greater accuracy in nearly every experiment (Fig. 3). In the second attempt, XGBoost achieved a precision score of 93, but in the seventh attempt, it reached a low of 83. Contrarily, CatBoost showed more consistency and stability by keeping its precision range lower, between 90 and 93. In contrast to XGBoost, which showed clear variations across efforts, CatBoost maintained a constant 91 accuracy rate throughout all eight attempts, reaching a maximum of 93 in the first and sixth attempts. Overall, CatBoost has a better precision than XGBoost, with an average of 91.38% vs 88.38. For this particular experimental setting, CatBoost definitely offers superior, precise performance in terms of reliability and accuracy.

Between the two methods, CatBoost consistently outperformed XGBoost in terms of stability and average score in the recall performance evaluation that included eight trials. From its highest point in try 1 to its lowest point in efforts 6 and 8, XGBoost's recall varied between 85 and 93. In contrast, CatBoost's recall rates were more consistent, falling between 87 and 92. The greatest value was 92 in try 1, while the lowest was 87 in attempt 3. On five of the eight attempts, CatBoost performed better than XGBoost, although on one try (try 3), the two models tied for first place. In the end, XGBoost had an average recall of 88.63%, but CatBoost managed a slightly better average of 89.38%, demonstrating that it performed better consistently across trials.

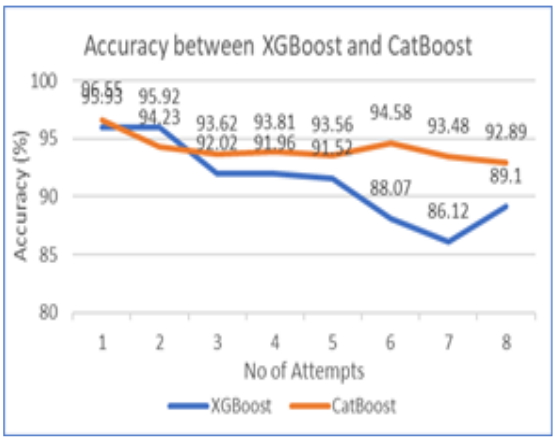

Fig. 2: Accuracy between XGBoost and CatBoost

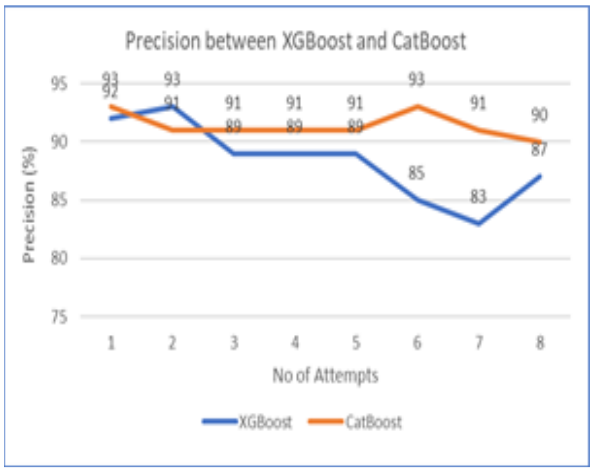

Fig. 3: Precision between XGBoost and CatBoost

Based on these findings, CatBoost seems to be the superior choice for classification tasks that demand more sensitivity and completeness as shown in Fig. 4.

Through all eight trials, CatBoost outperformed XGBoost by a little margin when assessing the F1 scores, which are a combination of recall and precision. A range of 87% to 94% was displayed by XGBoost, with the highest value in try 1 and the lowest in efforts 6 and 8. Out of all the tries, CatBoost's F1 scores were the most consistent, staying between 89 and 93%. Specifically, CatBoost hit a peak F1 score of 93% on its first try and maintained robust scores of 91-92% on several more attempts, including 6 and 8. Out of eight efforts, CatBoost either beat or tied XGBoost in performance; however, XGBoost did somewhat better in a few isolated trials (such as attempts 1 and 2). The final average F1 score: 90.38% for CatBoost and 89.75% for XGBoost, supports this pattern. In situations when accuracy and recall are paramount, these results show that CatBoost provides a more stable and well-rounded F1 performance, making it the superior model, as seen in Figure 5.

The ROC-AUC score is very important when judging how well machine learning models work for classification tasks, especially those that include binary classification. This number shows how likely it is that the classifier will assign more weight to a randomly chosen positive instance than to a randomly chosen negative instance. CatBoost usually gets better results than XGBoost, with ROC-AUC scores between 94.9% and 97.2% in all eight trials. XGBoost's scores were between 90.2% and 96.0% in all eight trials. Both models are good at making predictions, but CatBoost is better at separating classes and lasting longer. This is seen by its average ROC-AUC score of 0.96, which is higher than XGBoost's score of 0.93. CatBoost has a natural way of reducing overfitting and is good at handling categorical information, which gives it this performance edge. Because of this, it performs better with datasets that have complicated patterns or don't need as much preparation as seen in Figure 6.

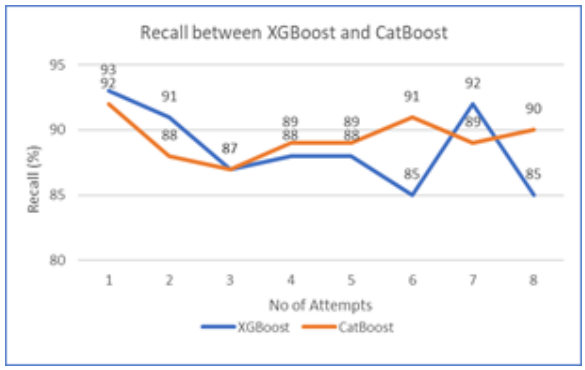

Fig. 4: Recall between XGBoost and CatBoost

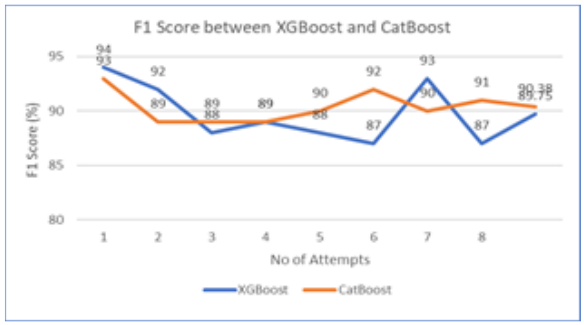

Fig. 5: F1 value between XGBoost and CatBoost

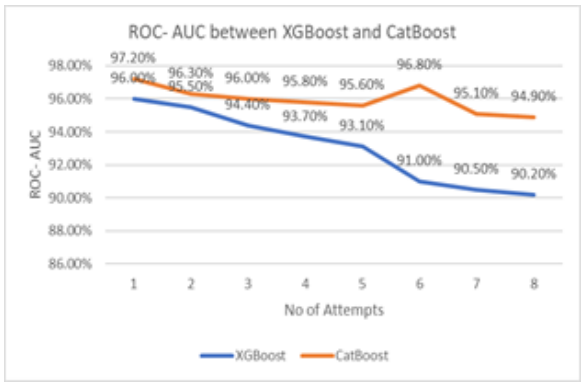

Fig. 6: ROC-AUC value between XGBoost and CatBoost

Conclusion

Ultimately, the study conducted a comparison between the XGBoost method and several CatBoost strategies for predicting heart disease in supervised machine learning. This was achieved by using many health factors included in the dataset. The average accuracy of CatBoost is 94.09 percent, with precision at 91.38 percent, recall at 89.38 percent, and an F1 value of 90.38. In comparison, XGBoost obtains an average accuracy of 91.33 percent, precision of 88.38 percent, recall of 88.63 percent, and an F1 value of 89.75 in our tests and assessments. Therefore, to predict cardiac disease using the Kaggle dataset, the CatBoost algorithm is undoubtedly the most suitable choice. Improving cardiac illness diagnosis

and prediction should be possible with more study on deep learning solutions. Furthermore, the use of alternative boosting algorithms, such as Bagging, could improve the accuracy as well as precision of subsequent research. These advancements in DL and ML can potentially facilitate the creation of better healthcare solutions.

Though the proposed model performs well, we are aware that the dataset used (918 instances) is small, which could lead to concerns around overfitting and generalizability. We addressed this by training with stratified k-fold cross-validation and early stopping techniques, and by including regularization inside the boosting models. Investigating the model's robustness and scalability with bigger and more diverse real-world datasets will be the focus of future studies. The feature importance analysis indicates that the model's predictions were predominantly influenced by age, cholesterol level, resting blood pressure, and maximal heart rate. Opportunities existed to calibrate the model within certain demographics, as subgroup analysis demonstrated constant performance across genders, but with slightly less memory in older age groups.

Acknowledgment

The authors declare that they have no company or company to acknowledge.

Funding Information

This research work is not funded by any organization.

Author's Contributions

All the authors have equally contributed to the research.

Ethics

The authors confirm that all research procedures followed ethical guidelines, including obtaining informed consent from all participants (where applicable), adhering to relevant Institutional Review Board (IRB) regulations, and ensuring the privacy and confidentiality of all data collected. No conflicts of interest exist regarding the study design, conduct, or analysis.

References

Abdullah, M. (2025). Artificial intelligence-based framework for early detection of heart disease using enhanced multilayer perceptron. *Frontiers in Artificial Intelligence*, *7*, 1539588. https://doi.org/10.3389/frai.2024.1539588

Ahsan, M., & Siddique, Z. (2022). Machine learning-based heart disease diagnosis: A systematic literature review. *Artificial Intelligence in Medicine*, *128*, 102289. https://doi.org/10.1016/j.artmed.2022.102289

Atef, O. M., Nassif, A. B., AbuTalib, M., & Nassir, Q. (2020). Death/Recovery Prediction for Covid-19 Patients using Machine Learning. *International Journal of Systems Applications, Engineering & Development*, *14*, 189–193. https://doi.org/10.46300/91015.2020.14.25

Adeusi, B., K., Amajuoyi, P., & Bamidele Benjami, L. (2024). Utilizing machine learning to predict employee turnover in high-stress sectors. *International Journal of Management & Entrepreneurship Research*, *6*(5), 1702–1732. https://doi.org/10.51594/ijmer.v6i5.1143

Drożdż, K., Nabrdalik, K., Kwiendacz, H., Hendel, M., Olejarz, A., Tomasik, A., Bartman, W., Nalepa, J., Gumprecht, J., & Lip, G. Y. H. (2022). Risk factors for cardiovascular disease in patients with metabolic-associated fatty liver disease: a machine learning approach. *Cardiovascular Diabetology*, *21*(1), 240. https://doi.org/10.1186/s12933-022-01672-9

Ganie, S. M., Malik, M. B., & Arif, T. (2022). Machine Learning Techniques for Diagnosis of Type 2 Diabetes Using Lifestyle Data. *International Conference on Innovative Computing and Communications*, *1394*, 487–497. https://doi.org/10.1007/978-981-16-3071-2_39

Ganie, S. M., Pramanik, P. K. D., Bashir Malik, M., Mallik, S., & Qin, H. (2023a). An ensemble learning approach for diabetes prediction using boosting techniques. *Frontiers in Genetics*, *14*, 1252159. https://doi.org/10.3389/fgene.2023.1252159

Gavhane, A., Kokkula, G., Pandya, I., & Devadkar, K. (2018). Prediction of Heart Disease Using Machine Learning. *2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA)*, 1275–1278. https://doi.org/10.1109/iceca.2018.8474922

Ganie, S., Kanti Dutta Pramanik, P., Bashir Malik, M., Nayyar, A., & Sup Kwak, K. (2023b). An Improved Ensemble Learning Approach for Heart Disease Prediction Using Boosting Algorithms. *Computer Systems Science and Engineering*, *46*(3), 3993–4006. https://doi.org/10.32604/csse.2023.035244

Hasan, Md. K., Alam, Md. A., Das, D., Hossain, E., & Hasan, M. (2020). Diabetes Prediction Using Ensembling of Different Machine Learning Classifiers. *IEEE Access*, *8*, 76516–76531. https://doi.org/10.1109/access.2020.2989857

Hijazi, H., Abu Talib, M., Hasasneh, A., Bou Nassif, A., Ahmed, N., & Nasir, Q. (2021). Wearable Devices, Smartphones, and Interpretable Artificial Intelligence in Combating COVID-19. *Sensors*, *21*(24), 8424. https://doi.org/10.3390/s21248424

Jahangiri, S., & Niaki, S. (2022). An Improved Naïve Bayes Approach to Diagnose Cardiovascular Disease: A Case Study. *Research Square*. https://doi.org/10.21203/rs.3.rs-1231978/v1

Juhola, M., Joutsijoki, H., Penttinen, K., & Aalto-Setälä, K. (2018). Detection of genetic cardiac diseases by Ca2+ transient profiles using machine learning methods. *Scientific Reports*, *8*(1), 9355. https://doi.org/10.1038/s41598-018-27695-5

Lin, H., Chung, J. W., Lao, Y., & Zhao, W. (2025). An online incremental and decremental gradient boosting decision tree framework. *Submitted to the International Conference on Learning Representations (ICLR 2025)*. ICLR Conference Organization, Global.

Louridi, N., Douzi, S., & El Ouahidi, B. (2021). Machine learning-based identification of patients with a cardiovascular defect. *Journal of Big Data*, *8*(1), 133. https://doi.org/10.1186/s40537-021-00524-9

Maheshwari, V., Mahmood, M. R., Sravanthi, S., Arivazhagan, N., ParimalaGandhi, A., Srihari, K., Sagayaraj, R., Udayakumar, E., Natarajan, Y., Bachanna, P., & Sundramurthy, V. P. (2021). Nanotechnology-Based Sensitive Biosensors for COVID-19 Prediction Using Fuzzy Logic Control. *Journal of Nanomaterials*, *2021*, 1–8. https://doi.org/10.1155/2021/3383146

Mohan, S., Thirumalai, C., & Srivastava, G. (2019). Effective Heart Disease Prediction Using Hybrid Machine Learning Techniques. *IEEE Access*, *7*, 81542–81554. https://doi.org/10.1109/access.2019.2923707

Nashif, S., Raihan, Md. R., Islam, Md. R., & Imam, M. H. (2018). Heart Disease Detection by Using Machine Learning Algorithms and a Real-Time Cardiovascular Health Monitoring System. *World Journal of Engineering and Technology*, *06*(04), 854–873. https://doi.org/10.4236/wjet.2018.64057

Nassif, A. B., Shahin, I., Bader, M., Hassan, A., & Werghi, N. (2022). COVID-19 Detection Systems Using Deep-Learning Algorithms Based on Speech and Image Data. *Mathematics*, *10*(4), 564. https://doi.org/10.3390/math10040564

Prokhorenkova, L., Gusev, G., Vorobev, A., Dorogush, A. V., & Gulin, A. (2018). CatBoost: Unbiased Boosting with Categorical Features. *Advances in Neural Information Processing Systems*, 6638–6648.

Rehman, S., Rehman, E., Ikram, M., & Jianglin, Z. (2021). Cardiovascular disease (CVD): assessment, prediction and policy implications. *BMC Public Health*, *21*(1), 1299. https://doi.org/10.1186/s12889-021-11334-2

Rezvani, S., Pourpanah, F., Lim, C. P., & Wu, Q. M. J. (2024). Methods for class-imbalanced learning with support vector machines: a review and an empirical evaluation. *Soft Computing*, *28*(20), 11873–11894. https://doi.org/10.1007/s00500-024-09931-5

Selvaraj, J., Jerith, G. G., Karthikeyan, R., & Senthil, K. (2025). Assessment of CatBoost for Diabetes Prevention in Comparison to XGBoost: AI model capable of predicting the onset of diabetes. *EAI Endorsed Transactions on Internet of Things*, *11*, 1–8. https://doi.org/10.4108/eetiot.5880

Sigrist, F. (2018). Gradient and Newton Boosting for Classification and Regression. *Expert Systems with Applications*, *167*, 114080. https://doi.org/10.1016/j.eswa.2020.114080

Uddin, S., Khan, A., Hossain, M. E., & Moni, M. A. (2019). Comparing different supervised machine learning algorithms for disease prediction. *BMC Medical Informatics and Decision Making*, *19*(1), 281. https://doi.org/10.1186/s12911-019-1004-8

Weng, S. F., Reps, J., Kai, J., Garibaldi, J. M., & Qureshi, N. (2017). Can machine-learning improve cardiovascular risk prediction using routine clinical data? *PLOS ONE*, *12*(4), e0174944. https://doi.org/10.1371/journal.pone.0174944

World Health Organization. (2021). *Cardiovascular diseases*. https://www.who.int/health-topics/cardiovascular-diseases#tab=tab_1

Zhang, B., Zhang, Z., Jiang, H., Liu, Y., Zheng, L., Zhou, Y., Huang, S., & Wu, J. (2025). Bilateral Differentially Private Vertical Federated Boosted Decision Trees. *ArXiv*, 2504.21739. https://doi.org/10.48550/arXiv.2504.21739

Zou, Q., Qu, K., Luo, Y., Yin, D., Ju, Y., & Tang, H. (2018). Predicting Diabetes Mellitus With Machine Learning Techniques. *Frontiers in Genetics*, *9*, 515. https://doi.org/10.3389/fgene.2018.00515