Indonesian Journal of Electronics, Electromedical Engineering, and Medical Informatics e-ISSN: 2656-8624
Homepage: https://ijeeemi.org/; Vol. 7, No. 2, pp. 401-414, May 2025

RESEARCH PAPER

OPEN ACCES

Optimizing Categorical Boosting Model with Optuna for Anti-Tuberculosis Drugs Classification

Yosua Satria Bara Harmoni, Kartika Maulida Hindrayani, and Dwi Arman Prasetya

Department of Data Science, Universitas Pembangunan Nasional Veteran Jawa Timur, Surabaya, Indonesia

ABSTRACT

Tuberculosis is one of the leading causes of death globally, with death rate reaching 1.30 million by 2022, an increase of 3.2% compared to the previous year. Indonesia is one of the countries with the highest number of tuberculosis cases in the world. The Directly Observed Treatment Short-course (DOTS) plays a role in improving the effectiveness of tuberculosis therapy by ensuring the availability of appropriate anti-tuberculosis drugs. However, errors in drug selection can lead to therapy failure, relapse, and Multi-Drug Resistant (MDR) cases. To overcome this, classification models based on patient medical record data can be used to improve the accuracy of drug selection. This research focuses on developing classification model to determine the type of drug using Categorical Boosting algorithm optimized with Optuna using Tree-structured Parzen Estimator. The data consisted of numerical variables, such as age, treatment duration, and categorical variables, such as history of diabetes mellitus, HIV status, drug combination. The CatBoost algorithm was chosen due to its ability to handle categorical data. Hyperparameter optimization was performed to obtain the best parameters. The preprocessing stage involved memory reduction, feature normalization, and encoding on 620 data samples, which were then divided into 90% training and 10% test data. Experimental results show CatBoost model produces an initial accuracy of 90%. After applying parameter optimization techniques using Optuna, the accuracy increased to 96%, showing 6% improvement. The model is able to accurately classify drugs combination, which can support the selection of more effective therapies for tuberculosis patients. Thus, the use of SMOTE to address class imbalance combined with Optuna for hyperparameter optimization was shown to improve the accuracy of CatBoost-based classification models. This finding confirms the effectiveness of SMOTE and Optuna methods in improving the accuracy of prediction models for drug type classification, contributing the improvement of tuberculosis treatment strategies.

PAPER HISTORY

Received March 02, 2025
Accepted April 25, 2025
Published May 18, 2025

KEYWORDS

Categorical Boosting;
Anti-Tuberculosis Drugs;
Hyperparameter optimization;
Optuna;
Classification

AUTHOR EMAIL

21083010029
@student.upnjatim.ac.id
kartika.maulida.ds
@upnjatim.ac.id
arman.prasetya.sada
@upnjatim.ac.id

1. INTRODUCTION

Tuberculosis (TB), caused by the acid- and alcohol-resistant Mycobacterium tuberculosis, primarily affects the lungs but can spread to other organs [1]. In 2019, it caused 1.4 million deaths globally, with Indonesia among the top contributors, reporting 250,000 new cases and 100,000 deaths annually [2]. Despite the introduction of DOTS to improve treatment outcomes, improper drug selection continues to drive relapse and Multi-Drug Resistant (MDR) cases [3], . Based on the problem, classification models using patient data such as age, gender, contact history, and comorbidities can support optimal drug selection. However, challenges like data heterogeneity, class imbalance, overfitting, and computational inefficiency remain obstacles to model reliability [4], [5].

CatBoost has shown high accuracy in classifying drug resistance in pulmonary TB by analyzing MMP and TIMP biomarkers, further improved by decision tree-based post-processing, with strong metrics across categories (True Positive Rate), such as Healthy (87%), Sens (89%), MDR (96%), and XDR (93%) [6]. Meanwhile, LightGBM, optimized with Optuna, outperformed other models like SVM, decision tree, random forest, and XGBoost in disease classification using indigenous patient data, achieving over 90% accuracy in chronic and infectious conditions [7]. Optuna-tuned CatBoost also improved diabetes prediction accuracy from 65% to 72%, highlighting the importance of algorithm choice and hyperparameter optimization for effective diagnosis in low-resource healthcare settings [8].

Ensemble learning and hyperparameter tuning with Optuna have proven effective in disease diagnosis and drug resistance classification, achieving over 90% accuracy in many studies. CatBoost, especially when optimized with Optuna, shows strong performance in medical classification [9], [10]. However, its use for anti-tuberculosis drug type classification remains limited. This study addresses that gap by developing an optimized CatBoost model to improve prediction accuracy and

Corresponding author: Kartika Maulida Hindrayani, kartika.maulida.ds@upnjatim.ac.id, Department of Data Science, Universitas Pembangunan Nasional Veteran Jawa Timur, Jl. Raya Rungkut Madya Gunung Anyar, Surabaya, 60294, Indonesia.
DOI: https://doi.org/10.35882/ijeeemi.v7i2.92

support clinical decision-making, offering a more efficient solution than previous approaches.

The ensemble learning approach with hyperparameter tuning has been proven effective in improving model accuracy. This research applies CatBoost to handle data classification with a combination of categorical and numerical features. The main advantage of this algorithm is its ability to process categorical data without complex encoding processes, making it more efficient and less error-prone [11]. With gradient boosting, the model is built incrementally to correct previous errors, resulting in more accurate predictions [12]. However, hyperparameter optimization remains a challenge. Therefore, this study uses optuna with the Tree-structured Parzen Estimator (TPE) algorithm to find the optimal combination such as learning rate, tree depth, and number of iterations [13]. This approach is expected to improve the performance of the model in the classification of drug regimens for tuberculosis patients, such as isoniazid, rifampicin, and pyrazinamide. Evaluation was conducted using confusion matrix to compare the performance of the developed model.

The contributions of this research include: 1) development of a CatBoost-based classification model optimized with optuna to improve accuracy in the selection of anti-tuberculosis drug regimens, 2) application of hyperparameter tuning technique based on Tree-structured Parzen Estimator (TPE) to find the optimal parameter combination, 3) evaluation of model performance using confusion matrix and k-fold cross validation to ensure the accuracy of classification of the type of drug consumed by patients, and 4) providing new insights in the utilization of ensemble learning and model optimization in clinical decision-making systems.

The structure of this research consists of several main sections. Section II describes the dataset used, the proposed method, and the training and testing schemes applied. Section III presents the experimental results obtained from the application of the CatBoost algorithm and Optuna optimization. Furthermore, Section IV analyzes related research, evaluates the performance of the model and discusses the limitations. Finally, Section V contains conclusions that summarize the research objectives, key findings, and future directions for further development. Each section is structured to provide a clear and logical progression of the study.

2. MATERIALS AND METHOD

This study outlines the research workflow to illustrate the sequential steps conducted during each testing phase. Fig. 1 presents the detailed flowchart representing the overall research process.

A. Data Collection

Various types of data can be collected through methods such as data warehouses, big data, and artificial intelligence, enabling executives to gain deeper insights into current conditions and make well-informed decisions [14], [15]. In this research, the dataset used for developing the tuberculosis classification model was sourced from medical records of patient tuberculosis RSUD Dr. Iskak which published in Ministry of Health's Tuberculosis Information System website (sitb.kemkes.go.id) from 2020 – 2024 data. The dataset comprises 620 samples with attributes including age (in years), gender (male/female), type of TB diagnosis, and other variables summarized in Table 1.

Inclusion criteria required that patients were diagnosed with tuberculosis, with or without comorbidities, had undergone various medical examinations, and possessed complete medical records. To maintain the relevance and ethical integrity of the analysis, a feature exclusion process was applied. Personal identifiers such as medical record numbers and patient names were removed to uphold data privacy, while the employment attribute was excluded due to its limited relevance to the predictive outcome. Additionally, the TCM test result feature was omitted because it contained a high proportion of missing values, which could negatively impact the performance and accuracy of the classification model.

The patient sample in this study captures a wide range of demographic and clinical characteristics, representing individuals across different age groups, from young adults to the elderly, and a balanced distribution of genders. It includes patients both with and without a history of diabetes mellitus, as well as those who tested positive or negative for HIV. The dataset further reflects diversity in exposure history to tuberculosis, encompassing both those with and without known contact. Variations in disease classification, including pulmonary and extrapulmonary tuberculosis, are also well-represented. Additionally, the patients come from a variety of occupational backgrounds ranging from students and the unemployed to various working sectors and originate from multiple districts and cities across Indonesia, ensuring a comprehensive reflection of the broader TB population.

Table 1. Overview of the TB dataset features, including feature names, types, and descriptions used in the data analysis process.

Feature	Description
Age	Age. patient
Gender	Gender
Contact	History of patient contact with TB patients
Diagnosis Type	Type of patient diagnosis
Anatomical Classification	Anatomical location of TB
Status	Patient's previous medication history
DM History	Patient's history of diabetes mellitus
HIV Test	Patient's HIV test result
HIV Classification	HIV status of the patient

Corresponding author: Kartika Maulida Hindrayani, kartika.maulida.ds@upnjatim.ac.id, Department of Data Science, Universitas Pembangunan Nasional Veteran Jawa Timur, Jl. Raya Rungkut Madya Gunung Anyar, Surabaya, 60294, Indonesia.
DOI: https://doi.org/10.35882/ijeeemi.v7i2.92

Child TB Score	TB assessment score in
X-ray	Radiology examination results
Treatment Duration	Duration of patient treatment in days.
Treatment Outcome	Patient's final treatment outcome
OAT Combination	Types of Anti-tuberculosis Drug alloys
Treatment Combination	Code of the type of medicine used

The target variable in this study classifies patients based on their therapeutic requirements to help optimize treatment outcomes. Three treatment regimens are considered: the first regimen, 2(HRZ)/4(HR), serves as the standard approach where H stands for isoniazid, R for rifampicin, and Z for pyrazinamide; the second regimen, 2(HRZE)/4(HR) - KDT Daily Dose, introduces ethambutol (E) into the initial phase to improve patient adherence; and the third regimen, 2(HRZE)/4(HR)3, modifies the continuation phase to be administered three times weekly to enhance treatment effectiveness [16]. A total of 233 patients received the first treatment, 303 followed the KDT Daily Dose regimen, and 84 were treated under the third category.

B. Data Preprocessing

Data preprocessing in this study included handling missing values, transforming data that convert treatment dates to treatment duration, and applying label encoding to transform categorical features into numerical format [17].

1. **Handling Missing Value**
   Missing values present across several features were systematically addressed to maintain data consistency and minimize biases that could otherwise degrade predictive accuracy [18].
2. **Data Transfomation**
   Treatment start and end dates, initially stored in date format, were transformed into a numerical feature representing treatment duration, ensuring that temporal information could be effectively utilized during model training.

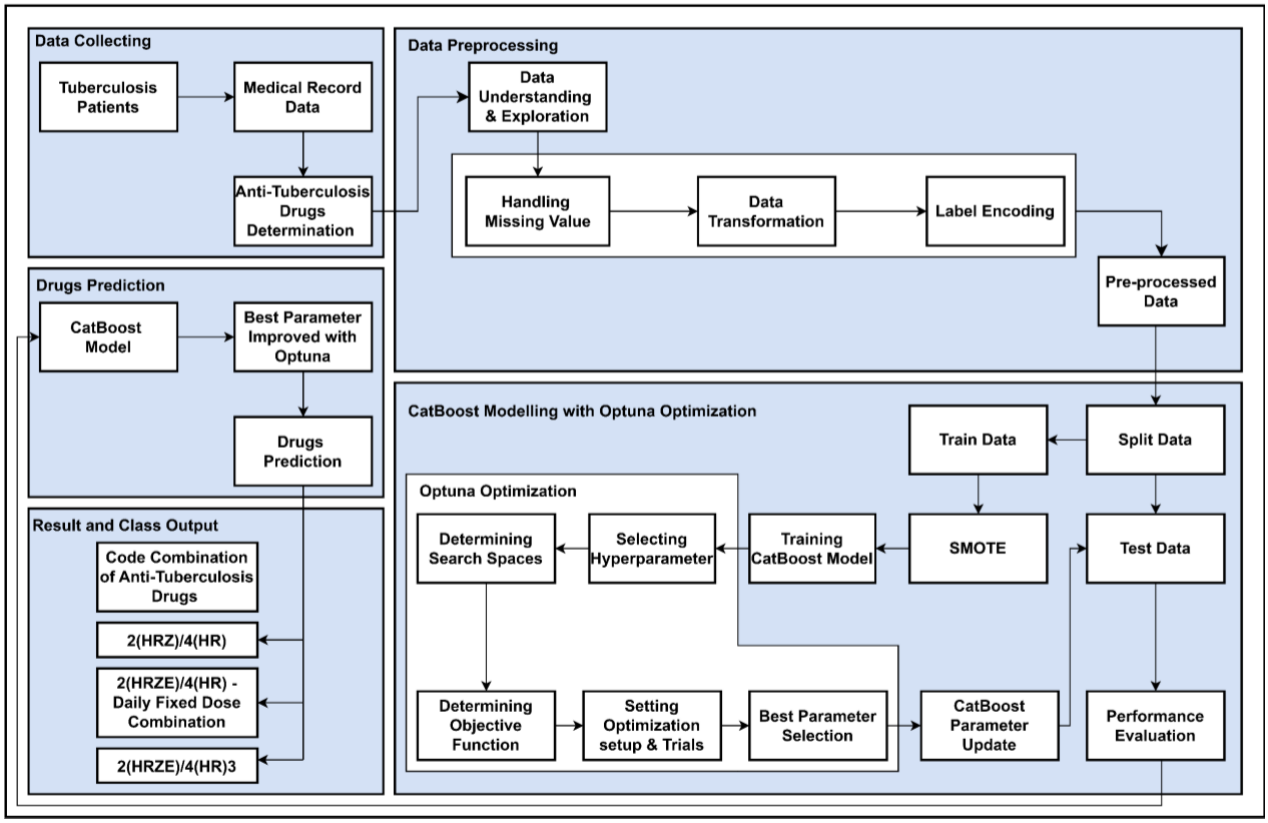


Fig 1. Research block diagram visualizes the systematic flow of the research process.

3. **Label Encoding**
   Categorical variables, such as gender and diagnosis type, were encoded into numerical values using label encoding, allowing algorithms to process these features correctly while preserving their categorical significance. These steps collectively improved the quality of the input data and strengthened the overall performance and reliability of the classification model [19].

Corresponding author: Kartika Maulida Hindrayani, kartika.maulida.ds@upnjatim.ac.id, Department of Data Science, Universitas Pembangunan Nasional Veteran Jawa Timur, Jl. Raya Rungkut Madya Gunung Anyar, Surabaya, 60294, Indonesia.
DOI: https://doi.org/10.35882/ijeeemi.v7i2.92

C. Data Splitting

Data splitting is a crucial technique for model validation, involving the division of a dataset into training and testing subsets. The model is trained on the training data and evaluated on the test data, ensuring an unbiased evaluation and minimizing the risk of overfitting. While the 90:10 ratio is often used, where 90% of the data is allocated for training and 10% for testing, other ratios like 70:30 or 60:40 are also common [20]. In this study, a 90:10 ratio was chosen, with 90% of the data used for training and 10% for testing. This choice was made to maximize the amount of data available for model training, which is particularly useful in scenarios where larger datasets are available and a higher training proportion is desirable Data splitting is a crucial technique for model validation, involving the division of a dataset into training and testing subsets Data splitting is a crucial technique for model validation, involving the division of a dataset into training and testing subsets [5], [21].

In addition, cross-validation is a widely used method in machine learning to improve the reliability of model performance assessments and minimize the risk of overfitting. In this study, the dataset is partitioned into ten roughly equal subsets through the Stratified K-Fold Cross-Validation method. During each of the five iterations, one fold is reserved for testing while the remaining four are utilized for training. The stratification ensures that each subset mirrors the original class distribution, preserving class balance throughout the folds and thereby strengthening the consistency and reliability of the model evaluation [22].

D. Oversampling with SMOTE

Synthetic Minority Oversampling Technique (SMOTE) is a statistical method used to balance the amount of data between minority and majority classes. Different from random oversampling, SMOTE does not simply duplicate data, but rather generates new synthetic samples using the k-NN algorithm [23]. Fig. 2 shows the ilustration of SMOTE.

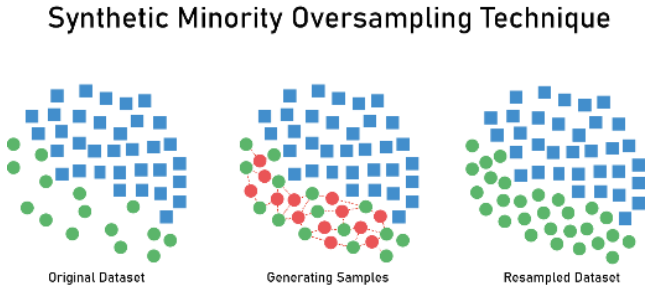


Fig. 2. SMOTE technique applied to balance class distribution in training dataset.

Fig. 2 starts with randomly selecting one data from the minority class, then finding the nearest neighbor using *k-NN*. Synthetic data is generated by connecting the two through a line and forming a convex combination. SMOTE creates synthetic samples to balance the minority class. Eq. (1) finds the *k-nearest neighbors of* each minority data, and then forms a new sample based on the ratio of the number of majority and minority data [24].

$$N\% = \frac{number\ of\ majority\ class}{number\ of\ minority\ class} \times\ 100\% \quad (1)$$

Nearest neighbor is chosen based on the calculation of the Euclidean distance between two data points [25]. For example, given two data points each having p dimensions, namely XT =[x_1,x_2,...,x_n] and YT=[y_1,y_2,...,y_n], the Euclidian distance of Eq. (2) is:

$$d(x,y) = \sqrt{\{(x_1 - y_1)^2 + \cdots + (x_n - y_n)^2\}} \quad (2)$$

Each term $(x_i - y_i)^2$ quantifies the squared deviation along each dimension. Once proximity is established, synthetic instances are generated by interpolating between the target sample and its nearest neighbors, thereby enriching the minority class with new, representative data points.Once the data with the closest distance is found, *SMOTE* will generate new data (*synthetic data*) [26]. It shows in Eq. (3).

$$x_{\{syn\}} = x_i + \delta \times (x_{\{knn\}} - x_i) \quad (3)$$

x_{syn} is the replicated synthetic data, while xi is the replicated data. x_{knn} refers to the data with the closest distance from xi, and δ is a random number between 0 and 1 used in the process of forming new samples.

E. CatBoost

CatBoost is a robust ensemble classification algorithm that enhances predictive performance through Gradient Boosting Decision Trees (GBDT), a method that sequentially trains models to correct errors from previous iterations. This model works by iteratively refining predictions, adding new weak learners that focus on correcting residual errors from previous models. The learning rate controls how much weight each new learner contributes to the overall model, progressively minimizing the loss function with each step. In terms of categorical feature handling, CatBoost computes the value for each category by considering the frequency of the class label and applying a smoothing technique to mitigate the impact of sparse categories [27], [28].

CatBoost is an advanced gradient boosting algorithm optimized for structured data, especially those containing categorical features. Unlike traditional methods that require extensive preprocessing, CatBoost handles categorical variables natively using permutation-driven target statistics combined with Bayesian smoothing to reduce bias from low-frequency categories. It also automatically manages missing values by assigning them to optimal splits during tree construction, eliminating the need for manual imputation while preserving data integrity [29].

To prevent overfitting, CatBoost integrates several key strategies. Ordered boosting is used to avoid target leakage by ensuring that training data is processed in a sequential and randomized manner, using only prior

Corresponding author: Kartika Maulida Hindrayani, kartika.maulida.ds@upnjatim.ac.id, Department of Data Science, Universitas Pembangunan Nasional Veteran Jawa Timur, Jl. Raya Rungkut Madya Gunung Anyar, Surabaya, 60294, Indonesia.
DOI: https://doi.org/10.35882/ijeeemi.v7i2.92

information. Regularization is further reinforced through smoothed encoding of categorical values, which stabilizes model behavior on sparse data. Additionally, CatBoost supports early stopping based on validation loss, allowing the training process to halt before overfitting occurs. These mechanisms collectively enable the model to generalize well and maintain high predictive performance even on limited or reused datasets [30], [31]. Its architecture can be seen at Fig. 3.

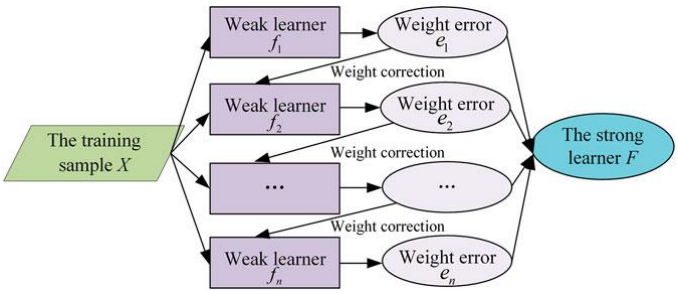


Fig. 3. CatBoost architecture illustrating the gradient boosting training process with high efficiency and performance.

The algorithm also includes mechanisms to prevent target leakage, ensuring model reliability by avoiding prediction shifts caused by improper data handling. To further improve accuracy, CatBoost employs ordered boosting to avoid bias from target leakage and uses techniques like random shuffling to enhance the model's generalization capability. Additionally, it minimizes bias during iterative learning by incorporating adjustments for the finite size of the training data, which helps preserve model performance even when reusing the dataset across multiple iterations. This approach ensures that the model remains adaptable and capable of generalizing effectively to new data. Unlike traditional boosting techniques, CatBoost excels in handling categorical variables without requiring extensive preprocessing, making it especially effective for structured datasets [32]. Its ability to process categorical variables without extensive preprocessing makes it particularly effective for structured datasets, as shown in Eq. (4) [11].

$$F_m\,(X) = F_m - 1(X) + a_m h_m(X, r_m - 1) \quad (4)$$

Eq. (4) describes an iterative learning process commonly used in *gradient boosting* frameworks. In this formulation, $F_m(X)$ represents the ensemble prediction at the $m-th$ iteration, which is updated by adding a new weak learner h_m, trained specifically on the residual errors r_{m-1} from the previous model $F_{m-1}(X)$. The term a_m denotes the learning rate, a scalar that controls the contribution of the new learner to the overall model. By progressively refining the prediction through the correction of residuals, this approach effectively minimizes the loss function and enhances the model's accuracy with each subsequent step.

$$ctr_i = \frac{countIn(Class) + prior}{totalCount + 1} \quad (5)$$

In Eq. (5), ctr_i is the $i-th$ categorical value, *countIn(Class)* counts label frequency after i in a random order, *totalCount* is the total occurrences, and prior is a constant. Model performance can be tuned via iterations and tree depth. Preventing prediction shift resulting from target leakage in target shuffling is essential to ensure the integrity and reliability of the model's performance. CatBoost uses ordered boosting, as gradient estimates are typically unknown [33].

$$h_t = \arg\min_{h \in \mathcal{H}} \frac{1}{n} \sum_{k=1}^{n} [g_t(x_k, y_k) - h(x_k)]^2 \quad (6)$$

To minimize the loss function in iterative learning frameworks, models typically rely on the $argmin$ operation, which identifies the parameter values that yield the lowest possible error. First, the conditional distribution in the Eq. (6) of the gradient $g_t(x_k, y_k) \mid x_k$ shifts from the distribution on the test data $g_t(x, y) \mid x$. Second, this shift causes the *base predictor* h_t to be biased towards the theoretical solution. Third, this bias affects the generalization of model F_t due to target leakage, where gradients are computed using the same target data from prior training steps., F_{t-1}. This causes the conditional distribution $F_{t-1}(x_k) \mid x_k$ on the training data x_k to differ from the distribution $F_{t-1}(x) \mid x$ on the test data [33]. If two independent samples D_1 and D_2 are used to estimate h_1 and h_2, then Eq. (7):

$$E[F_2(x)] = f(x) - \frac{1}{n} c_2 \left(x_2 - \frac{1}{2}\right) + O\left(\frac{1}{n^2}\right) \quad (7)$$

Eq. (7) expresses the expected value of the model's prediction at the second iteration, denoted by $E[F_2(x)]$, as an approximation of the true function $f(x)$. This expression includes a correction term, $\frac{1}{n} c_2 \left(x_2 - \frac{1}{2}\right)$, which captures the first-order bias introduced by the finite size of the training data, with the constant c_2 representing the sensitivity to that bias. The final term, $\left(\frac{1}{n^2}\right)$, accounts for higher-order effects that diminish more rapidly as the dataset size n increases. In case, reusing the same dataset repeatedly across iterations induces systematic bias that scales inversely with data volume. Therefore, ensuring sufficient data variability whether through sampling strategies or augmentation is essential to preserve the model's ability to generalize effectively.

F. Hyperparameter Optimization

Optimization helps enhance system performance, improve decision-making accuracy, and maximize the efficiency of resource utilization [34], [35]. This study uses Optuna for hyperparameter optimization, leveraging dynamic parameter space and pruning to accelerate the search. As shown in Fig. 4, the process defines objectives and parameter ranges, evaluates combinations iteratively, and prunes low-potential options to focus on the most promising until the maximum iteration is reached [36], [37], [13].

Corresponding author: Kartika Maulida Hindrayani, kartika.maulida.ds@upnjatim.ac.id, Department of Data Science, Universitas Pembangunan Nasional Veteran Jawa Timur, Jl. Raya Rungkut Madya Gunung Anyar, Surabaya, 60294, Indonesia.
DOI: https://doi.org/10.35882/ijeeemi.v7i2.92

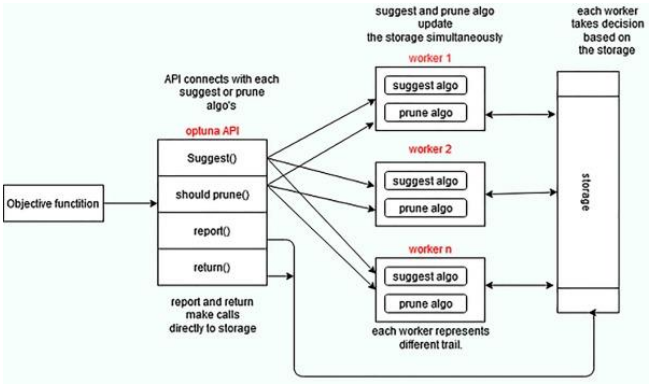


Fig 4. Structural architecture of the Optuna hyperparameter optimization framework.

Optuna uses an improved Bayesian optimization with a Parzen tree optimizer, analyzing past trial data to suggest better hyperparameter combinations. By updating search areas adaptively and applying Bayesian probabilities (Eq. 8), it efficiently refines configurations to enhance model performance [38].

$$l(\theta) = p(y < a|\theta, H)$$
$$z(\theta) = p(y < a|a, H) \quad (8)$$

In this case, $a = \min\{y1, y2, \ldots, yn\}$; $H = \{(\theta 1, y1), (\theta 2, y2), \ldots, (\theta n, yn)\}$, where H is a history vector containing pairs of *hyperparameters*, and $l(\theta)$ is the density formed from observations of various observations {θn}. The TPE algorithm in Optuna optimizes hyperparameters by maximizing the ratio *l(θ)/z(θ)* to minimize validation loss. It updates the search iteratively, focusing on the most promising areas, making it more efficient than manual tuning in complex, high-dimensional spaces [39].

Optimizing the performance of the CatBoost model often involves fine-tuning key hyperparameters that govern the learning process and model structure. Parameters that used such as learning rate, tree depth, number of iterations, and various forms of regularization (e.g., l2 lef reg) are typically the focus of this tuning process. The primary goal is to achieve an effective balance between model complexity and its generalization ability, reducing the risk of overfitting while ensuring the model captures essential data patterns [40].

Automated frameworks like Optuna significantly streamline this hyperparameter optimization process, often utilizing advanced samplers such as Tree-structured Parzen Estimator (TPE). TPE adopts a Bayesian approach, dynamically building probabilistic models based on historical performance from previous experiments one model reflecting successful hyperparameter configurations and another for less successful ones. By intelligently sampling areas likely to yield superior results based on these models, TPE efficiently navigates the complex hyperparameter space, enabling faster convergence toward optimal configurations compared to exhaustive search strategies like random or grid search [41].

G. Performance Evaluation

A thorough grasp of domain-specific knowledge is crucial when applying classification models in healthcare, particularly in tasks such as determining the most appropriate drug regimen for individual patients. In this context, the implications of misclassification can be clinically significant especially when incorrect predictions lead to suboptimal treatment choices, reduced therapeutic effectiveness, or the development of drug resistance. As such, evaluating model performance should not rely solely on statistical accuracy but must also consider the real-world impact of predictive errors on patient outcomes and treatment success [42].

This study uses a confusion matrix to evaluate model performance by comparing predicted and actual values [43], [44]. As shown in Table 2, it clearly illustrates how well the model's predictions align with real outcomes.

Table 2. Confusion matrix showing classification prediction result distribution and predicted class outcomes.

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

The confusion matrix formula is used to perform calculations that produce evaluation metrics such as accuracy, recall, precision, and F1-Score [45]. To thoroughly evaluate the CatBoost model's performance, several specific metrics are utilized, including accuracy, precision, recall, and F1-score. In the modelling process, employing these metrics allows for a more comprehensive and nuanced understanding of the model's predictive capabilities across various practical contexts.

1. Accuracy, represents the proportion of correctly predicted instances relative to the total number of predictions, offering a general assessment of the model's overall performance. However, accuracy alone can be misleading, particularly when the dataset is imbalanced. It is also a metric used to assess how well the model performs the classification correctly, which shows in Eq. (9):

$$\left[\text{Accuracy} = \frac{TP + TN}{TP + FP + FN + TN}\right] \quad (9)$$

2. Recall, measures the percentage of positive data that was correctly identified by the system. The higher the recall value, the better the class is recognized. Meanwhile, recall (or sensitivity) evaluates the proportion of actual positive cases that are correctly identified by the model, which is critical in applications where missing positive cases can have significant consequences which shows in Eq. (10):

$$\left[\text{Recall} = \frac{TP}{TP + FN}\right] \quad (10)$$

Corresponding author: Kartika Maulida Hindrayani, kartika.maulida.ds@upnjatim.ac.id, Department of Data Science, Universitas Pembangunan Nasional Veteran Jawa Timur, Jl. Raya Rungkut Madya Gunung Anyar, Surabaya, 60294, Indonesia.
DOI: https://doi.org/10.35882/ijeeemi.v7i2.92

3. Precision, indicates the ratio of the number of correctly classified positive data compared to the total data categorized as positive. To address accuracy problem, precision is also considered, which measures the proportion of correctly predicted positive instances among all instances predicted as positive, reflecting the model's ability to minimize false positives which shows in Eq. (11):

$$\left[\text{Precision} = \frac{TP}{TP + FP}\right] \quad (11)$$

4. F1-Score, the harmonic mean of precision and recall. F1-Score values range from 0 to 1, where a value of 1 indicates perfect performance. Employing these metrics allows for a more comprehensive and nuanced understanding of the model's predictive capabilities across various practical contexts which shows in Eq. (12):

$$\left[F1\text{-Score} = 2 \times \frac{1}{\frac{1}{\text{Recall}} + \frac{1}{\text{Precision}}}\right] \quad (12)$$

Furthermore, the Area Under the Curve (AUC) is utilized to measure the model's ability to differentiate between true positive and false positive rates across varying threshold values. A higher AUC score reflects stronger model capability in correctly distinguishing between classes [46]. The AUC can be calculated in Eq. (13).

$$AUC = \int_0^1 TPR \; d(FPR) \quad (13)$$

where the True Positive Rate (TPR) represents the proportion of actual positives correctly identified, and the False Positive Rate (FPR) captures the proportion of negatives incorrectly classified as positive. AUC values range from 0 to 1, with a score of 1 indicating flawless classification and 0.5 suggesting random guessing. In addition, key metrics such as precision, recall, and the F1-score are essential for optimizing model evaluation.

Determining an appropriate anti-tuberculosis drug regimen relies on the accuracy of classification models analyzing patient data. Metrics like accuracy, recall, precision, F1-Score, and AUC ensure predictions align with actual outcomes and are reliable for clinical practice. While AUC measures the model's ability to differentiate between classes, recall and precision are essential to minimize misclassification errors, which could lead to incorrect drug regimen choices. In tuberculosis treatment, misclassifications can result in increased drug resistance, treatment failure, or severe side effects. Thus, optimizing classification models is essential to improve therapeutic efficacy and reduce risks from ineffective treatment.

3. RESULTS

A. Data Preprocessing

Data preprocessing successfully optimized memory usage by converting numeric data types to smaller sizes, speeding up data processing without compromising the quality of analysis. In addition, merging the Treatment Start Date and Treatment End Date columns into Treatment Duration in days resulted in numerical data that was easier to process. The label encoding technique was also applied to convert categorical variables into numerical format, facilitating processing by machine learning algorithms.

1. **Handling Missing Value**
   In handling missing values, empty entries found in the "HIV Classification" and "Treatment Outcome" features were filled with the label "Unknown" to ensure data completeness and prevent potential biases that could disrupt the model's learning process.
2. **Data Transformation**
   The data transformation process is done by combining two time columns, namely Treatment Start Date and Treatment End Date, to produce a new variable called Treatment Duration which is calculated in days. This value is obtained by calculating the difference between the end date and the treatment start date. Initially, the data in both columns were of type Date, but for further analysis purposes, the data type was converted to numeric format in Table 3.

Table 3. Raw and transformed data show date becoming treatment duration features.

	Raw Data	Transformed Data
Feature	Start Date End Date	Duration of Treatment

3. **Label Encoding**
   In data preprocessing, categorical variables in the form of text need to be converted into a numerical format in order to be processed by machine learning algorithms. One commonly used method is label encoding, which converts each category into a unique numeric value [47]. In Python, Scikit-Learn provides the LabelEncoder function for this purpose. Using LabelEncoder, each unique label in a categorical variable is mapped to an integer, such as 0, 1, 2, and so on. Table 4 allows machine learning algorithms to understand and process categorical data more effectively.

Table 4. Raw and transformed data show date becoming treatment duration features.

Feature	Before	After
Gender	Male, Female	
Contact	Yes, No	
Diagnosis Type	Bacteriological, Clinical	
Anatomical Classification	Pulmonary, Extrapulmonary	
OAT Combination	Child, One	

Corresponding author: Kartika Maulida Hindrayani, kartika.maulida.ds@upnjatim.ac.id, Department of Data Science, Universitas Pembangunan Nasional Veteran Jawa Timur, Jl. Raya Rungkut Madya Gunung Anyar, Surabaya, 60294, Indonesia.
DOI: https://doi.org/10.35882/ijeeemi.v7i2.92

Treatment History	New, Treated After Failure of Category 1, Treated After Treatment Interruption, Relapse, Others, Unknown	0-5
HIV Test	Negative, Non-reactive, Positive, Reactive, Unknown	
Treatment Outcome	Failure, Death, Complete Treatment, Treatment Interruption, Cured, Unknown	
DM History	No, Yes, Unknown	
HIV Classification	Negative, Positive, Unknown	
X-ray	Negative, Positive, TDL	

The preprocessing steps applied in this study are expected to enhance model accuracy by providing cleaner and more structured data, while also reducing the risk of overfitting. This approach aims to ensure that the model can generalize more effectively when encountering previously unseen data.

B. Oversampling Result

Before the application of SMOTE, the class distribution within the "Treatment Combination" feature was notably imbalanced, consisting of 233 instances for the 2(HRZ)/4(HR) regimen, 303 instances for the 2(HRZE)/4(HR) - KDT Daily Dose regimen, and only 84 instances for the 2(HRZE)/4(HR)3 regimen. It can be seen in Fig. 5.

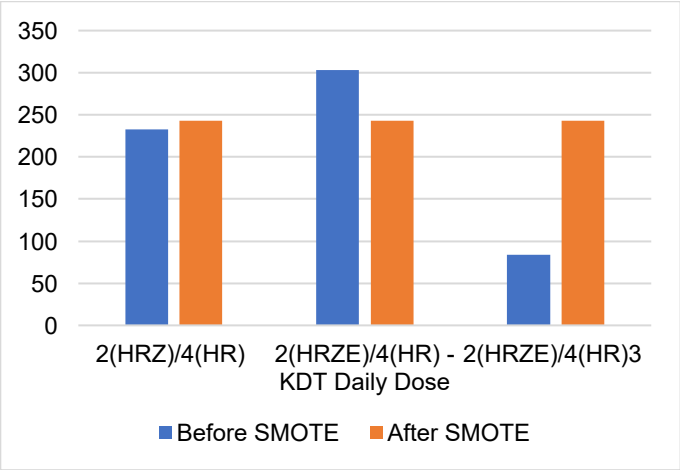


Fig. 5. Class distribution comparison before and after applying the SMOTE oversampling technique.

In addition, the Chi-Square test for class proportions confirmed a statistically significant imbalance, yielding a Chi-Square value of 110.0323 and a p-value of 0.0000. To address this issue, SMOTE oversampling was applied, resulting in a perfectly balanced dataset, with each class now containing 273 instances. Following this adjustment, the Chi-Square test produced a value of 0.0000 and a p-value of 1.0000, indicating no statistically significant difference among the classes and confirming that the class distribution had been successfully equalized. This balanced data structure is expected to improve model training by providing a more representative learning process, enhancing prediction accuracy, and minimizing the risk of model bias and overfitting. The metrics can be seen in Table 5.

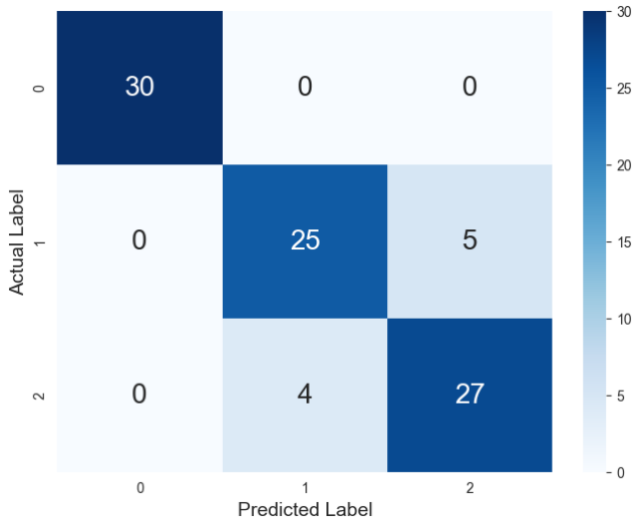
Table 5. Chi-Square test results and p-value comparisons for feature relevance assessment.

Metrics	Before	After
Chi - Square	110.03	0.0000
P - Value	0.0000	1.0000

Class imbalance within medical datasets can critically undermine the performance of predictive models, particularly when dealing with complex treatment classifications such as tuberculosis regimens. Ensuring balanced class representation during model training is essential to provide equitable learning across all patient groups. This balance not only enhances predictive accuracy but also minimizes the risk of algorithmic bias, ultimately supporting more reliable and fair treatment recommendations.

C. CatBoost Result

The CatBoost model with SMOTE demonstrated strong classification performance, with high accuracy achieved using a 90:10 train-test split. Class 0 (2(HRZ)/4(HR)) was predicted with near-perfect accuracy, while some misclassifications occurred between class 1 (2(HRZE)/4(HR) - KDT Daily Dose) and class 2 (2(HRZE)/4(HR)3), likely due to similar feature patterns. Although SMOTE improved class balance, overlapping decision boundaries remain a challenge. Further enhancements through feature engineering and hyperparameter tuning with Optuna are recommended to boost precision, particularly in distinguishing between closely related treatment regimens.



Corresponding author: Kartika Maulida Hindrayani, kartika.maulida.ds@upnjatim.ac.id, Department of Data Science, Universitas Pembangunan Nasional Veteran Jawa Timur, Jl. Raya Rungkut Madya Gunung Anyar, Surabaya, 60294, Indonesia.
DOI: https://doi.org/10.35882/ijeeemi.v7i2.92

Fig. 6. Confusion matrix each class results for the CatBoost classification model.

Based on the Fiq. 6, the performance of the model shows quite good results with an accuracy value of 90%. In addition, other evaluation metrics such as precision, recall, and F1-score all reached 90%. Before applying SMOTE, the CatBoost model achieved an accuracy of 89%, with strong performance across various metrics. SMOTE provides an improvement of 1 percent each. These values are calculated using the weighted average method, which considers the proportion of each class in the data, thus Table 6 providing a comprehensive overview of the model's effectiveness in classification.

Table 6. Weighted average performance metrics of CatBoost model classification results based on confusion matrics.

Metrics	Result			
	Accuracy	Precision	Recall	F1
Weighted Average	0.90	0.90	0.90	0.90

D. Hyperparameter Optimization Result

Referring to Table 5, the model produces an accuracy value of 90%. Although it is already quite high, there is still an opportunity to improve the performance of the model by searching for more optimal hyperparameters. The optimization process was conducted with a total of 300 trials, which allowed for a more in-depth exploration of the parameters. After optimization, the best parameters were applied to the CatBoost model to improve classification performance. Details of the parameters used in this process are presented in Table 7.

Table 7. Hyperparameter values used in the final CatBoost model configuration based on Optuna optimization.

Hyperparameters	Search Domain	Set Values
iterations	100, 1000	300
depth	4, 10	4
learning_rate	0.01, 0.3	0.286
l2_leaf_reg	1e-5, 10	0.0002
border_count	32, 256	201
random_strength	0.0, 1.0	0.008
bagging_temperature	1e-5, 10	4.624

As a result, after updating the parameters, the model performance was evaluated through confusion matrix visualization which can be seen in the following figure. The application of CatBoost with updated parameters aims to improve the accuracy of the model in predicting the desired results. This parameter update is proven to provide improvements by reduction in prediction errors that occur in the model. After the update, the confusion matrix provides a clearer picture of the model's performance [48]. Fig. 7 shows a significant improvement in the classification accuracy produced by CatBoost.

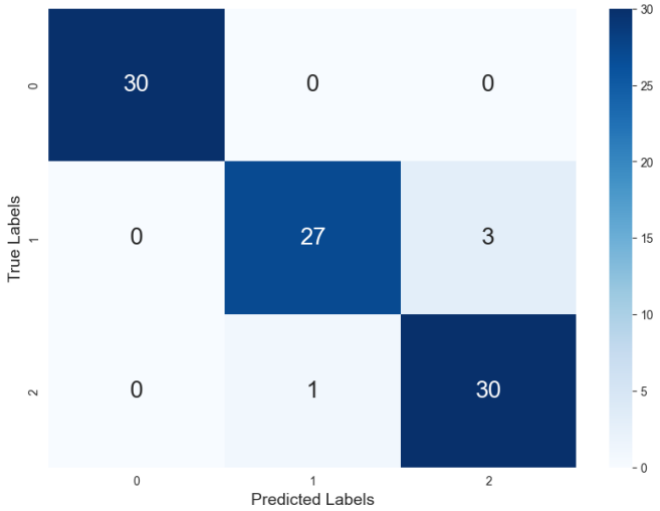


Fig. 7. Confusion matrix each class results for the final CatBoost classification model.

Looking at the results shown in the Fig. 7, there is an accuracy value of 96%. Table 8 indicates that the CatBoost model has demonstrated near-optimal performance, close to the ideal state. The classification analysis revealed that the regimen 2(HRZE)/4(HR) – KDT Daily Dose was most frequently misclassified, with 3 out of 30 instances incorrectly labeled as 2(HRZE)/4(HR)3. This is reflected in its lower recall score compared to the other classes, suggesting that the model often failed to correctly identify all true instances of this regimen.

Table 8. Weighted average performance metrics of final catboost model classification results.

Measure	Result				
	Acc	Precision	Recall	F1	AUC
Weighted Average	0.96	0.96	0.96	0.96	0.99

4. DISCUSSION

A. Performance Evaluation & Comparison

Based on the analysis conducted, there are two model testing scenarios involving the CatBoost classification approach and the combination of CatBoost classification with optimization using Optuna. All models were executed and analyzed for performance through a number of evaluation metrics. The results of this evaluation process were then compared to identify any improvement or degradation in performance between each model. A summary of the performance comparison results of the four models can be seen in detail in Table 9.

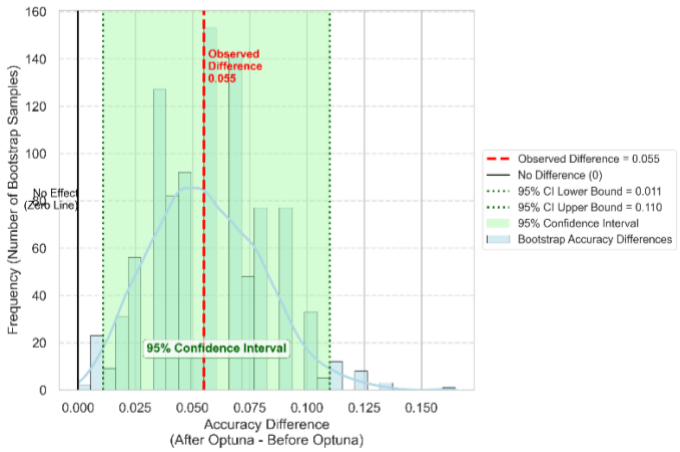
Table 9. Comparison of performance metrics based on catboost model classification results.

Measure	Method	Result			
		Acc	Precision	Recall	F1
	CatBoost	0.90	0.90	0.90	0.90

Corresponding author: Kartika Maulida Hindrayani, kartika.maulida.ds@upnjatim.ac.id, Department of Data Science, Universitas Pembangunan Nasional Veteran Jawa Timur, Jl. Raya Rungkut Madya Gunung Anyar, Surabaya, 60294, Indonesia.
DOI: https://doi.org/10.35882/ijeeemi.v7i2.92

Weighted Average	CatBoost + Optuna	0.96	0.96	0.96	0.96

The initial performance of the CatBoost model, prior to hyperparameter tuning, demonstrated strong results with balanced metrics, such as accuracy, precision, recall, and F1-score for each reaching 90%, highlighting its effectiveness in handling datasets with predominantly categorical variables. Nevertheless, the model's capacity to generalize under increased data complexity remained limited. To address this, Optuna was employed for hyperparameter optimization, leading to a consistent 6% improvement across all major performance indicators without compromising prediction accuracy. Post-optimization evaluation using 5-fold cross-validation yielded an average accuracy of 0.93 and an AUC of 0.99, underscoring the enhanced robustness of the model. Additionally, the reliability of these improvements was verified through a bootstrap significance test with 1000 resampling iterations, confirming the statistical soundness of the optimization process. A summary of the testing results of the models can be seen in detail in Fiq 8.



Fiq 8. Comparison of performance metrics based on catboost model classification results.

Referring to Fig. 8, The statistical analysis using 1000 bootstrap iterations confirmed a significant improvement in model accuracy following hyperparameter tuning with Optuna. The observed increase of 0.055 in accuracy, rising from 0.90 to 0.96, was supported by a one-tailed p-value of 0.008, indicating that the enhancement is unlikely to be due to random variation. Furthermore, the 95% confidence interval for the accuracy difference, ranging from 0.011 to 0.110, lies entirely above zero, reinforcing the conclusion that the optimized model delivers a statistically and practically superior performance. The results were influenced by the performance of several CatBoost parameters. Fig. 9 shows the hyperparameter importance that contributed most significantly to the improved model performance after the tuning process.

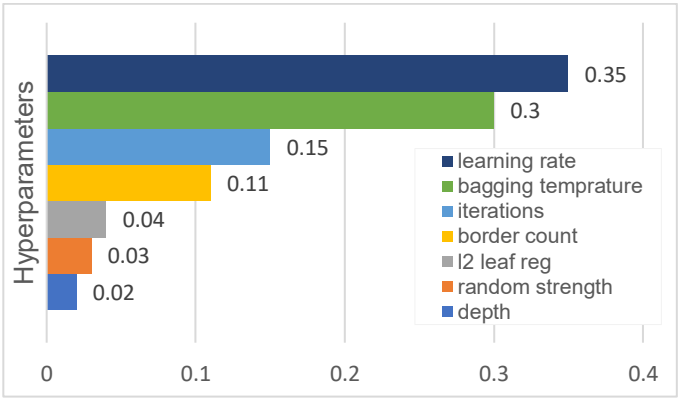


Fig. 9. Hyperparameter importances influencing CatBoost model performance and accuracy based on Optuna optimization.

Based on Fig. 9, Although the overall results demonstrated a clear performance improvement following hyperparameter optimization with Optuna, a deeper examination of the individual hyperparameters further enriches the discussion. Sensitivity analysis based on the importance scores reveals that the most influential hyperparameters contributing to model enhancement were the learning rate (0.35) and bagging temperature (0.30). These two parameters showed the greatest impact on the model's predictive ability, indicating their critical role in balancing the speed of convergence with generalization. Other parameters such as iterations (0.15) and border count (0.11) also contributed moderately, while l_2 leaf reg (0.04), random_strength (0.03), and depth (0.02) showed comparatively lower influence. Understanding the relative contribution of each hyperparameter offers valuable insight into the model's sensitivity and serves as a strategic reference point for future optimization efforts, especially when dealing with complex, high-dimensional datasets [49].

Compared to traditional Decision Tree and ensemble-based AdaBoost algorithms (Table 10), CatBoost demonstrates clear superiority across all performance metrics. Specifically, CatBoost achieved an accuracy, precision, recall, and F1-score of 96%, outperforming Decision Tree and AdaBoost, which both recorded 94% and 91%, respectively, on these metrics. In terms of discriminative capability, CatBoost attained an AUC of 99%, surpassing Decision Tree at 98% and AdaBoost at 97%. This consistent advantage highlights CatBoost's robustness in handling categorical data and moderate complexity with minimal preprocessing, reinforcing its effectiveness and compatibility with the dataset employed in this study.

Table 10. Performance comparison of AdaBoost, CatBoost, and Decision Tree classifiers.

Evaluation	CatBoost	Decision Tree	AdaBoost
Accuracy	96%	94%	91%
Precision	96%	94%	91%
Recall	96%	94%	91%

Corresponding author: Kartika Maulida Hindrayani, kartika.maulida.ds@upnjatim.ac.id, Department of Data Science, Universitas Pembangunan Nasional Veteran Jawa Timur, Jl. Raya Rungkut Madya Gunung Anyar, Surabaya, 60294, Indonesia.
DOI: https://doi.org/10.35882/ijeeemi.v7i2.92

F-1 Score	96%	94%	91%
AUC	99%	98%	97%

B. Similar Research

When compared to other studies on health summarized in Table 9, the results of this research show superior performance in various key metrics, although the random forest ensemble model provides better accuracy (98.8%) [50]. Based on a recent study, the decision tree approach [51] applied managed to achieve an accuracy rate of 92.72%. According to the findings by Falcao et al [52], the XGBoost algorithm attained an accuracy of 84.6%, demonstrating its solid performance in classifying drug resistance ases. Meanwhile, another study reveals that the Adaptive Boosting (AdaBoost) algorithm achieved a high accuracy rate of 93.42% in the early identification of tuberculosis [53]. Although these studies show strong results, the use of ensemble learning in the form of CatBoost and the more advanced Optuna Optimization in this study consistently resulted in higher accuracy and improved predictive performance. Table 11 confirms the effectiveness of the combination of CatBoost and Optuna optimization in addressing prediction challenges in healthcare.

Table 11. Comparison of similar research studies related to TB classification ensemble and optimization techniques.

Study	Method	Acc (%)
This Study	CatBoost + Optuna	96
Rochman et al [50]	Random Forest	98.8
Fayaz et al [51]	Decision Tree	92.72
Falcao et al [52]	XGBoost	84.6
Karmani et al [53]	AdaBoost	93.42

C. Limitations and Future Work

The integration of CatBoost with Optuna for hyperparameter optimization significantly enhanced model performance, but several methodological limitations warrant attention. The study focused solely on CatBoost, limiting the ability to compare performance across other algorithms. Additionally, although Optuna's adaptive nature is statistically efficient, it requires considerable computational time, especially when expanding the iteration count and hyperparameter search space. The integration of Optuna into the modeling workflow also demands considerable programming effort, particularly in designing stable and effective objective functions. Other concerns include the potential for the optimization process to converge on local optima, as well as the model's sensitivity to synthetically processed data (e.g., SMOTE), which could hinder generalization across different datasets. Moreover, the observed frequent misclassification of the class regimen is likely due to overlapping feature patterns and ambiguous naming structures, which confuse the model. This issue could be mitigated by further refining feature distinctions and incorporating more relevant domain variables. Lastly, the risk of bias like stemming from imbalanced class distributions, overrepresented feature patterns, or preprocessing techniques can undermine model fairness and generalizability. To address these challenges, a combination of cross-validation, feature importance analysis, and continuous performance monitoring across subgroups is essential for building a more robust and equitable model.

Despite some methodological limitations, the optimized CatBoost-Optuna model demonstrates significant potential in enhancing tuberculosis (TB) treatment regimens. The optimized CatBoost-Optuna model, with a remarkable accuracy of 96%, has substantial potential for improving tuberculosis (TB) treatment regimens. By providing personalized drug recommendations based on patient-specific factors, the model enhances therapeutic efficacy and helps reduce the risk of multi-drug resistance (MDR). It offers a practical application in clinical settings, where it can be integrated into electronic health record systems to assist healthcare providers in selecting appropriate treatments, especially in high-burden areas. The model's high precision, recall, and AUC ensure accurate and reliable recommendations, reducing treatment failures and relapse rates while supporting efficient resource allocation. This is particularly beneficial in optimizing the Directly Observed Treatment Short-course (DOTS) program and improving overall patient outcomes. Despite challenges in distinguishing between certain regimens, the model demonstrates significant improvement in managing complex cases compared to traditional methods. For future work, expanding the model to include a wider range of treatment regimens, incorporating additional clinical variables, and testing its generalizability across diverse populations could further enhance its utility. Additionally, ongoing monitoring of model performance in real-world clinical settings will be crucial to ensure its long-term efficacy and adaptability in TB management.

5. CONCLUSION

This study evaluates the use of a popular machine learning algorithm, CatBoost, to assess its ability to classify the appropriate drug types for tuberculosis treatment, in order to prevent drug resistance. Prior to modeling, the dataset was balanced using SMOTE, and hyperparameter optimization was performed through the Optuna method with the TPE (Tree-structured Parzen Estimator) sampler. The results of this experiment show that combination of CatBoost with Optuna proved to be the most effective approach, with near-perfect achievement on key metrics such as F1-score, accuracy, recall, and precision of 96% and AUC of 99%, demonstrating its ability to significantly improve the performance of classification models.

This finding not only demonstrates the importance of

Corresponding author: Kartika Maulida Hindrayani, kartika.maulida.ds@upnjatim.ac.id, Department of Data Science, Universitas Pembangunan Nasional Veteran Jawa Timur, Jl. Raya Rungkut Madya Gunung Anyar, Surabaya, 60294, Indonesia.
DOI: https://doi.org/10.35882/ijeeemi.v7i2.92

oversampling techniques in overcoming class imbalance as well as the effectiveness of hyperparameter optimization, but also emphasizes the practical value of the developed model. The high accuracy achieved opens up great opportunities in supporting clinical decision-making, especially in selecting the most appropriate anti-tuberculosis drug therapy. Thus, this approach has the potential to assist medical personnel in developing more targeted treatment regimens, thereby minimizing the risk of multi-drug-resistant tuberculosis (MDR-TB). In the future, further development can be directed towards the application of feature selection methods to improve feature efficiency, exploration of other ensemble learning techniques to strengthen model stability, and strategies that can handle data limitations without degrading prediction performance.

REFERENCES

[1] E. H. Tobin and D. Tristram, "Tuberculosis Overview," in *StatPearls*, Treasure Island (FL): StatPearls Publishing, 2025. Accessed: Apr. 10, 2025. [Online]. Available:http://www.ncbi.nlm.nih.gov/books/NBK441916/

[2] B. Utomo *et al.*, "Comparison Epidemiology between Tuberculosis and COVID-19 in East Java Province, Indonesia: An Analysis of Regional Surveillance Data in 2020," *Trop. Med. Infect. Dis.*, vol. 7, no. 6, Art. no. 6, Jun. 2022, doi: 10.3390/tropicalmed7060083.

[3] Y. Kang *et al.*, "Treatment Outcomes of Patients with Multidrug-Resistant Tuberculosis: Comparison of Pre- and Post-Public–Private Mix Periods," *Tuberc. Respir. Dis.*, vol. 84, no. 1, pp. 74–83, Jan. 2021, doi: 10.4046/trd.2020.0093.

[4] A. A. Khan, O. Chaudhari, and R. Chandra, "A review of ensemble learning and data augmentation models for class imbalanced problems: Combination, implementation and evaluation," *Expert Syst. Appl.*, vol. 244, p. 122778, Jun. 2024, doi: 10.1016/j.eswa.2023.122778.

[5] M. Idhom, D. A. Prasetya, P. A. Riyantoko, T. M. Fahrudin, and A. P. Sari, "Pneumonia Classification Utilizing VGG-16 Architecture and Convolutional Neural Network Algorithm for Imbalanced Datasets," vol. 4, no. 1.

[6] A. I. Lavrova and E. B. Postnikov, "An Improved Diagnostic of the Mycobacterium tuberculosis Drug Resistance Status by Applying a Decision Tree to Probabilities Assigned by the CatBoost Multiclassifier of Matrix Metalloproteinases Biomarkers," *Diagnostics*, vol. 12, no. 11, p. 2847, Nov. 2022, doi: 10.3390/diagnostics12112847.

[7] L.-H. Lai *et al.*, "The Use of Machine Learning Models with Optuna in Disease Prediction," *Electronics*, vol. 13, no. 23, Art. no. 23, Jan. 2024, doi: 10.3390/electronics13234775.

[8] S. D. Kurniawan, P. Purwono, A. Ma'Arif, and I. Suwarno, "Diabetes Classification Problem with CatBoost Method and Optuna Gradient Boosting Optimization," in *2023 6th International Conference on Information and Communications Technology (ICOIACT)*, Yogyakarta, Indonesia: IEEE, Nov. 2023, pp. 361–366. doi: 10.1109/ICOIACT59844.2023.10455940.

[9] X. Wang, W. Wang, H. Ren, X. Li, and Y. Wen, "Prediction and analysis of risk factors for diabetic retinopathy based on machine learning and interpretable models," *Heliyon*, vol. 10, no. 9, p. e29497, May 2024, doi: 10.1016/j.heliyon.2024.e29497.

[10] H. Koçak and G. ÇetiN, "The Diagnosis of Diabetes Mellitus with Boosting Methods," *El-Cezeri Fen Ve Mühendis. Derg.*, May 2023, doi: 10.31202/ecjse.1242207.

[11] J. T. Hancock and T. M. Khoshgoftaar, "CatBoost for big data: an interdisciplinary review," *J. Big Data*, vol. 7, no. 1, p. 94, Nov. 2020, doi: 10.1186/s40537-020-00369-8.

[12] Y. Liu, S. Wu, Z. Wu, and S. Zhou, "Application of gradient boosting machine in satellite-derived bathymetry using Sentinel-2 data for accurate water depth estimation in coastal environments," *J. Sea Res.*, vol. 201, p. 102538, Oct. 2024, doi: 10.1016/j.seares.2024.102538.

[13] P. Srinivas and R. Katarya, "hyOPTXg: OPTUNA hyper-parameter optimization framework for predicting cardiovascular disease using XGBoost," *Biomed. Signal Process. Control*, vol. 73, p. 103456, Mar. 2022, doi: 10.1016/j.bspc.2021.103456.

[14] S. Arora, S. R. Thota, and S. Gupta, "Artificial Intelligence-Driven Big Data Analytics for Business Intelligence in SaaS Products," in *2024 First International Conference on Pioneering Developments in Computer Science & Digital Technologies (IC2SDT)*, Aug. 2024, pp. 164–169. doi: 10.1109/IC2SDT62152.2024.10696409.

[15] K. M. Hindrayani and J. Timur, "Business Intelligence For Educational Institution : A Literature Review," vol. 2, no. 1, 2020.

[16] J. C. Johnston, Cooper ,Ryan, and D. and Menzies, "Chapter 5: Treatment of tuberculosis disease," *Can. J. Respir. Crit. Care Sleep Med.*, vol. 6, no. sup1, pp. 66–76, Mar. 2022, doi: 10.1080/24745332.2022.2036504.

[17] A. Mumuni and F. Mumuni, "Automated data processing and feature engineering for deep learning and big data applications: A survey," *J. Inf. Intell.*, vol. 3, no. 2, pp. 113–153, Mar. 2025, doi: 10.1016/j.jiixd.2024.01.002.

[18] T. Emmanuel, T. Maupong, D. Mpoeleng, T. Semong, B. Mphago, and O. Tabona, "A survey on missing data in machine learning," *J. Big Data*, vol. 8, no. 1, p. 140, Oct. 2021, doi: 10.1186/s40537-021-00516-9.

[19] W. Zhu, R. Qiu, and Y. Fu, "Comparative Study on the Performance of Categorical Variable Encoders in Classification and Regression Tasks," Jan. 18, 2024, *arXiv*: arXiv:2401.09682. doi: 10.48550/arXiv.2401.09682.

[20] M. Sivakumar, S. Parthasarathy, and T. Padmapriya, "Trade-off between training and testing ratio in

Corresponding author: Kartika Maulida Hindrayani, kartika.maulida.ds@upnjatim.ac.id, Department of Data Science, Universitas Pembangunan Nasional Veteran Jawa Timur, Jl. Raya Rungkut Madya Gunung Anyar, Surabaya, 60294, Indonesia.
DOI: https://doi.org/10.35882/ijeeemi.v7i2.92

machine learning for medical image processing," *PeerJ Comput. Sci.*, vol. 10, p. e2245, Sep. 2024, doi: 10.7717/peerj-cs.2245.
[21] V. R. Joseph, "Optimal Ratio for Data Splitting," *Stat. Anal. Data Min. ASA Data Sci. J.*, vol. 15, no. 4, pp. 531–538, Aug. 2022, doi: 10.1002/sam.11583.
[22] O. A. M. López, A. M. López, and D. J. Crossa, "Overfitting, Model Tuning, and Evaluation of Prediction Performance," in *Multivariate Statistical Machine Learning Methods for Genomic Prediction [Internet]*, Springer, 2022. doi: 10.1007/978-3-030-89010-0_4.
[23] A. X. Wang, S. S. Chukova, and B. P. Nguyen, "Synthetic minority oversampling using edited displacement-based k-nearest neighbors," *Appl. Soft Comput.*, vol. 148, p. 110895, Nov. 2023, doi: 10.1016/j.asoc.2023.110895.
[24] A. Zhang, H. Yu, Z. Huan, X. Yang, S. Zheng, and S. Gao, "SMOTE-R*k*NN: A hybrid re-sampling method based on SMOTE and reverse *k*-nearest neighbors," *Inf. Sci.*, vol. 595, pp. 70–88, May 2022, doi: 10.1016/j.ins.2022.02.038.
[25] D. A. Prasetya, P. T. Nguyen, R. Faizullin, I. Iswanto, and F. Armay, "Resolving the Shortest Path Problem using the Haversine Algorithm".
[26] D. R. I. M. Setiadi, K. Nugroho, A. R. Muslikh, S. W. Irianandа, and A. A. Ojugo, "Integrating SMOTE-Tomek and Fusion Learning with XGBoost Meta-Learner for Robust Diabetes Recognition," *J. Future Artif. Intell. Technol.*, vol. 1, no. 1, pp. 23–38, May 2024, doi: 10.62411/faith.2024-11.
[27] S. Emami and G. Martínez-Muñoz, "Condensed-gradient boosting," *Int. J. Mach. Learn. Cybern.*, vol. 16, no. 1, pp. 687–701, Jan. 2025, doi: 10.1007/s13042-024-02279-0.
[28] Y. F. Zamzam, T. H. Saragih, R. Herteno, Muliadi, D. T. Nugrahadi, and P.-H. Huynh, "Comparison of CatBoost and Random Forest Methods for Lung Cancer Classification using Hyperparameter Tuning Bayesian Optimization-based," *J. Electron. Electromed. Eng. Med. Inform.*, vol. 6, no. 2, pp. 125–136, Mar. 2024, doi: 10.35882/jeeemi.v6i2.382.
[29] A. Hadianto and W. H. Utomo, "CatBoost Optimization Using Recursive Feature Elimination," *J. Online Inform.*, vol. 9, no. 2, Art. no. 2, Aug. 2024, doi: 10.15575/join.v9i2.1324.
[30] J. Kooistra, "Applications of eXplainable Artificial Intelligence in Public Employment Services Decision Support Systems".
[31] R. Chen *et al.*, "Islanding detection method for microgrids based on CatBoost," *Front. Energy Res.*, vol. 10, Jan. 2023, doi: 10.3389/fenrg.2022.1016754.
[32] P. A. Riyantoko, T. M. Fahrudin, K. M. Hindrayani, and M. Idhom, "Exploratory Data Analysis and Machine Learning Algorithms to Classifying Stroke Disease," *IJCONSIST J.*, vol. 2, no. 02, pp. 77–82, Jun. 2021, doi: 10.33005/ijconsist.v2i02.49.
[33] L. Prokhorenkova, G. Gusev, A. Vorobev, A. V. Dorogush, and A. Gulin, "CatBoost: unbiased boosting with categorical features".
[34] Q. Jing, "Enhanced Decision Model for Optimize Operational Performance and Cost Efficiency Under Disc Interval-Valued Fermatean Fuzzy Acknowledge," *IEEE Access*, vol. 12, pp. 194423–194435, 2024, doi: 10.1109/ACCESS.2024.3520223.
[35] D. A. Prasetya, A. P. Sari, M. Idhom, and A. Lisanthoni, "Optimizing Clustering Analysis to Identify High-Potential Markets for Indonesian Tuber Exports," vol. 7, no. 1, 2025.
[36] S. Shekhar, A. Bansode, and A. Salim, "A Comparative study of Hyper-Parameter Optimization Tools," Jan. 17, 2022, *arXiv*: arXiv:2201.06433. doi: 10.48550/arXiv.2201.06433.
[37] D. Zegarra Rodríguez, O. Daniel Okey, S. S. Maidin, E. Umoren Udo, and J. H. Kleinschmidt, "Attentive transformer deep learning algorithm for intrusion detection on IoT systems using automatic Xplainable feature selection," *PloS One*, vol. 18, no. 10, p. e0286652, 2023, doi: 10.1371/journal.pone.0286652.
[38] Y. Zhou, Z. Dong, and X. Bao, "A Ship Trajectory Prediction Method Based on an Optuna–BILSTM Model," *Appl. Sci.*, vol. 14, no. 9, Art. no. 9, Jan. 2024, doi: 10.3390/app14093719.
[39] Y. Rimal, N. Sharma, and A. Alsadoon, "The accuracy of machine learning models relies on hyperparameter tuning: student result classification using random forest, randomized search, grid search, bayesian, genetic, and optuna algorithms," *Multimed. Tools Appl.*, vol. 83, no. 30, pp. 74349–74364, Sep. 2024, doi: 10.1007/s11042-024-18426-2.
[40] A. Bahrami, M. Rakhshaninejad, R. Ghousi, and A. Atashi, "Enhancing machine learning performance in cardiac surgery ICU: Hyperparameter optimization with metaheuristic algorithm," *PLOS ONE*, vol. 20, no. 2, p. e0311250, Feb. 2025, doi: 10.1371/journal.pone.0311250.
[41] S. Sieradzki and J. Mańdziuk, "Modified Adaptive Tree-Structured Parzen Estimator for Hyperparameter Optimization," Feb. 02, 2025, *arXiv*: arXiv:2502.00871. doi: 10.48550/arXiv.2502.00871.
[42] I. Scott, D. Cook, and E. Coiera, "Evidence-based medicine and machine learning: a partnership with a common purpose," *BMJ Evid.-Based Med.*, vol. 26, no. 6, pp. 290–294, Dec. 2021, doi: 10.1136/bmjebm-2020-111379.
[43] A. Muhaimin, W. Wibowo, and P. A. Riyantoko, "Multi-label Classification Using Vector Generalized Additive Model via Cross-Validation," *J. Inf. Commun. Technol.*, vol. 22, 2023, doi: 10.32890/jict2023.22.4.5.
[44] 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, Art. no. 4, Dec. 2021, doi: 10.3390/technologies9040081.
[45] M. Conciatori, A. Valletta, and A. Segalini, "Improving the quality evaluation process of machine learning algorithms applied to landslide time series analysis,"

Corresponding author: Kartika Maulida Hindrayani, kartika.maulida.ds@upnjatim.ac.id, Department of Data Science, Universitas Pembangunan Nasional Veteran Jawa Timur, Jl. Raya Rungkut Madya Gunung Anyar, Surabaya, 60294, Indonesia.
DOI: https://doi.org/10.35882/ijeeemi.v7i2.92

Comput. Geosci., vol. 184, p. 105531, Feb. 2024, doi: 10.1016/j.cageo.2024.105531.
[46] Ş. K. Çorbacıoğlu and G. Aksel, "Receiver operating characteristic curve analysis in diagnostic accuracy studies: A guide to interpreting the area under the curve value," *Turk. J. Emerg. Med.*, vol. 23, no. 4, pp. 195–198, Oct. 2023, doi: 10.4103/tjem.tjem_182_23.
[47] I. V. D. Srihith, A. D. Donald, T. A. S. Srinivas, G. Thippanna, and P. V. Lakshmi, "Numeric Metamorphosis: Converting Categorical Features with Python," *J. Adv. Res. Mob. Comput.*, vol. 5, no. 3, Art. no. 3, Sep. 2023.
[48] J. K. Sayyad, K. Attarde, and N. Saadouli, "Optimizing e-Commerce Supply Chains With Categorical Boosting: A Predictive Modeling Framework," *IEEE Access*, vol. 12, pp. 134549–134567, 2024, doi: 10.1109/ACCESS.2024.3447756.
[49] S. Kumar, U. Weesakul, D. R. Kumar, P. Thangavel, W. Wipulanusat, and J. Sunkpho, "A machine learning approach for corrosion rate modeling in Patna water distribution network of Bihar," *Sci. Rep.*, vol. 15, no. 1, p. 11678, Apr. 2025, doi: 10.1038/s41598-025-96044-0.
[50] E. M. S. Rochman, M. -, H. Suprajitno, I. Kamilah, A. Rachmad, and I. Santosa, "Tuberculosis classification using random forest with K-prototype as a method to overcome missing value," *Commun Math Biol Neurosci*, vol. 2023, no. 0, p. Article ID 81, Nov. 2023, doi: 10.28919/cmbn/7873.
[51] S. A. Fayaz *et al.*, "Machine learning algorithms to predict treatment success for patients with pulmonary tuberculosis," *PLOS ONE*, vol. 19, no. 10, p. e0309151, Oct. 2024, doi: 10.1371/journal.pone.0309151.
[52] I. W. S. Falcao *et al.*, "Model for predicting drug resistance based on the clinical profile of tuberculosis patients using machine learning techniques," *PeerJ Comput. Sci.*, vol. 10, p. e2246, Oct. 2024, doi: 10.7717/peerj-cs.2246.
[53] P. Karmani, A. A. Chandio, I. A. Korejo, O. W. Samuel, and M. Aborokbah, "Machine learning based tuberculosis (ML-TB) health predictor model: early TB health disease prediction with ML models for prevention in developing countries," *PeerJ Comput. Sci.*, vol. 10, p. e2397, Oct. 2024, doi: 10.7717/peerj-cs.2397.

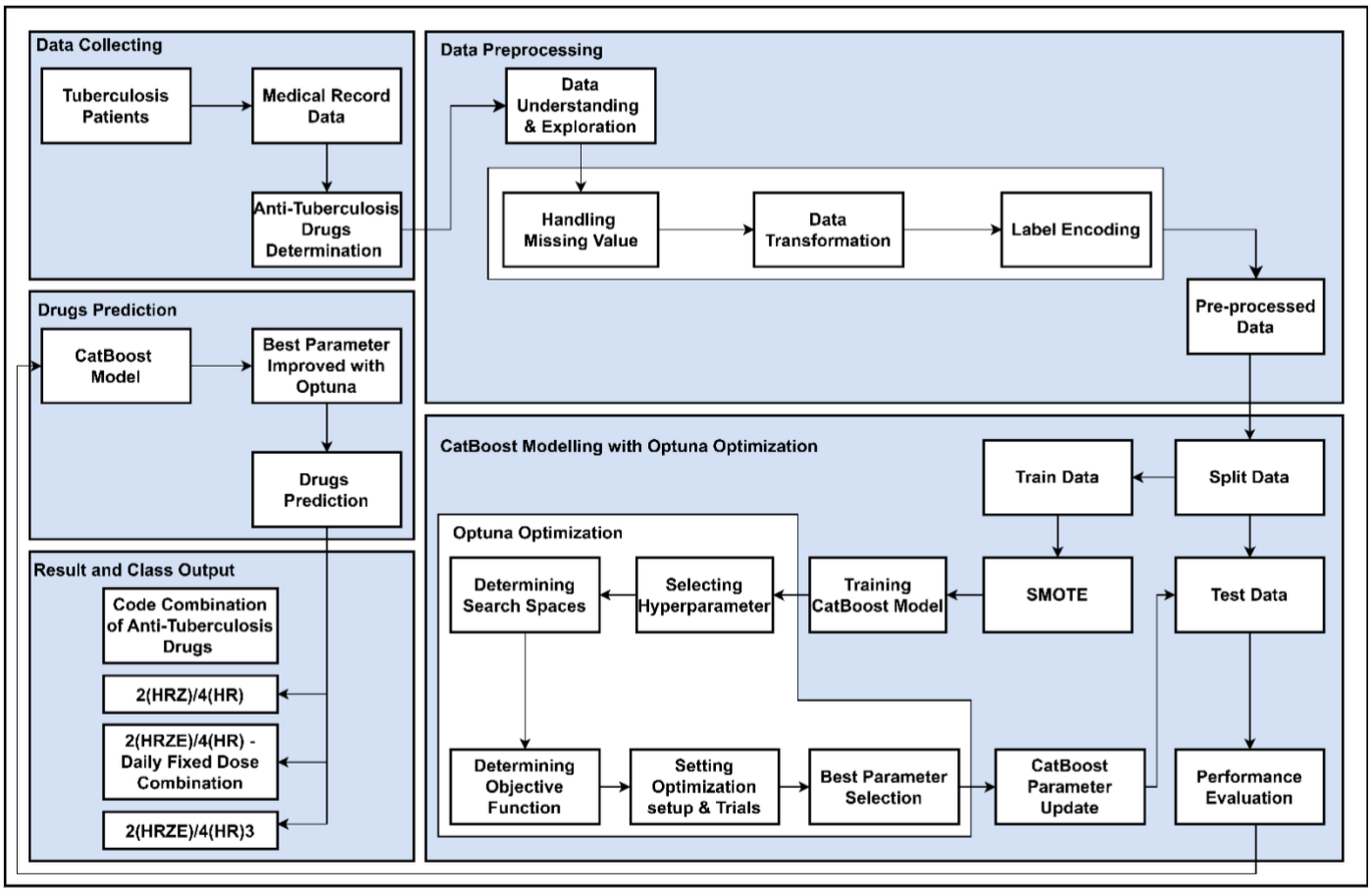
AUTHOR BIOGRAPHY

Yosua Satria Bara Harmoni is a final-year undergraduate student in the Data Science program at the Faculty of Computer Science, Universitas Pembangunan Nasional "Veteran" Jawa Timur, Surabaya. He has been actively pursuing his studies since 2021 and became a member of the Institute of Electrical and Electronics Engineers (IEEE) in 2024, demonstrating his commitment to advancing technology and innovation. His academic and research interests encompass various aspects of data science, including data analysis, machine learning, predictive modeling, data visualization, time series forecasting, and data-driven decision-making, particularly within the business and healthcare domains. Yosua is passionate about applying analytical techniques and innovative methodologies to solve real-world challenges through data-centric approaches.

Kartika Maulida Hindrayani is a Lecturer of Data Science at the University Pembangunan Nasional Veteran Jawa Timur. She earned her Bachelor's Degree and Master's Degree in Information Systems from Institut Teknologi Sepuluh Nopember (ITS) in 2015 and 2020 respectively. Her research focuses on the development of decision support system, data mining, operation research, and business intelligence. She holds a qualification for Associate Data Scientist from Indonesian Professional Certification Authority and a member of the Institute of Electrical and Electronics Engineers (IEEE).

Dwi Arman Prasetya earned his Bachelor's degree in Engineering from Institut Teknologi Sepuluh Nopember (ITS), Surabaya, Indonesia, in 2004. He completed his Master's degree in Engineering at Universitas Brawijaya, Malang, Indonesia, in 2010, and obtained his Doctor of Engineering degree from Tokushima University, Japan, in 2013. He currently serves as an Associate Professor in Data Science at Universitas Pembangunan Nasional Veteran Jawa Timur, Surabaya, Indonesia. His research interests span robotics, swarm robotics, artificial intelligence, virtual reality, and the internet of things. He is actively involved as a member of the Electrical Engineering Education Forum Indonesia (FORTEI Indonesia), a member of the Institute of Electrical and Electronics Engineers (IEEE), and also holds the position of Deputy Head of the Certification Department at The Institution of Engineers Indonesia.

Corresponding author: Kartika Maulida Hindrayani, kartika.maulida.ds@upnjatim.ac.id, Department of Data Science, Universitas Pembangunan Nasional Veteran Jawa Timur, Jl. Raya Rungkut Madya Gunung Anyar, Surabaya, 60294, Indonesia.
DOI: https://doi.org/10.35882/ijeeemi.v7i2.92



Corresponding author: Kartika Maulida Hindrayani, kartika.maulida.ds@upnjatim.ac.id, Department of Data Science, Universitas Pembangunan Nasional Veteran Jawa Timur, Jl. Raya Rungkut Madya Gunung Anyar, Surabaya, 60294, Indonesia.
DOI: https://doi.org/10.35882/ijeeemi.v7i2.92

Corresponding author: Kartika Maulida Hindrayani, kartika.maulida.ds@upnjatim.ac.id, Department of Data Science, Universitas Pembangunan Nasional Veteran Jawa Timur, Jl. Raya Rungkut Madya Gunung Anyar, Surabaya, 60294, Indonesia.
DOI: https://doi.org/10.35882/ijeeemi.v7i2.92