INTERNATIONAL JOURNAL OF ADVANCED COMPUTING AND INFORMATICS
VOLUME 2, ISSUE 1, 2026, pp. 1 – 11
ISSN: 3089-7483, DOI: https://doi.org/10.71129/ijaci.v2i1.pp1-11

RESEARCH ARTICLE

Stacked Ensemble of Gradient Boosting Machine, Categorical Boosting, and Extreme Gradient Boosting for Enhanced Tuberculosis Diagnosis

Amir Hamzah Dinnillah [a], **Muhammad Fuad Abdullah** [b]

[a]Faculty of Information Technology, Universitas Nusa Mandiri, 12540 Jakarta, Indonesia
[b]Faculty of Information and Communication Technology, Universiti Teknikal Malaysia Melaka, 76100, Melaka, Malaysia

ABSTRACT - Tuberculosis (TB), a deadly infectious disease caused by Mycobacterium tuberculosis (MTB), remains one of the leading causes of death worldwide, particularly in low and middle-income countries. Despite over a century of eradication efforts, TB continues to pose a significant public health challenge. One of the main obstacles in controlling TB is the limitation of diagnostic facilities that can quickly and accurately detect the disease. This study aims to develop a predictive model for TB diagnosis using various machine learning (ML) algorithms, including Decision Tree, Neural Network, Gaussian Naive Bayes, Logistic Regression, AdaBoost, Categorical Boosting (CatBoost), Gradient Boosting Machine (GBM), and Extreme Gradient Boosting (XGBoost). To prepare an optimal dataset, the research also included data preprocessing using encoding, scaling, and correlation analysis techniques. The dataset comprised 1,200 questionnaires collected from three hospitals in Sindh, with 950 respondents meeting the inclusion criteria. Experimental results showed that the GBM model achieved the highest accuracy of 95.61%, followed by CatBoost at 94.74% and XGBoost at 94.30%. To further enhance accuracy, stacked ensemble method was applied by combining GBM, CatBoost, and XGBoost as a meta-model, resulting in a remarkable accuracy of 96.49%. These findings indicate that the proposed stacked ensemble method can significantly improve TB diagnosis, offering faster and more efficient solutions compared to traditional methods. This research has the potential to enhance early detection and treatment of TB, especially in areas with limited healthcare resources.

Keywords: Tuberculosis, Machine Learning, Predictive Model, Diagnosis, Stacked Ensemble

1. Introduction

Tuberculosis (TB), a deadly infectious disease caused by Mycobacterium tuberculosis (MTB), remains one of the primary global causes of mortality, particularly affecting low- and middle-income nations [1]. It predominantly targets the respiratory system, with the lungs being the primary site of infection, though it can spread to other organs as well [2]. Despite over a century of eradication efforts, TB continues to be a significant public health challenge. This persistence is driven by several factors, including MTB's ability to evade the immune system, compounded by social health disparities, limited medical personnel, and underdeveloped healthcare infrastructure [3]. Furthermore, the existence of latent tuberculosis infection (Latent Tuberculosis Infection - LTBI) exacerbates the difficulty in reducing the incidence of new cases.

The early detection of tuberculosis (TB) is essential to curbing its transmission and improving treatment outcomes [4]. Nevertheless, conventional diagnostic techniques, such as sputum cultures, imaging, and interferon-gamma (IFN-γ) assays, have their inherent limitations. For example, the interferon-gamma release assay (IGRA) struggles to differentiate TB from latent tuberculosis infection (LTBI) or pneumonia, while acid-fast bacillus (AFB) staining can detect Mycobacterium, but it cannot distinguish between non-Mtb and Mtb [5]. Although sputum culture is highly sensitive, it takes over two weeks to produce results, and chest X-rays are restricted by cost and available infrastructure. Moreover, non-adherence to TB treatment has raised the risk of developing drug-resistant strains, worsening the global health crisis [6].

In line with early TB detection efforts, previous research has proposed the utilization of machine learning (ML) algorithms for early TB detection, namely Decision Tree (DT), Gaussian Naive Bayes (GNB), Logistic Regression Classifier (LRC), Adaptive Boosting (AdaBoost), and Neural Network (NN). Experimental results from these studies demonstrated that these models performed well in TB diagnosis, with DT achieving 92.11%, GNB 89.04%, LRC 90.35%, AdaBoost 93.42%, and NN 92.98%. These outcomes indicate that ML algorithms can be highly effective for TB diagnosis, offering quicker and more efficient solutions compared to conventional diagnostic methods. However, despite prior research demonstrating the effectiveness of individual ML algorithms as an initial TB diagnostic tool and AdaBoost even showing the best performance in accuracy (93.42%), precision (95.76%), and F1 score (93.78%) a significant gap remains in exploring the synergistic potential of combining various gradient boosting techniques into a unified ensemble [7].

Received 08 April 2025, Accepted 15 June 2025
Available online 28 July 2025

* **Corresponding Author**
E-mail: 14240044@nusamandiri.ac.id (Amir Hamzah Dinnillah)
© The Author(s) 2026. This is an open-access article under the CC BY license (https://creativecommons.org/licenses/by/4.0/)

In this study proposes an innovative predictive model for TB diagnosis that aims to address gaps in previous studies, where the exploration of the synergistic potential of combining various gradient boosting techniques into a single integrated ensemble was limited, and comparative evaluation using consistent datasets was often lacking. The proposed model leverages a stacked ensemble method, synergistically integrating the strengths of high-performing gradient boosting algorithms is Categorical Boosting (CatBoost), Gradient Boosting Machine (GBM), and Extreme Gradient Boosting (XGBoost) as base learners. Model validation was conducted using a comprehensive dataset derived from 950 patient questionnaires collected from three hospitals in Sindh, encompassing various patient characteristics and TB relevant symptoms such as cough, chest pain, smoking habits, and TB risk exposure. To ensure optimal performance, careful hyperparameter optimization for each base model and the overall stacked ensemble was performed using the Optuna framework. The primary objective of this approach is to achieve highly accurate diagnostic capabilities, indicated by a significant improvement in performance metrics, thereby offering a faster and more efficient solution for TB detection compared to conventional diagnostic methods.

2. Related Works

Balogun et al. [8] aimed to predict TB treatment outcomes and identify associated risk factors using machine learning algorithms. They analyzed secondary data spanning 15 years from hospital medical records, including variables such as age, gender, and length of hospital stay. Five classification models were employed is Binary Logistic Regression (BLR), Discriminant Analysis (DA), Multilayer Perceptron (MLP), Radial Basis Function (RBF), and Decision Tree (DT). The results indicated that the MLP model performed the best, achieving an overall classification accuracy of 73.4%. The study concluded that age and length of hospital stay were significant risk factors and recommended MLP for future predictions. However, the study is limited by potential bias in the hospital data and the need for a larger dataset, as well as the inclusion of additional factors.

Smith et al. [9]aimed to predict the microbiological confirmation of Mycobacterium tuberculosis in young children (<5 years) using easily accessible clinical, demographic, and radiological factors. The researchers evaluated eleven machine learning models, including Random Forest and Support Vector Machine (SVM), trained on a prospective cohort data from Kenya. The models demonstrated high accuracy (AUROC ranging from 0.84 to 0.90 for invasive samples and 0.83 to 0.89 for non-invasive samples), with the Polynomial SVM model achieving the lowest misclassification rate of 0.14 for invasive cases. Key factors such as family TB contact history, immunological evidence of TB, and chest X-ray findings consistently showed significant influence. The study concluded that machine learning could accurately predict microbiological confirmation of TB. However, the study is limited by the broad categorization of variables and the lack of external validation.

Gichuhi et al. [10] aimed to identify individual risk factors for non-compliance with TB treatment in the Mukono district of Uganda using a machine learning approach. They analyzed retrospective data from 838 patients obtained from healthcare facility records and applied five classification algorithms, including Support Vector Machine (SVM), Random Forest (RF), and AdaBoost. SVM demonstrated the highest accuracy at 91.28%, although AdaBoost showed the best Area Under the Curve (AUC). The study identified factors such as TB type, GeneXpert results, age, and gender as predictors of non-compliance. The study concluded that machine learning models can accurately identify patients at risk of non-compliance. However, the research is limited by the use of healthcare facility registries, which may not capture all relevant socio-economic or environmental factors, and by the definition of non-compliance as a surrogate measure.

Kuang et al. [11] aimed to develop a rapid and accurate predictor for drug resistance in MTB using genomic sequencing data. The methodology involved training 24 binary classifiers using Logistic Regression, Random Forest, and a 1D Convolutional Neural Network (CNN) on 10,575 isolates. The results indicated that the 1D CNN model achieved the best F1-score, ranging from 81.1% to 98.2%, outperforming the rule-based Mykrobe tool. For isoniazid, the CNN model achieved an accuracy of 96.2%. The study concluded that ML methods can accurately predict TB drug resistance. Critically, the research is considered innovative for applying deep learning to a large cohort; however, the CNN model only slightly outperformed traditional ML methods and still requires hyperparameter optimization.

Shu et al. [12] aimed to develop a ML model capable of distinguishing between Crohn's Disease (CD) and intestinal tuberculosis (ITB). The methodology involved collecting clinical data from 241 patients across 51 parameters and testing six ML methods, with XGBoost exhibiting the best performance. The XGBoost model achieved a diagnostic accuracy of 0.884 and an Area Under the Receiver Operating Characteristic Curve (AUROC) of 0.946. In real clinical testing, the model demonstrated an accuracy of 0.860, with a strong agreement rate (90.7%) with the multidisciplinary team (MDT) assessment. The study concluded that the developed model is effective for differential diagnosis between ITB and CD. However, the primary limitations of this study include its retrospective nature, the use of data from a single center, a small validation sample size, and the exclusion of all relevant clinical data, such as CT results.

3. Material and Methods

In this study, the developed model for TB detection utilizes advanced machine learning techniques to enhance classification accuracy. The approach involves a comprehensive process of extracting key features from the dataset and performing data preprocessing, followed by the application of various machine learning algorithms as primary classifiers.

Furthermore, optimization of model parameters and selective feature selection are conducted to improve detection performance and minimize false positive rates. **Fig. 1** provides an overview of the proposed architecture, along with an analysis of the model's performance metrics.

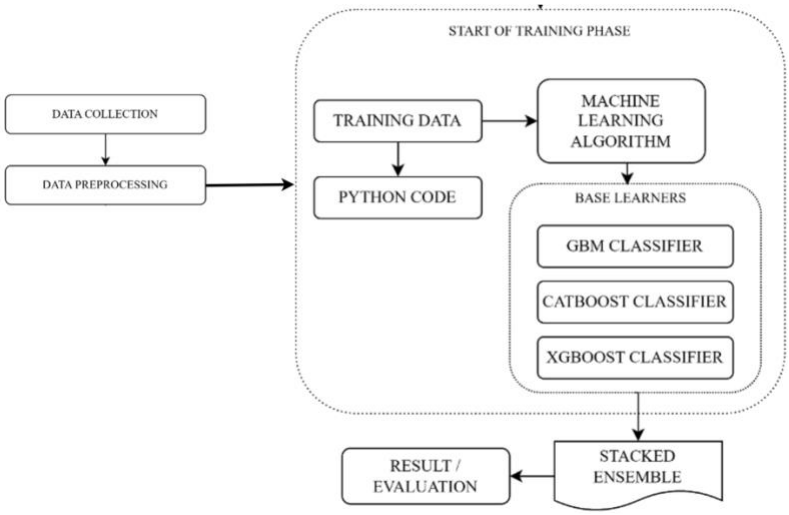

Fig. 1. Architecture of TB Predictor

3.1 Dataset

This study utilizes the same dataset as the prior research conducted by Karmani et al. (2024) [7], which involved 1,200 TB screening questionnaires randomly distributed across three hospitals in Sindh: the Institute of Chest Diseases (TB Sanatorium) Kotri, Liaquat University of Medical and Health Sciences (LUMHS) Jamshoro, and Civil Hospital Hyderabad. The questionnaire consisted of four sections: patient personal information, exploratory questions related to TB, supplementary information, and comments/feedback. The questionnaire was semi-structured, primarily comprising dichotomous (Yes/No) questions, supplemented by a few partially open-ended questions. In order to minimize bias, this study employed random probability sampling techniques, and the questionnaire underwent pre-testing to ensure its accuracy and psychometric reliability.

In contrast to the research conducted by Karmani et al., the present study applies alternative models and algorithms to analyze the same dataset. Of the 1,200 questionnaires distributed, 950 responses were received, yielding a response rate of 79.16%. After excluding incomplete data, 760 responses were selected for further analysis, as presented in **Table 1**. The overall response rate achieved was 63.3%. The collected data were subsequently analyzed using alternative machine learning (ML) algorithms, with the objective of developing a more efficient and accurate predictive model for TB diagnosis.

Table 1. Statistical summary of the survey responses

Hospitals	**Number of questionnaires**	
	Distributed	**Returned**
Institute of Chest Diseases (TB Sanatorium), Kotri	400	360
Liaquat University of Medical and Health Sciences, Jamshoro	400	280
Civil Hospital, Hyderabad	400	310
Aggregate	1,200	950
Ambiguous questionnaires	(-)190	
Final sample of the study	n = 760	

This dataset is utilized in data analysis for predicting TB, comprising data from 760 patients with 26 feature columns, which include both categorical and numerical variables. The dataset contains information regarding patient characteristics such as the presence of cough, cough duration and type, chest pain, breathing condition, body temperature, chills, and whether the patient experiences pulmonary effusion. Additional features include dietary habits, energy adequacy, physical condition, smoking habits, living density, and exposure to TB risk. A total of 373 records are available, with cough, chest pain, and breathing difficulties being the primary symptoms considered in this study for TB prediction. **Table 2** shows a sample of the dataset, while **Table 3** provides a detailed description of the features used in the analysis.

Table 2. Sample of dataset

Patient ID	**Cough**	**Cough Duration**	**Cough Type**	**…**	**Prediction**
1	No	0	No	…	Not suspected
2	No	0	No	…	Not suspected
3	No	0	No	…	Not suspected
4	No	0	No	…	Not suspected
5	No	0	No	…	Not suspected

Patient ID	Cough	Cough Duration	Cough Type	…	Prediction
…	…	…	…	…	…
756	Yes	3	Productive	…	TB suspected
757	Yes	3	Productive	…	TB suspected
758	Yes	3	Productive	…	TB suspected
759	Yes	3	Productive	…	TB suspected
760	Yes	3	Productive	…	TB suspected

Table 3. Description of dataset features

Features	Possible values
Cough	{0 = No, 1 = Yes}
Cough duration	{0 = No cough, 0.5 = Mild, 1 = Occasional, 1.5 = Intermittent, 2 = Moderate, 2.5 = Frequent, 3 = Persistent}
Cough type	{0 = No, 1 = Non-Productive, 2 = Productive}
Mucus	{0 = Bloody, 1 = Clear, 2 = No}
Chest pain	{0 = No, 1 = Yes}
Breathe state	{0 = Dyspnea, 1 = Normal}
Body temperature	{0 = High, 1 = Normal}
Chills	{0 = No, 1 = Yes}
Pulmonary effusion	{0 = No, 1 = Yes}
ESR value	{0 = Distributed, 1 = Normal}
Diet	{0 = Balanced, 1 = Malnutrition}
Physique	{0 = Healthy, 1 = Weight Loss}
Energy adequacy	{0 = Fatigue, 1 = Fit}
Smoking	{0 = No, 1 = Yes}
Crowding	{0 = No, 1 = Yes}
Exposed	{0 = No, 1 = Yes}

3.2 Pre-Proccessing

Prior to the implementation of machine learning algorithms, a comprehensive preprocessing phase was carried out to ensure the quality, consistency, and suitability of the dataset for modeling tasks. This step is essential in any data-driven study, particularly in medical diagnostics, where data integrity directly influences model performance and reliability. The preprocessing workflow includes several critical stages, such as verifying the presence of missing values, eliminating irrelevant attributes, encoding categorical variables into numerical form, and splitting the dataset into training and testing subsets. Each of these steps plays a pivotal role in preparing the data to be effectively utilized by the selected classification models. The first step in preprocessing is to remove irrelevant features *"*$Unnamed: 0$*"*column, which provided no relevant information, was consequently removed from the dataset. This column contained values deemed extraneous to the model, merely increasing dataset complexity without contributing to the classification process.

$$\text{Correlation}\,(X, y) \approx 0 \tag{2}$$

where X denotes the irrelevant feature (in this case, the $Unnamed: 0$ column), y is the target variable, and the near-zero correlation indicates that there is little to no linear relationship between the feature and the target, justifying its removal.

Next, categorical variables are converted into numerical representations, the majority of features within the dataset are inherently categorical variables, necessitating their conversion into a numerical representation for compatibility with machine learning algorithms. This encoding process was executed using the $atype(\text{"Category"}).cat.codes$ method, which transforms each distinct category into a corresponding numerical code. A notable exception applied to the target column (Prediction) for the XGBoost and Stacked Ensemble models, where encoding was performed independently. This distinction was crucial as these specific models require a more explicit categorical representation for their internal processing mechanisms.

$$X_{\text{encoded}} = f(X) = \text{LabelEncoder}(X) \tag{3}$$

where X represents the original categorical feature, $\text{LabelEncoder}(X)$ denotes the transformation function that assigns a unique numerical code to each category in X, and X_{encoded} is the resulting numerically encoded feature used for model training.

Finally, the data is split into training and testing sets using the $train\_test\_split$ function from the scikit-learn library, with 70% allocated for training and 30% for testing. The split uses the $random\_state = 1$ parameter to ensure consistent and reproducible results, and the $stratify = ydata$ parameter to maintain a balanced distribution of the target classes in both the training and testing datasets. Overall, these preprocessing stages aim to prepare the data in an optimal format for the modeling phase, maintain data integrity, and ensure that relevant variability is preserved.

3.3 GBM Classifier

GBM is an ensemble learning algorithm that iteratively combines multiple weak learners typically decision trees into a strong predictive model. In the context of medical classification tasks such as TB detection, GBM is well-regarded for its capability to model non-linear relationships and adapt to complex data structures. The algorithm operates by

sequentially minimizing a defined loss function, where each new model is trained to correct the residual errors of the previous ensemble using a gradient descent approach [13]. The optimization process for GBM can be represented by the following equations:

Initial model prediction (base learner):

$$F_0(\mathrm{x}) = \frac{\arg\min}{\gamma} \sum_{i=1}^{n} L(\mathrm{y}_{i,}\, \gamma) \tag{4}$$

where $F_0(\mathrm{x})$ represents the initial model prediction (also known as the base learner), γ is the constant prediction value to be optimized, $L(\mathrm{y}_{i,}\, \gamma)$ is the loss function measuring the error between the true label y_i and the constant prediction γ, and n is the total number of training samples.

Computation of pseudo-residuals at iteration m:

$$r_{im} = -\left[\frac{\partial \mathrm{L}\left(\mathrm{y}_{i,}\, F(x_i)\right)}{\partial\, F(x_i)}\right]_{F(x)=F_{m-1}(x)} \tag{5}$$

where r_{im} denotes the pseudo-residual for the i-th instance at iteration m, $\mathrm{L}\left(\mathrm{y}_{i,}\, F(x_i)\right)$ is the loss function comparing the true label $\mathrm{y}_{i,}$ with the model prediction $F(x_i)$, and $F_{m-1}(x)$ is the prediction from the previous iteration $(m-1)$. The partial derivative measures the gradient of the loss with respect to the model output, serving as the direction for the next model update.

Training weak learner $h_m(x)$ to fit residuals:

$$h_m(x) = \mathrm{fit}\,(x_i, r_{im}) \tag{6}$$

where $h_m(x)$ is the weak learner (typically a decision tree) trained at iteration m, x_i represents the input features, and r_{im} are the pseudo-residuals computed from the previous iteration. The weak learner is trained to approximate these residuals in order to correct the prediction errors made so far.

Model update at each iteration:

$$F_m(x) = F_{m-1}(x) + \eta \cdot h_m(x) \tag{7}$$

where $F_m(x)$ is the updated prediction after the m -th iteration, $F_{m-1}(x)$ is the prediction from the previous iteration, η is the learning rate that controls the contribution of the new weak learner, and $h_m(x)$ is the weak learner trained to fit the pseudo-residuals at iteration m.

Final prediction after M iterations:

$$F_M(x) = F_0(x) + \sum_{m=1}^{m} \eta \cdot h_m(x) \tag{8}$$

where $F_M(x)$ is the final prediction after M iterations, $F_0(x)$ is the initial model prediction (base learner), η is the learning rate that scales the contribution of each weak learner, and $h_m(x)$ represents the weak learner trained at iteration m. The summation aggregates the improvements made by all M weak learners.

In this study, the optimization of the GBM Classifier model was performed comprehensively using GridSearchCV to achieve optimal classification performance. This technique systematically searches through key hyperparameter combinations, including $n\_estimators$, which determines the number of trees in the ensemble, $learning\_rate$, which controls the size of the model's update, and max $\_depth$, which regulates the complexity of each decision tree. The choice of GridSearchCV is based on its ability to conduct an exhaustive search methodically, enabling the identification of the optimal configuration within the defined parameter space. This optimization process is crucial to ensure that the model achieves its maximum predictive capacity while maintaining generalization on new data.

The performance of the model is illustrated through a confusion matrix presented in **Table 4** is used to assess the classification performance. The confusion matrix is crucial as it clearly shows the counts of True Positives (TB cases correctly identified), False Positives (non-TB cases misclassified as TB), True Negatives (non-TB cases correctly identified), and False Negatives (TB cases not detected). This visualization provides a detailed understanding of the errors made by the model, which is essential for diagnostic analysis.

The application of GBM in this research is justified by its strong generalization ability and resilience to overfitting when properly configured. Furthermore, GBM offers flexibility in choosing among various loss functions, enabling better alignment with the specific characteristics of medical datasets. Therefore, GBM is an appropriate and effective choice to enhance the accuracy and reliability of automated TB classification using a machine learning framework.

3.4 CatBoost Classifier

CatBoost is a gradient boosting algorithm specifically designed to handle categorical variables efficiently without requiring extensive preprocessing or manual encoding. As an advanced ensemble method, CatBoost constructs a sequence of decision trees where each successive tree attempts to correct the prediction errors of the previous ensemble. This makes it particularly suitable for complex classification tasks such as TB diagnosis, where feature interactions and nonlinear relationships are common [14]. The training process of CatBoost relies on the minimization of a specified loss function using gradient descent. In classification tasks, CatBoost commonly employs Log Loss or Cross-Entropy Loss, expressed mathematically as:

$$\mathcal{L}(\mathrm{y}, \hat{P}) = -[\mathrm{y}\ \log(\hat{P}) + (1 - \mathrm{y})\log(1 - \hat{P})] \tag{9}$$

where $\mathrm{y} \in \{0,1\}$ is the true class label, and $\hat{P}$ is the predicted probability of the positive class. During training, CatBoost computes the gradient of this loss and builds trees that minimize the residuals in a stage-wise manner:

$$F_m(x) = F_{m-1}(x) + \eta \cdot h_m(x) \tag{10}$$

where $F_m(x)$ is the prediction at stage mmm m, η is the learning rate, and $h_m(x)$ is the decision tree trained on the residuals.

To achieve optimal performance, the CatBoost model in this study was optimized using $GridSearchCV$. This exhaustive search method systematically explores predefined hyperparameter combinations to identify the best settings. Key hyperparameters tuned include $iterations$, $depth$, $learning\_rate$, and $loss\_function$. This optimization process is crucial for maximizing the model's generalization ability and mitigating the risk of overfitting or underfitting.

The model's performance is visually presented a confusion matrix in Table 4 is used to provide of the classification performance. The confusion matrix is valuable as it clearly displays the number of True Positives (correctly detected TB cases), False Positives (non-TB cases incorrectly classified as TB), True Negatives (correctly detected non-TB cases), and False Negatives (undetected TB cases). This visualization offers deeper insights into the types of errors made by the model, which is crucial for diagnostic analysis.

CatBoost was selected due to its strengths in preventing overfitting via ordered boosting and its native support for categorical features, which are prevalent in questionnaire-based clinical datasets. The model's robustness, efficiency, and reduced need for extensive preprocessing make it highly appropriate for medical classification scenarios such as automated TB detection.

3.5 XGBoost Classifier

XGBoost is a scalable, regularized boosting technique that extends the principles of gradient boosting to deliver enhanced speed and performance. It is particularly effective in high-dimensional classification tasks, including medical diagnostics such as TB detection, due to its capability to handle sparse and structured data, implement automatic regularization, and optimize parallel computation [15]. XGBoost operates by minimizing a differentiable loss function through an additive learning approach. In each iteration, the model fits a new decision tree to the residuals of the previous ensemble's predictions. The objective function combines both the loss term and a regularization term to penalize model complexity:

$$\mathcal{L}^{(t)} = \sum_{i=1}^{n} l(\mathrm{y}_i, \hat{y}^{(t-1)} + f_t(x_i)) + \Omega(f_t) \tag{11}$$

where $\mathcal{L}^{(t)}$ is the objective function at iteration t, $t(\mathrm{y}_i, \hat{y}^{(t-1)} + f_t(x_i))$ is the loss function measuring the difference between the true label y_i, and the updated prediction, $\hat{y}^{(t-1)}$ is the ensemble's prediction at iteration $t-1$, $f_t(x_i)$ is the new function (typically a decision tree) added at iteration t, and $\Omega(f_t)$ is the regularization term penalizing the complexity of f_t to avoid overfitting. Regularization Term:

$$\Omega(f) = \gamma T + \frac{1}{2}\ \lambda \sum_{j=1}^{T} \mathrm{w}_j^2 \tag{12}$$

where $\Omega(f)$ is the regularization term used to penalize the complexity of the model, T is the number of leaves in the decision tree f, γ is the penalty applied to each leaf (controlling model complexity), λ is the L2 regularization coefficient, and $\mathrm{w_j}$ is the score (weight) assigned to the j -th leaf. This term helps control overfitting by discouraging overly complex trees.

To align with the model's requirement for numerical input, the class labels were encoded using `LabelEncoder`, and categorical features were transformed into integer codes. XGBoost's built-in handling of sparse inputs and robustness against overfitting achieved through regularization and early stopping mechanisms makes it a suitable candidate for TB classification tasks where both model accuracy and generalizability are critical.

The performance optimization of the XGBoost model was carried out using GridSearchCV. This hyperparameter search method systematically explores key hyperparameter combinations, including $n\_estimators$, $learning\_rate$, and max $\_depth$, to identify the optimal configuration. Model validation was performed using $StratifiedKFold$ with 5 folds. This cross-validation approach ensures that class proportions are maintained across each fold, which is crucial for potentially imbalanced datasets, such as in medical classification.

Performance of the model is displayed visually using a confusion matrix presented in Table 4 is utilized to assess the classification performance. The confusion matrix is essential as it provides a clear breakdown of True Positives (TB cases correctly identified), False Positives (non-TB cases incorrectly labeled as TB), True Negatives (non-TB cases correctly classified), and False Negatives (TB cases that were missed). This visualization helps to gain a deeper understanding of the model's error patterns, which is critical for diagnostic evaluation.

XGBoost inherent capability to handle sparse inputs and its robustness against overfitting, primarily achieved through its advanced regularization mechanisms and early stopping protocols, render it a highly suitable candidate for TB classification tasks. In this critical medical context, where both model accuracy and generalizability are paramount, XGBoost offers a powerful and effective framework for achieving reliable diagnostic outcomes.

Table 4. Confusion Matrix for Evaluating Classifiers Performance.

GBM Classifier	**Predicted Normal**	**Predicted Tuberculosis**
Actual Normal	105	5
Actual Tuberculosis	5	113
CatBoost Classifier	**Predicted Normal**	**Predicted Tuberculosis**
Actual Normal	104	6
Actual Tuberculosis	6	112
XGBoost Classifier	**Predicted Normal**	**Predicted Tuberculosis**
Actual Normal	104	6
Actual Tuberculosis	7	111

Table 5. Configuration of Hyperparameter for Classifiers Perfomance.

Classifier	**Hyperparameter**	**Testing Value**	**Method for Finding the Best Parameters**
GBM	n_estimators	800, 1000	GridSearchCV
	learning_rate	0.05, 0.1	
	max_depth	6, 8	
CatBoost	iterations	800, 1000	GridSearchCV
	depth	6, 8	
	learning_rate	0.05, 0.1	
	loss_function	Logloss, CrossEntropy	
XGBoost	n_estimators	800, 1000	GridSearchCV
	learning_rate	0.05, 0.1	
	max_depth	6, 8	

3.5 Stacked Ensemble Classifiers

To ensure robust model development, the CatBoost Classifier, GBM Classifier, and XGBoost Classifier were implemented using a rigorous training strategy involving Grid Search Cross-Validation to determine the optimal combination of hyperparameters, including the number of estimators $n\_estimators$ learning rate (η), and maximum tree depth ($max\_depthmax$) The cross-validation procedure employed StratifiedKFold to preserve class distribution within each fold, which is essential for classification tasks involving imbalanced data such as TB diagnosis.

In this study, a stacked ensemble model was employed as an advanced machine learning strategy to enhance classification performance by leveraging the complementary strengths of multiple base learners [16]. Specifically, the ensemble integrates three high-performing gradient boosting algorithms CatBoost Classifier, Gradient Boosting Classifier, and XGBoost Classifier as base models, while a Logistic Regression model functions as the meta-learner. These base models are trained independently using a consistent 5-fold $StratifiedKFold$ cross-validation to ensure robust performance across imbalanced class distributions, particularly relevant in medical classification tasks such as TB suspicion. Each base model takes the same input feature vector x and outputs a predicted probability or class label. The outputs from these models are concatenated into a new feature vector:

$$z = [h_1(x), h_2(x), h_3(x)] \tag{13}$$

where z is the new feature vector formed by concatenating the outputs of multiple base models $(e.g., h_1, h_2, \text{and } h_3)$, each representing a different model's prediction for the same input x. This vector serves as the input to a meta-learner in ensemble methods such as stacked. The meta-model, implemented as **Logistic Regression**, uses the output vector z as input and computes the final prediction:

$$f(X) = \sigma\,(w\,z + b) \tag{14}$$

Prior to model training, categorical variables were transformed into numerical representations to comply with the input requirements of gradient boosting algorithms. Additionally, class labels were encoded using LabelEncoder, given

that XGBoost requires integer-based targets. Hyperparameter optimization was conducted using the Optuna framework, which efficiently explores the hyperparameter space through a Tree-structured Parzen Estimator (TPE) sampler. The optimization objective was defined to maximize the F1-score of the minority class ("TB Suspected"), ensuring that both precision and recall were balanced in the final model. Final Stacked Model Function:

$$f(\mathrm{x}) = \sigma(w_1 \cdot h_1(\mathrm{x}) + w_2 \cdot h_2(\mathrm{x}) + w_3 \cdot h_3(\mathrm{x}) + \mathrm{b}) \quad (15)$$

where $f(\mathrm{x})$ is the final prediction of the stacked ensemble model, $h_1(\mathrm{x})$, $h_2(\mathrm{x})$, and $h_3(\mathrm{x})$ are the outputs from the base models, w_1, w_2, and w_3 are the corresponding weights learned by the meta-learner, b is the bias term, and σ is the activation function (commonly the sigmoid function) used to map the weighted sum into a probability or final decision value.

After identifying the optimal set of hyperparameters, the best-performing models were retrained and integrated into a final stacked classifier. Performance evaluation was conducted using standard classification metrics (accuracy, precision, recall, F1-score), and the results were visualized using interactive bar plots and a stylized confusion matrix. The critical advantage of the stacked approach lies in its ability to capture diverse decision boundaries from heterogeneous models, thereby improving generalization and reducing overfitting. This multi-layered architecture proves particularly effective for complex and high-stakes classification problems in the medical domain, where sensitivity and specificity must be carefully balanced.

4. Results and Discussion

To rigorously assess the classification performance of the proposed model in distinguishing between Normal and Tuberculosis cases, a confusion matrix analysis was performed. This method enables a granular evaluation of prediction outcomes by identifying not only the correctly classified instances but also the types of miss classification made by the model. Specifically, the Normal class represents individuals without tuberculosis, while the Tuberculosis class denotes confirmed TB cases.

Table 6. Confusion matrix purposed method

Confusion Matrix	**Predicted Normal**	**Predicted Tuberculosis**
Actual Normal	105	5
Actual Tuberculosis	3	115

Table 6 presents the confusion matrix summarizing the model's classification performance. The model successfully identified 105 out of 110 actual *Normal* cases, with only 5 misclassified as *Tuberculosis*. Similarly, it correctly classified 115 out of 118 actual *Tuberculosis cases*, with just 3 labeled incorrectly as *Normal.* These results suggest that the model is not only capable of detecting tuberculosis cases with a high degree of sensitivity but also maintains good specificity by minimizing false alarms in normal patients. The low number of false negatives is particularly important in medical diagnostics, as failing to detect tuberculosis could have serious health implications. Overall, the confusion matrix indicates that the model achieves a well-balanced performance in distinguishing between the two classes.

Table 7. Comparison of Performance Metrics Across All Models Classifiers.

Model	Accuracy	Precision	Recall	F1-Score
Stacked Ensemble	**96.49 %**	**95.83 %**	**97.46 %**	**96.64 %**
GBM	95.61 %	95.76 %	95.76 %	95.76 %
CatBoost	94.74 %	94.92 %	94.92 %	94.92 %
XGBoost	94.30 %	94.87 %	94.07 %	94.47 %
Decision Tree	92.54 %	93.91 %	91.53 %	92.70 %
AdaBoost	90.79 %	91.45 %	90.68 %	91.06 %
Neural Network	90.79 %	91.45 %	90.68 %	91.06 %
Gaussian Naive Bayes	89.91 %	86.82 %	94.92 %	90.69 %
Logistic Regression	88.60 %	87.10 %	91.53 %	89.26 %

Based on **Table 7**, the proposed Stacked Ensemble method clearly demonstrates superior performance compared to all other evaluated models. It achieves the highest scores in all key metrics 96.49% accuracy**,** 95.83% precision**,** 97.46% recall**,** and 96.64% F1-score surpassing strong individual models such as GBM, CatBoost, and XGBoost. The combination of high recall and high precision reflects not only the model's ability to accurately identify tuberculosis cases but also its capacity to minimize false alarms, which is essential in clinical decision-making. The improvement in performance can be attributed to the nature of Stacked, which integrates multiple base learners to capture diverse patterns and reduce generalization error. Unlike single models that may be prone to overfitting or underfitting depending on the data distribution, the Stacked Ensemble benefits from the complementary strengths of its components, resulting in a more robust and generalizable classifier. This highlights its potential as a reliable tool in tuberculosis detection tasks.

To evaluate the performance of the stacked ensemble model in differentiating between 'TB Suspected' and 'Not Suspected' classes, we utilized the Receiver Operating Characteristic (ROC) curve and the Area Under the Curve (AUC). The calculated ROC-AUC score was 0.98, indicating the model's excellent discriminative ability between the two classes. The generated ROC curve demonstrated a remarkably high True Positive Rate (TPR) across nearly the entire range of

False Positive Rate (FPR). This signifies that the model can identify 'TB Suspected' cases with very few false positives. This graph illustrates near-perfect performance, which is critically important in the context of early TB detection, where false negatives must be minimized to prevent further disease transmission. The following ROC Curve illustrates an ROC-AUC of 0.98, reflecting the model's excellent prediction quality, with the area under the curve being near-perfect.

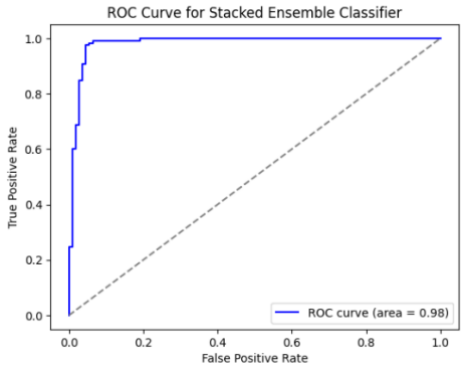

Fig. 2. ROC Curve for stacked ensemble classifier

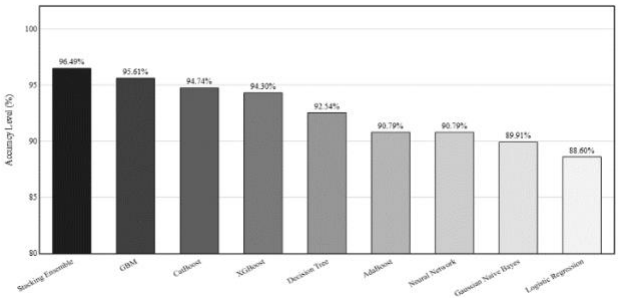

Fig. 3. Comparative analysis of accuracy levels of various classifier

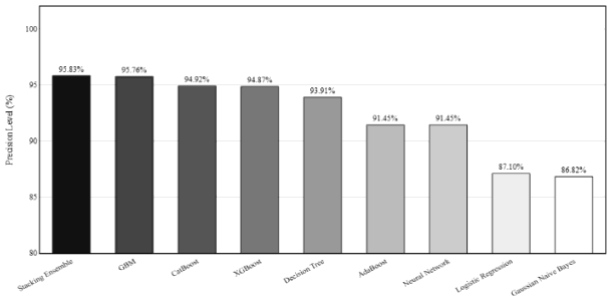

Fig. 4. Comparative analysis of precision levels of various classifier

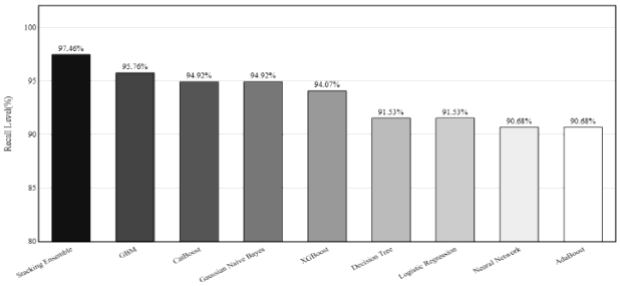

Fig. 5. Comparative analysis of recall Rates of various classifier

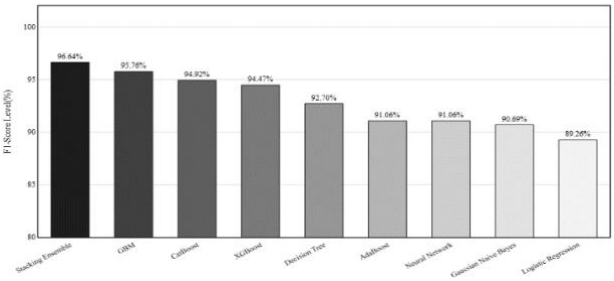

Fig. 6. Comparative analysis of f1-score level of various classifier

When compared to the findings by [7], who reported the best performance using the AdaBoost method with an accuracy of 93.42%, precision of 95.76%, and F1-score of 93.78%, the results of our study demonstrate a clear and consistent improvement across all evaluation metrics. Utilizing the same dataset, the proposed Stacked Ensemble model achieves an accuracy of 96.49% as depicted in **Fig. 3**, precision of 95.83% as depicted in **Fig. 4**, recall of 97.46% as depicted in **Fig. 5**, and an F1-score of 96.64% as depicted in **Fig. 6**. These improvements, although in some cases numerically modest, are statistically and practically significant in medical classification tasks where even small gains in recall or F1-score can translate into critical improvements in diagnostic reliability. Notably, while Karmani et al.'s model focused on boosting a single weak learner, our approach integrates multiple base classifiers in a stacked architecture, enabling the model to capture more complex patterns and reduce the generalization error. The Stacked Ensemble not only outperforms AdaBoost in accuracy and F1-score but also demonstrates superior recall, indicating better sensitivity in identifying tuberculosis cases.

While the proposed Stacked Ensemble model delivers superior predictive performance, one important consideration is the interpretability of the model. Unlike single decision trees, which are inherently transparent, ensemble models such as stacked especially those involving gradient boosting methods tend to function as "black boxes." This may pose challenges in clinical adoption, where understanding the rationale behind a prediction is critical. However, interpretability tools such as SHAP (SHapley Additive exPlanations) or feature importance analysis can be employed to provide post-hoc explanations, offering insights into which features most strongly influence the model's decisions.

Integrating such interpretability frameworks in future work would enhance the model's usability and trustworthiness in medical settings.

5. Conclusions

This study presents a robust machine learning-based framework for the early and accurate diagnosis of Tuberculosis (TB), utilizing a stacked ensemble approach that combines the predictive strengths of CatBoost, GBM, and XGBoost. By leveraging a clinically relevant dataset previously used by Karmani et al., and applying advanced data preprocessing, feature engineering, and hyperparameter optimization techniques, the proposed model achieved substantial improvements in diagnostic performance across all key evaluation metrics. The Stacked Ensemble model attained an accuracy of 96.49%, precision of 95.83%, recall of 97.46%, and an F1-score of 96.64%, outperforming all individual baseline models as well as prior research benchmarks.

In contrast to previous studies that often relied on standalone classifiers or traditional ensemble techniques such as AdaBoost, this research highlights the efficacy of a stacked architecture in capturing complex, nonlinear patterns in TB-related clinical data. The integration of diverse gradient boosting models enabled the system to generalize better and reduce the risk of overfitting, which is particularly crucial in medical applications where sensitivity and specificity are critical. Moreover, the model demonstrated a well-balanced confusion matrix with a minimal number of false negatives and false positives, reinforcing its practical utility in real-world TB screening contexts.While the predictive performance of the proposed model is promising, interpretability remains a key consideration for clinical adoption. Stacked ensemble, especially those based on gradient boosting, are inherently opaque, which may hinder trust and transparency in decision-making processes. Future work should focus on integrating interpretability frameworks such as SHAP or LIME to provide actionable insights into the model's reasoning. Doing so would enhance its acceptance by healthcare professionals and facilitate its implementation in TB diagnostic workflows, particularly in resource-constrained settings.

In summary, this research highlights the potential of ensemble machine learning methods, particularly Stacked models, to significantly enhance the accuracy of tuberculosis (TB) diagnosis. It contributes to the growing body of evidence supporting the use of AI-driven tools in public health and clinical diagnostics, providing a replicable and scalable solution for improving TB detection using accessible questionnaire-based data. The key findings demonstrate that this approach can improve diagnostic accuracy by leveraging readily available data. However, the study has some limitations, such as the need for larger and more diverse datasets to further validate the model. Future research could expand this work by testing the model across different populations and clinical settings, as well as optimizing it for real-world applications.

Conflicts of Interest

The author declares no conflict of interest.

References

[1] N. F. Khabibullina, D. M. Kutuzova, I. A. Burmistrova, and I. V. Lyadova, 'The Biological and Clinical Aspects of a Latent Tuberculosis Infection', *TropicalMed*, vol. 7, no. 3, p. 48, Mar. 2022, doi: 10.3390/tropicalmed7030048.

[2] A. M. Olmo-Fontánez and J. Turner, 'Tuberculosis in an Aging World', *Pathogens*, vol. 11, no. 10, p. 1101, Sep. 2022, doi: 10.3390/pathogens11101101.

[3] R. A. Salama and N. A. Rizk, 'Tuberculosis Elimination: Implications and Challenges', *Natl J Community Med*, vol. 14, no. 09, pp. 610–617, Sep. 2023, doi: 10.55489/njcm.140920233127.

[4] B. Dong, Z. He, Y. Li, X. Xu, C. Wang, and J. Zeng, 'Improved Conventional and New Approaches in the Diagnosis of Tuberculosis', *Front. Microbiol.*, vol. 13, p. 924410, May 2022, doi: 10.3389/fmicb.2022.924410.

[5] L.-S. Li, L. Yang, L. Zhuang, Z.-Y. Ye, W.-G. Zhao, and W.-P. Gong, 'From immunology to artificial intelligence: revolutionizing latent tuberculosis infection diagnosis with machine learning', *Military Med Res*, vol. 10, no. 1, p. 58, Nov. 2023, doi: 10.1186/s40779-023-00490-8.

[6] K. Naidoo, R. Perumal, S. L. Ngema, L. Shunmugam, and A. M. Somboro, 'Rapid Diagnosis of Drug-Resistant Tuberculosis–Opportunities and Challenges', *Pathogens*, vol. 13, no. 1, p. 27, Dec. 2023, doi: 10.3390/pathogens13010027.

[7] 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 Computer Science*, vol. 10, p. e2397, Oct. 2024, doi: 10.7717/peerj-cs.2397.

[8] O. S. Balogun, 'Investigating Machine Learning Methods for Tuberculosis Risk Factors Prediction – A Comparative Analysis and Evaluation'.

[9] J. P. Smith *et al.*, 'Machine learning to predict bacteriologic confirmation of Mycobacterium tuberculosis in infants and very young children', *PLOS Digit Health*, vol. 2, no. 5, p. e0000249, May 2023, doi: 10.1371/journal.pdig.0000249.

[10] H. W. Gichuhi, M. Magumba, M. Kumar, and R. W. Mayega, 'A machine learning approach to explore individual risk factors for tuberculosis treatment non-adherence in Mukono district', *PLOS Glob Public Health*, vol. 3, no. 7, p. e0001466, Jul. 2023, doi: 10.1371/journal.pgph.0001466.

[11] X. Kuang, F. Wang, K. M. Hernandez, Z. Zhang, and R. L. Grossman, 'Accurate and rapid prediction of tuberculosis drug resistance from genome sequence data using traditional machine learning algorithms and CNN', *Sci Rep*, vol. 12, no. 1, p. 2427, Feb. 2022, doi: 10.1038/s41598-022-06449-4.

[12] Y. Shu *et al.*, 'A Machine Learning Method for Differentiation Crohn's Disease and Intestinal Tuberculosis', *JMDH*, vol. Volume 17, pp. 3835–3847, Aug. 2024, doi: 10.2147/JMDH.S470429.

[13] K.-M. Liao, C.-F. Liu, C.-J. Chen, J.-Y. Feng, C.-C. Shu, and Y.-S. Ma, 'Using an Artificial Intelligence Approach to Predict the Adverse Effects and Prognosis of Tuberculosis', *Diagnostics*, vol. 13, no. 6, p. 1075, Mar. 2023, doi: 10.3390/diagnostics13061075.

[14] 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.

[15] S. Govindarajan, S. R. Manuskandan, and R. Swaminathan, 'Diagnostics of Multi Drug Resistant Tuberculosis in Chest Radiographs using Local Textures & Extreme Gradient Boosting', *Current Directions in Biomedical Engineering*, vol. 9, no. 1, pp. 721–724, Sep. 2023, doi: 10.1515/cdbme-2023-1181.

[16] A. Mahajan *et al.*, 'A Novel Stacking-Based Deterministic Ensemble Model for Infectious Disease Prediction', *Mathematics*, vol. 10, no. 10, p. 1714, May 2022, doi: 10.3390/math10101714.