International Journal of Computer Applications (0975 - 8887)
Volume 187 - No.42, September 2025

Machine Learning Frameworks for Effective Diagnosis of Parkinson's Disease using NB, GNN, and GBM Algorithms

Tarun S.J.
Department of Electronics and Communication Engineering
Biluru Gurubasava Mahaswamiji Institute of Technology
Mudhol - 587313, Karnataka, India

Gururaj S. Kori
Department of Electronics and Communication Engineering
Biluru Gurubasava Mahaswamiji Institute of Technology
Mudhol - 587313, Karnataka, India

Supreet Hiremath
Department of Electronics and Communication Engineering
Biluru Gurubasava Mahaswamiji Institute of Technology
Mudhol - 587313, Karnataka, India

Poornima M. Chanal
Department of Electronics and Communication Engineering
Basaveshwar Engineering College
Bagalkote - 587103, Karnataka, India

ABSTRACT

Parkinson's Disease (PD) is a neuro degenerative disorder affecting millions of patients globally, causing motor impairments like tremors & stiffness, impacting on daily activities and quality of life. Parkinson's disease arises when dopamine producing neurons in the substantia nigra, a region of the midbrain, disrupting the normal functioning of the basal ganglia. This neuronal loss leads to difficulties in speech, writing, walking and performing everyday tasks. As the condition progresses, symptoms worsen and non-motor issues such as cognitive decline, mood disorders and sleep disturbances often emerge. The frameworks investigates the potential of Machine Learning (ML) algorithms in predicting PD. Machine learning algorithms like Naive Bayes (NB) Classifier, Graph Neural Network (GNN) and Gradient Boosting Machine (GBM) Protocols are applied to patient's data like demographics, clinical evaluations and potential biomarkers etc. Naive Bayes classifier is a simple but effective probabilistic model that performs well with categorical data and assumes feature independence. Graph Neural Network is a flexible algorithm capable of modeling complex non-linear relationships in data. Gradient Boosting is powerful ensemble method that iteratively improve predictions by combining weak learners, optimizing for accuracy and minimizing errors. Simulation results shows the performance of the proposed ML algorithms, which significantly enhances prediction of PD in terms of accuracy upto 96.4%, sensitivity of 97.1%, selectivity ranging upto 94.3%, positive and negative predictive values, and F1-score etc.

Keywords

Parkinson Disease (PD), Naive Bayes (NB) Classifier, Graph Neural Network (GNN), Gradient Boosting Machine (GBM)

1. INTRODUCTION

Parkinson's disease is a neuro-degenerative movement disorder that primarily affects brain cells in the substantia nigra, which produce the neuro-transmitter dopamine. Dopamine is crucial for transmitting signals that enable smooth and coordinated muscle movements. Symptoms of Parkinson's disease manifest when approximately 80% of dopamine producing cells are lost, though the exact cause of this cell death remains unknown [1], [2].

Although there is no cure for Parkinson's disease, various therapies effectively manage symptoms and improve patient's quality of life. The most commonly prescribed medication, levodopa/carbidopa, boosts dopamine levels in the brain, while other drugs, such as anti-cholinergics, help control involuntary muscle movements. Advanced cases may benefit from deep brain stimulation and similar therapies, which reduce tremors and decrease reliance on medication. Additionally, rehabilitation plays a crucial role in managing PD, incorporating strength training, gait & balance exercises, yoga, meditation, and hydrotherapy to enhance physical & mental health, ultimately improving the overall quality of life for individuals with PD [3]–[5].

According to World Health Organization (WHO) and National Institutes of Neurological Disorders and Stroke (NINDS), in 2019 over 8.5 million individuals worldwide were living with PD, a prevalence that has doubled over the past 25 years. This trend is expected to persist, with the global burden of PD projected to surpass more than 20 million cases by 2040. PD accounted for 5.8 million Disability Adjusted Life Years (DALYs), an 81% increase since 2000, and caused 3,29,000 deaths, reflecting a rise of over 100% in next few years. According to the study, more than 7 million elders in India are affected by PD. with an estimated 60,000 new diagnoses annually equivalent to one person every fifteen minutes. Parkinson's disease impacts men and women equally (but more

found in men), while the average onset age is more than 55 years [6], [7].
proposed algorithms aims to predict PD with improved accuracy by using machine learning algorithms such as Naive Bayes Classifier, Graph Neural Network and Gradient Boosting Protocols. Proposed ML frameworks are trained on a range of risk factors associated with PD, including attributes such as age, gender, genetic information, neuroimaging data, motor and non-motor symptoms, family and medical history, cognitive assessments, and lifestyle factors.

1.1 Objectives

The primary objective of utilizing machine learning for Parkinson's disease prediction is to develop and validate robust model that can accurately identify early signs of Parkinson's disease, enabling timely diagnosis and intervention. The objectives are as follows:

—The main objective is to create an efficient model for early diagnosis of Parkinson's disease using Naive Bayes Classifier, Graph Neural Network and Gradient Boosting Machines.

—To combine various data sources, including patient health histories, voice recordings, movement analysis, neuro-imaging, and clinical records to optimize feature selection and improve diagnostic accuracy.

—To compare the diagnostic efficacy, scalability, and reliability of Naive Bayes Classifier, Graph Neural Network and Gradient Boosting Machines for Parkinson's disease prediction and progression analysis.

The rest of the paper is organized as follows: A comprehensive literature survey on Parkinson's disease prediction using machine learning is presented in Section 2. The implementation of machine learning algorithms, like Naive Bayes Classifier, Graph Neural Network and Gradient Boosting Machines, is discussed in Section 3. Section 4 provides a detailed description of the dataset, simulation setup, and performance analysis. Finally, the conclusion and future research directions are discussed in Section 5.

2. RELATED WORK

A new system using computers and machine learning algorithms like Boosted Linear Regression & Multilayer Perception are used to analyse sleep patterns, smell loss, and brain scans of the PD patients is done in [8]. The results are better than other methods in finding Parkinson's disease early, with an accuracy rate of almost 95%. The results are also compared with exiting ML algorithms like Decision Tree (DT) with accuracy of 92.87%, Logistic Regression (LR) with accuracy of 89%, K-Nearest Neighbors (KNN) with accuracy of 87.17%.
A Deep Learning (DL) model for early detection of PD using a genetic algorithm and KNN technique, achieving over 94% accuracy and 95% precision is discussed in [9]. The model optimizes feature selection, reducing complexity and enhancing performance compared to other methods. The hybrid algorithm utilizes transfer learning with models like ResNet50, VGG19, and Inception-V3 for feature extraction from handwritten records.
The key symptoms of PD include tremor, bradykinesia, and rigidity. Similar symptoms are observed in related disorders, such as dementia with Lewy bodies and multiple system atrophy, though these conditions are less responsive to therapy. Feature selection is carried out using rough set theory, while dimensionality reduction is performed through Principal Component Analysis (PCA). The model's performance is evaluated using classifiers such as Deep Neural Networks (DNN), Random Forests (RF), and Support Vector Machines (SVM). Efficiency is measured using metrics like the confusion matrix, accuracy, precision, and recall is discussed in [10], [11].
Dysphonia, affecting nearly 90% of PD patients, making early detection crucial. Speech signals from 252 subjects are applied to machine learning algorithms to classify PD based on language features. By integrating multiple classifiers, a diagnostic accuracy up to 94% is achieved [12]. In [13], Artificial Neural Network (ANN) is applied to predict Parkinson from acoustic datasets. Cross-validation and hyper parameter tuning ensured accurate results. The classifiers achieved accuracies of 93.41% and 92.35%, with improved sensitivity, specificity, precision, and AUC.
Machine learning and Deep learning techniques, such as KNN and Feed forward Neural Network (FNN) models, are employed to differentiate between PD patients and healthy individuals based on voice signals [14]. The study uses the UCI dataset, consisting of 195 recordings from 31 patients. The FNN model achieved an accuracy of 94.11%, while the KSVM model reached 93.89%, highlighting the potential of ML and DL for early PD diagnosis.
Machine learning enhances Parkinson's disease prediction by analyzing speech, motor, and cognitive biomarkers. Classifiers like Random Forest, KNN, and XGBoost process voice and movement data for early detection. Techniques like SMOTE ensure balanced training, while normalization improves precision. Performance metrics, including accuracy and F1-score, evaluate models, with XGBoost and Random Forest showing superior results. AI driven diagnostics enable early, cost effective detection, aiding timely intervention and treatment planning [15], [16].
Artificial Intelligence and Machine Learning (AIML) driven models have significantly improved PD detection by analyzing neuroimaging, sensor data and motor biomarkers. Techniques such as CNNs, XGBoost, and deep learning models like VGG19 and ResNet-50 enhance diagnostic precision using MRI, EEG, gait and voice data. VGG19-INC achieves 93.45% accuracy in PD classification, while LIME enhances explainability by highlighting critical image features [17], [18].
Table 1 provides an overview of several databases employed in the machine learning based diagnosis of PD. These datasets contain diverse types of data, such as voice recordings, clinical evaluations, and movement related information, all of which are essential for accurate PD diagnosis and monitoring its progression. Among the datasets are prominent sources like the UCI Parkinson's Disease Dataset (PDD) and the Parkinson's Telemonitoring Dataset (PDT), along with others like the mPower Dataset and the Parkinson's Progression Markers Initiative (PPMI). Furthermore, additional datasets such as NewHandPD, the Hungarian Statlog Database (HSD), and IEEE Dataport expand the range of available data for PD research.
Table 2 offers a detailed summary of different supervised, unsupervised, and reinforcement learning algorithms, along with their associated performance metrics, applied to the analysis of Parkinson's disease. It highlights key metrics such as accuracy, sensitivity, specificity, F1 score, precision, and recall, which are used to assess the effectiveness of these models in diagnosing PD.
Algorithms such as SVM, LR, and DT may struggle with noise and are susceptible to overfitting. While RF and Multi Layer Perceptron (MLP) are effective, they are challenging to interpret. XGB and AdaBoost tend to be computationally heavy and sensitive to outliers. Markov Models and Bayesian Networks rely on assumptions that could limit their accuracy, and deep learning models like CNN, RNN, and DNN demand large datasets and substantial computa-

Table 1. : Various Database for Parkinson's Disease

Database	Description	Link
UCI Parkinson's Disease Dataset (UCI Repository)	Includes acoustic features from voice recordings of 80 subjects (40 with PD, 40 without).	https://archive.ics.uci.edu
Parkinson's Telemonitoring Dataset (UCI Repository)	Contains 5,875 instances, used for classifying PD based on speech features.	https://archive.ics.uci.edu
Parkinson Speech Dataset (Kaggle)	Voice data from patients with and without PD, with disease severity scores.	https://www.kaggle.com
mPower Dataset (PhysioNet)	Data collected via a mobile app, including voice recordings and motor assessments.	https://physionet.org
Parkinson's Progression Markers Initiative (PPMI)	Comprehensive data from clinical, imaging, and biological sources for PD research.	https://www.ppmi-info.org
PubMed	Database of medical research articles that often includes studies on Parkinson's Disease and diagnostic techniques.	https://pubmed.ncbi.nlm.nih.gov
NewHandPD	Dataset of hand movement data from PD patients, used for research on movement related symptoms and diagnostics.	https://newhandpd.org
Hungarian Statlog Database (UCI Repository)	Contains data for classifying patients with PD using clinical data.	https://archive.ics.uci.edu
Di-Scri Database	Data from clinical tests and physical assessments used to predict PD severity and progression.	https://www.di-scri.org
IEEE Dataport	A comprehensive resource for various datasets, including those focused on PD diagnosis and monitoring.	https://ieee-dataport.org

Table 2. : Machine Learning Algorithms and Performance Metrics Analysis

Machine Learning Algorithms	Performance Metric	Accuracy (%)
SVM, LR, DT, KNN, RF, MLP	Accuracy, Sensitivity, Specificity, F1 Score	91.4
XGBoost, AdaBoost	Accuracy, Sensitivity, Error Rate	92.3
Bayesian Networks, Markov Models	Accuracy, Precision, Recall	89.6
CNN, RNN, ANN, DNN, RL	Accuracy, Sensitivity, Specificity, AUC	94.1
Randomized Search CV, FNN	Accuracy, Precision, F1-Score	94.11

tional resources. Choosing the optimal model involves balancing data availability, computational complexity, and interpretability.
Naive Bayes Classifier offers probabilistic modeling based on feature independence, making it computationally efficient for large datasets. Graph Neural Network will help to capture complex, nonlinear relationships between input features and disease progression. Gradient Boosting will improve prediction accuracy by combining weak learners to iteratively correct errors made by previous models. By integrating these supervised and semi supervised ML algorithms, the proposed models to enhance early diagnosis, predict disease progression, and enable personalized treatment recommendations for PD patients, thus improving clinical outcomes and healthcare management.

3. PROPOSED MACHINE LEARNING FRAMEWORKS

This section explores three machine learning models Naive Bayes (NB) Classifier, Graph Neural Network (GNN) and Gradient Boosting Machine (GBM) for predicting the presence or absence of Parkinson's disease. Figure 1 depicts the proposed framework for PD diagnosis. These models are trained using a comprehensive dataset containing diverse patient information, as outlined in Table 1. The dataset undergoes data preprocessing, which includes cleaning, normalization, and feature selection. After preprocessing, the data is divided into training (70%) and testing (30%) sets. The training data is used to build predictive models that identify patterns related to PD. The models are evaluated on the testing dataset using performance metrics such as accuracy, sensitivity, specificity, and F1-score. To enhance model reliability, K-fold cross validation is applied, splitting the dataset into 10 subsets, with each subset serving as the testing set once. The average performance across these iterations provides a robust estimate of the model's accuracy, ensuring reliable PD diagnosis outcomes.

3.1 Naive Bayes (NB)

Naive Bayes is a probabilistic classifier based on Bayes' Theorem, commonly used for predicting the presence or absence of PD. It operates under the conditional independence assumption, where it assumes that the features (e.g., clinical measurements, patient demographics) are conditionally independent given the class label (PD or no PD). This simplifies the calculation of the posterior probability of a class by decomposing the joint likelihood of the features into individual feature likelihoods. The strengths of the Naive Bayes algorithm for predicting the presence or absence of PD lie in its simplicity, computational speed, and effectiveness with high dimensional clinical data. However, it may face challenges when the conditional independence assumption is strongly violated or when handling continuous data without proper preprocessing, such as transforming it into discrete values or applying Gaussian Naive Bayes for continuous features [19]-[21].
The figure (2) illustrates the workflow of a Naive Bayes based classification model for PD detection. It begins with a set of training documents (D) used to compute prior probabilities $P(C)$, word probabilities $P(X_i|C)$, and weight parameters X_i. A test

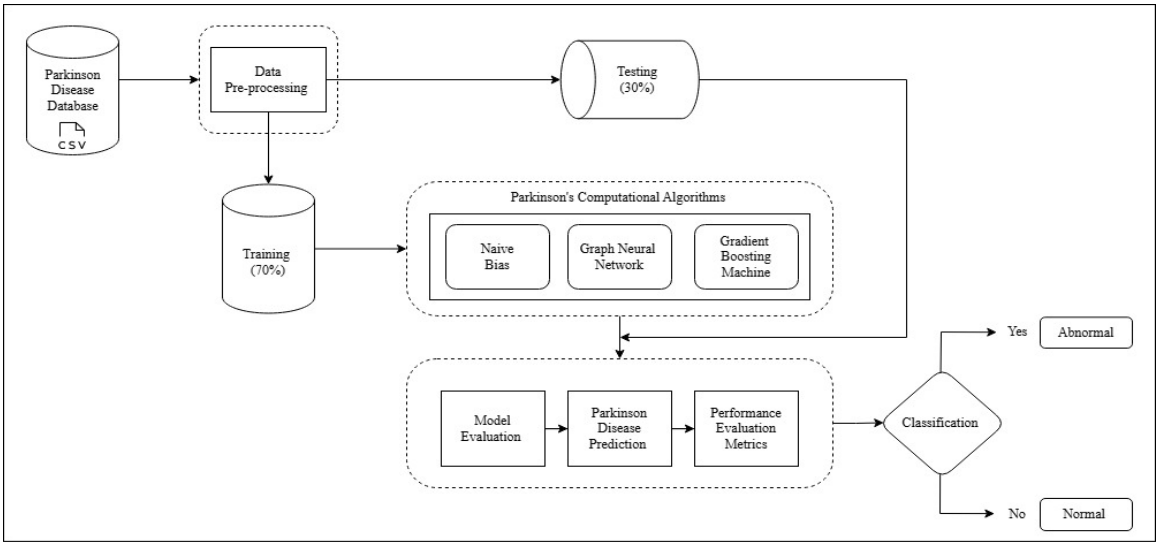

Fig. 1: Machine Learning Frameworks for Parkinson's Disease Diagnosis

document (d) is then processed, where conditional probabilities $P(X_i|X_i, C)$ are computed, and the model predicts whether the document corresponds to a PD or healthy case. The classification is based on Bayesian probability theory, ensuring efficient and interpretable PD detection.
The NB classifier is based on Bayes' theorem, which is expressed in equation (1).

$$P(C|X) = \frac{P(X|C)P(C)}{P(X)} \tag{1}$$

where,

—$P(C|X)$ is the posterior probability of class C (PD or No PD) given features X.
—$P(X|C)$ is the likelihood of the observed data given class C.
—$P(C)$ is the prior probability of class C.
—$P(X)$ is the marginal probability of the observed data.

Since NB algorithm assumes that features $X_1, X_2, ..., X_n$ are conditionally independent given C, the likelihood can be decomposed. Equation (2) represents the NB assumption, where the likelihood $P(X|C)$ is factorized as the product of individual feature probabilities, assuming conditional independence given class C.

$$P(X|C) = \prod_{i=1}^{n} P(X_i|C) \tag{2}$$

Substituting equation (2) into equation (1), we obtain equation (3). Equation (3) shows the posterior probability $P(C|X)$, which is proportional to the product of the prior probability $P(C)$ and the likelihood $P(X_i|C)$.

$$P(C|X) \propto P(C) \prod_{i=1}^{n} P(X_i|C) \tag{3}$$

where, $\propto$ denotes proportionality since $P(X)$ is a constant for all classes.

$$\hat{C} = \arg\max_{C} \left[P(C) \prod_{i=1}^{n} P(X_i|C) \right] \tag{4}$$

The classification decision is based on selecting the class C that maximizes the probability as shown in equation (4).

$$P(C = PD|X) = \frac{P(C = PD) \prod_{i=1}^{n} P(X_i|C = PD)}{P(X)} \tag{5}$$

Naive Bayes based Parkinson's disease diagnosis is shown in equation (5).

$$P(C = NoPD|X) = \frac{P(C = NoPD) \prod_{i=1}^{n} P(X_i|C = NoPD)}{P(X)} \tag{6}$$

Similarly, for a healthy individual it shown in equation (6), using Bayes' theorem with likelihood decomposition.

$$P(C = PD|X) > P(C = NoPD|X) \Rightarrow PDDiagnosed \tag{7}$$

The final classification decision is made based on equation (7). Otherwise, the patient is classified as healthy (No PD).
The following steps outline the process of applying the Naive Bayes Algorithm for predicting the presence or absence of Parkinson's Disease (PD). Each step describes a key operation from calculating probabilities to evaluating the model's performance.

—**Prior Probabilities**: Calculate $P(c)$ for each class c by determining the frequency of each class in the dataset, i.e., $P(c) = \frac{countofinstancesinc}{totalinstancesinS}$.
—**Conditional Probabilities**: Compute $P(a = v|c)$ for each attribute a and its value v given each class c, by calculating the relative frequency of each value v for attribute a in class c.
—**Classification**: For a new instance, calculate the posterior probability for each class c using Bayes' Theorem and select the class with the highest posterior probability.

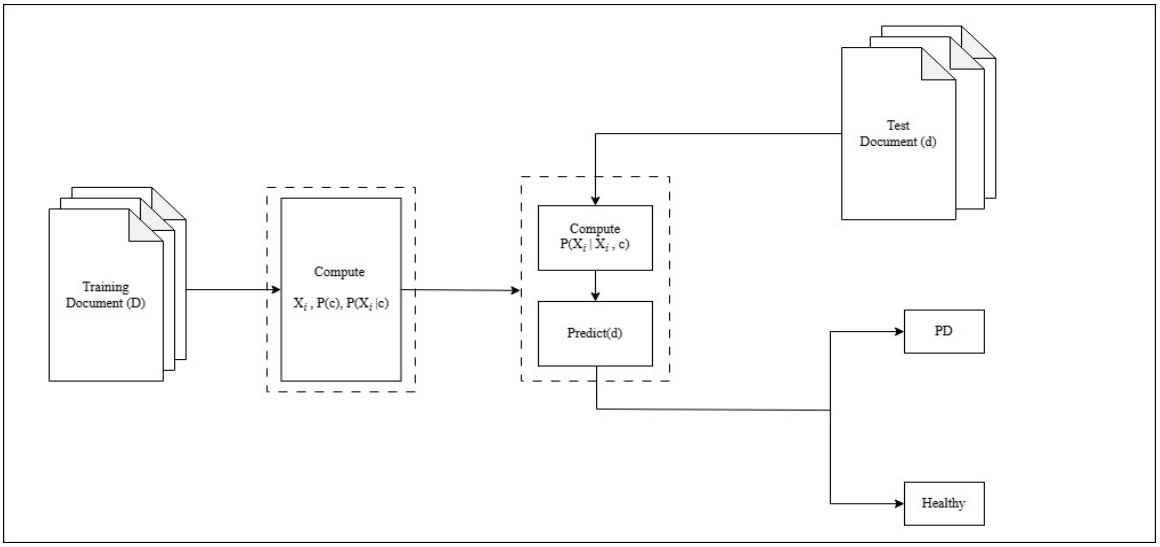

Fig. 2: Naive Bayes Framework for Parkinson's Disease Diagnosis

—**Model Evaluation**: Evaluate the Naive Bayes model using performance metrics such as accuracy, precision, recall, and F1-score on the testing dataset.

Algorithm (1) describes the steps of the Naive Bayes classifier, where prior and conditional probabilities for each class and feature are calculated. These probabilities are then used to construct the classification model, enabling predictions based on the learned relationships between features and classes.

Algorithm 1 Naive Bayes Algorithm

Require: Training dataset S
Ensure: Naive Bayes Model
Calculate prior probabilities for each class c:
for each class c in target attribute **do**
    $P(c) = \frac{countofinstancesinc}{totalinstancesinS}$
end for
Calculate conditional probabilities for each attribute a given the class c:
for each attribute a in attributes **do**
    for each value v of attribute a **do**
        for each class c in target attribute **do**
            $P(a = v|c) = \frac{countofinstanceswherea=vandclass=c}{countofinstancesinc}$
        end for
    end for
end for
Return Naive Bayes Model with calculated probabilities $P(c)$ and $P(a = v|c)$

3.2 Graph Neural Networks (GNN)

Graph Neural Networks offers an innovative method for diagnosing Parkinson's disease, utilizing semi supervised learning to enhance prediction accuracy. In this approach, patients' data is modelled as a graph, where each node represents a patient, and edges denote relationships or similarities between patients. This graph based structure enables GNNs to capture intricate patterns and interactions within the data. The GNN architecture employs multiple layers that aggregate information from both nodes and edges, allowing the model to generate rich representations of patients. By leveraging both labelled and unlabelled data in a semi supervised learning framework, GNNs improve model performance by minimizing a loss function that incorporates both supervised and unsupervised components. This combination of graph based learning and semi supervised methods provides a powerful tool for doctors, enabling more accurate PD diagnoses and better informed treatment decisions [22]-[25].
The figure (3) represents the workflow of a GNN for PD prediction. It starts with input graphs, where nodes represent entities (e.g., brain regions or biomarkers) and edges define relationships. These graphs pass through GNN layers, which learn node representations by aggregating information from neighbours. The node embeddings capture structural and feature based relationships. A readout layer condenses node embeddings into a graph embedding, representing the entire graphs feature space. Finally, classification layers process the graph embedding to predict PD or a healthy condition, enabling an interpretable deep learning based diagnosis.
In GNN a graph is represented as $G = (V, E)$, where:

—V is the set of N nodes (patients or feature representations).
—E is the set of edges (relationships between nodes).
—$\mathbf{X} \in R^{N \times d}$ is the input feature matrix.
—$\mathbf{A} \in R^{N \times N}$ is the adjacency matrix.
—$\mathbf{D}$ is the degree matrix, where $D_{ii} = \sum_j A_{ij}$.

To ensure numerical stability, a normalized adjacency matrix is shown in equation (8) & (9).

$$\tilde{\mathbf{A}} = \mathbf{A} + \mathbf{I} \tag{8}$$

$$\tilde{\mathbf{D}}_{ii} = \sum_j \tilde{\mathbf{A}}_{ij} \tag{9}$$

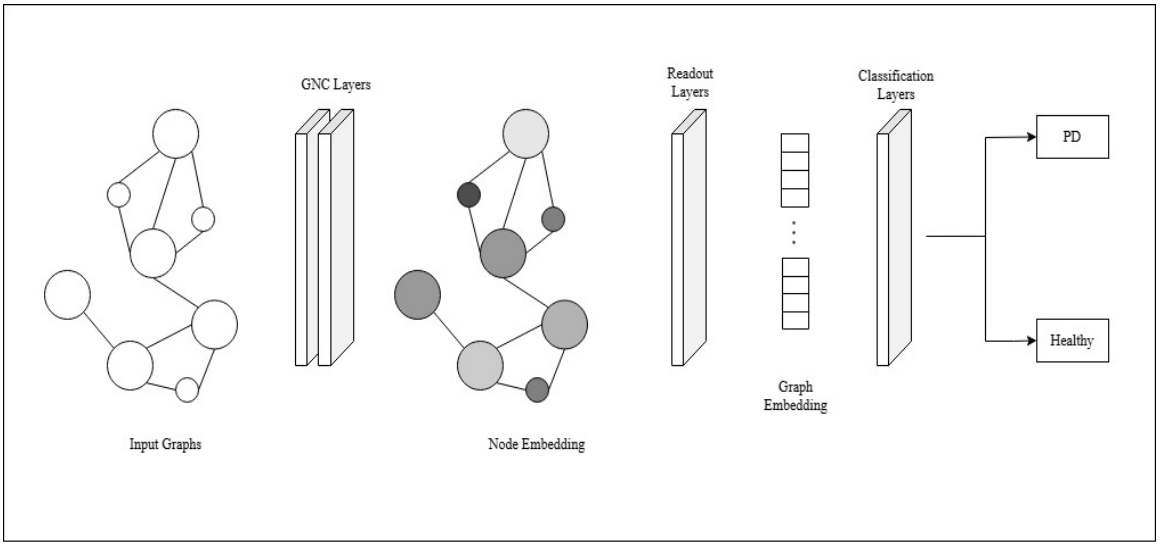

Fig. 3: Graph Neural Networks Framework for Parkinson's Disease Diagnosis

where, $\mathbf{I}$ is the identity matrix.
The feature transformation in a Graph Convolutional Layer (GCN) is given by equation (10).

$$\mathbf{H}^{(l)} = \sigma\left(\tilde{\mathbf{D}}^{-\frac{1}{2}}\tilde{\mathbf{A}}\tilde{\mathbf{D}}^{-\frac{1}{2}}\mathbf{H}^{(l-1)}\mathbf{W}^{(l)}\right) \tag{10}$$

where:

—$\mathbf{H}^{(l)}$ is the feature representation at layer l.
—$\mathbf{W}^{(l)}$ is the learnable weight matrix for layer l.
—σ is an activation function (e.g., ReLU).

Equation (11) & (12) shows the input layer and hidden layer operation of GNN.

$$\mathbf{H}^{(0)} = \mathbf{X} \tag{11}$$

$$\mathbf{H}^{(l)} = \sigma\left(\tilde{\mathbf{D}}^{-\frac{1}{2}}\tilde{\mathbf{A}}\tilde{\mathbf{D}}^{-\frac{1}{2}}\mathbf{H}^{(l-1)}\mathbf{W}^{(l)}\right) \tag{12}$$

Each node aggregates information from its neighbours with feature update rule as shown in equation (13).

$$\mathbf{h}_i^{(l)} = \sigma\left(\sum_{j\in\mathcal{N}(i)} \frac{1}{\sqrt{d_i d_j}}\mathbf{W}^{(l)}\mathbf{h}_j^{(l-1)}\right) \tag{13}$$

where:

—$\mathbf{h}_i^{(l)}$ is the updated feature vector for node i.
—$\mathcal{N}(i)$ represents the neighbors of node i.
—d_i, d_j are node degrees.

After multiple GCN layer operations, the final graph level representation is computed and readout function that aggregates the node representations from the last GCN layer is shown in equation (14).

$$\mathbf{z}_G = READOUT(\{\mathbf{h}_i^{(L)} | i \in V\}) \tag{14}$$

where, L is the last GCN layer. Equation (15) express the Use mean pooling computing and averaging the node embeddings from the final GCN layer.

$$\mathbf{z}_G = \frac{1}{|V|}\sum_{i\in V}\mathbf{h}_i^{(L)} \tag{15}$$

A softmax classifier predicts the probability of PD presence based on the learned graph representation as shown in equation (16).

$$P(y|G) = softmax(W_o\mathbf{z}_G + b) \tag{16}$$

where:

—W_o and b are learnable classification parameters.
—$P(y|G)$ is the probability of PD presence.

The classification loss function (cross entropy loss) measures how well the predicted probabilities align with the actual labels are shown in equation (17).

$$\mathcal{L} = -\sum_{i=1}^{N} y_i \log P(y_i|G_i) \tag{17}$$

where y_i is the true label (PD or No PD).
The following key steps outline the process of training a Graph Neural Network for predicting Parkinson's disease using graph structured patient data. These steps enable the model to capture complex relationships and improve diagnostic accuracy.

—**Initialization**: Randomly initialize the weights W_l for each layer l using the graph data (node features X and adjacency matrix A) to transform the node features.
—**Forward Pass**: For each layer l, aggregate the neighbouring node features using the adjacency matrix A, and compute the aggregated features by considering the neighbours of each node i, i.e., $neighbors = get\_neighbors(i, A)$ and $aggregated\_features[i] = \sum_{j\in neighbors} X_j$.

—**Feature Update**: Apply the learned weights W_l to the aggregated features and pass them through an activation function $activation(\cdot)$ to update the node features, i.e., $X \leftarrow activation(aggregated\_features \times W_l)$.

—**Output**: After iterating through all layers, use the updated node features X to make predictions, such as predicting the presence or absence of PD.

Algorithm 2 Graph Neural Network (GNN)

Require: Graph data (node features X, adjacency matrix A)
Ensure: GNN Model
Initialize weights for each layer:
for layer l from 0 to $L-1$ **do**
    $W_l \leftarrow random\_initialization(num\_features)$
end for
Forward pass through the GNN:
for layer l from 0 to $L-1$ **do**
    aggregated_features $\leftarrow []$
    for node i in range(num_nodes) **do**
        neighbours $\leftarrow get\_neighbours(i, A)$
        neighbour_features $\leftarrow [X[j] for j in neighbours]$
        aggregated_feature $\leftarrow neighbour\_features$
        aggregated_features.append(aggregated_feature)
    end for
    $X \leftarrow activation(aggregated\_features \times W_l)$
end for
return updated node features X as GNN output

Algorithm (2) outlines the training process of a GNN to predict the presence or absence of PD. Where, each patient is represented as a node, and relationships between patients are modeled as edges in the graph. The input includes node features (X), which represent patient data (e.g., clinical symptoms), and the adjacency matrix (A), which captures the relationships between nodes.

3.3 Gradient Boosting Machines (GBM)

Gradient Boosting Machines predict the presence or absence of Parkinson's disease by iteratively combining decision trees, where each tree corrects the residual errors of previous trees using gradient descent. The model is trained on a labelled dataset with features such as clinical symptoms and test results. Through successive iterations, GBMs learn complex patterns and refine predictions. The final prediction is made by aggregating the outputs of all trees, typically through a weighted sum or majority vote. Key hyper parameters like learning rate, number of trees, and tree depth must be carefully tuned to optimize performance and prevent over fitting [26]-[28].
The figure (4) illustrates the working mechanism of a GBM for PD prediction. It sequentially trains multiple decision trees, where each tree learns from the errors of the previous one. The first tree is trained on the initial dataset, and subsequent trees correct the misclassification by assigning higher weights to incorrectly predicted samples. The final prediction is obtained by aggregating the weighted outputs $(X1, X2, ..., Xn)$ of all trees. Correct and incorrect predictions are marked, highlighting the models iterative refinement process, which enhances accuracy and robustness for disease classification.
Given a dataset in GBM with N samples, where each sample has feature set X_i and corresponding label y_i, the dataset is represented in equation (18).

$$D = \{(X_i, y_i) | i = 1, 2, \ldots, N\} \tag{18}$$

The initial prediction model $F_0(X)$ is chosen as the constant value that minimizes the loss function is expressed in equation (19).

$$F_0(X) = \arg\min_c \sum_{i=1}^{N} L(y_i, c) \tag{19}$$

where, $L(y_i, c)$ is the loss function (e.g., Mean Squared Error for regression or Log Loss for classification).
For each boosting iteration $m = 1, 2, \ldots, M$, the model is refined in equations (20)-(24).
The residuals represent the negative gradient of the loss function with respect to the previous models prediction is presented in equation (20).

$$r_i^{(m)} = -\frac{\partial L(y_i, F(X_i))}{\partial F(X_i)} \tag{20}$$

A new decision tree that fits a weak learner decision tree $h_m(X)$ is trained to approximate is discussed in equation (21)with residuals.

$$h_m(X) \approx r_i^{(m)} \tag{21}$$

The optimal step size γ_m is determined by minimizing the loss function in equation (22).

$$\gamma_m = \arg\min_\gamma \sum_{i=1}^{N} L(y_i, F_{m-1}(X_i) + \gamma h_m(X_i)) \tag{22}$$

The model is updated by incorporating the weak learners contribution in equation (23).

$$F_m(X) = F_{m-1}(X) + \gamma_m h_m(X) \tag{23}$$

After M iterations, the final prediction function for PD diagnosis is presented in equation (24).

$$\hat{y} = \sigma(F_M(X)) \tag{24}$$

where $\sigma(\cdot)$ is the sigmoid activation function used for binary classification to output the probability of Parkinson's disease presence. The following key steps outline the process of using Gradient Boosting Machines, which iteratively minimize the loss function, compute pseudo residuals, and update the model to capture complex patterns, optimizing performance and preventing over fitting.

—**Training Data**: The dataset X contains feature data (e.g., clinical symptoms and medical history), while y contains corresponding labels (PD or no PD).

—**Initialization**: The algorithm initializes with an initial model $F_0(x)$, often a constant value such as the mean of the target variable y.

—**Iteration**: In each iteration m, compute the pseudo residuals $r_i = y_i - F_{m-1}(x_i)$ for each instance i, and train a weak learner (e.g., a decision tree) to predict these residuals.

—**Model Update**: Update the model by adding the predictions of the weak learner $h_m(x)$, scaled by the learning rate ν and an optimal multiplier γ_m to adjust the contribution of the weak learner.

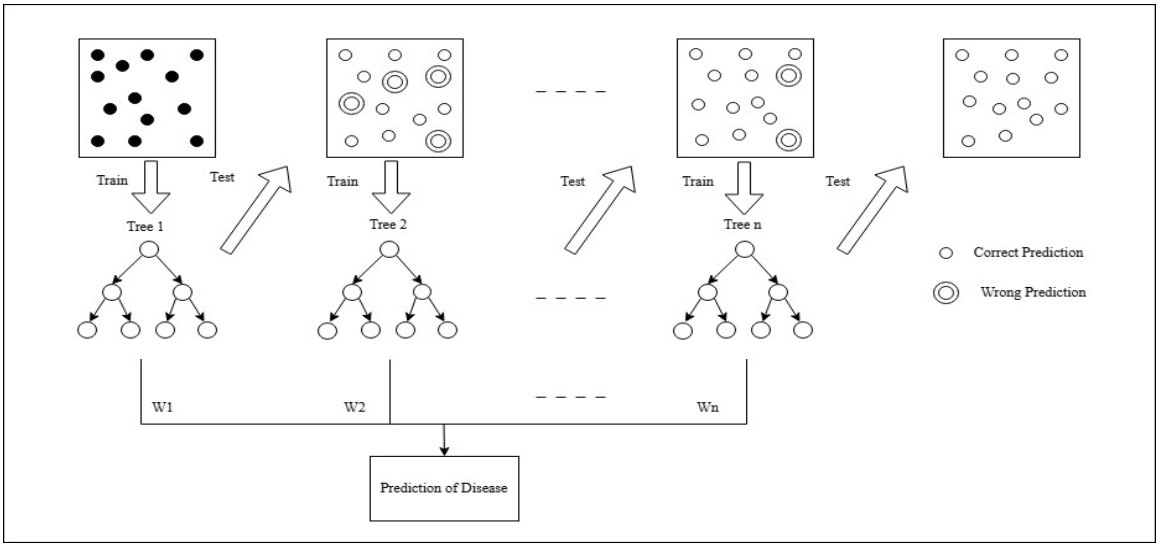

Fig. 4: Gradient Boosting Machines Framework for Parkinson's Disease Diagnosis

—**Final Model**: After M iterations, the final model $F_M(x)$ aggregates all weak learners' predictions and is used to predict whether a patient has Parkinson's Disease based on the features X.

Algorithm (3) details the process of training a Gradient Boosting Machine (GBM), where weak learners are iteratively fit to pseudo residuals, and the model is updated using gradient descent. This approach refines predictions by minimizing the loss function, effectively capturing complex data patterns while preventing over fitting through careful control of model complexity and hyper parameters.

Algorithm 3 Gradient Boosting Machines (GBM)

Require: Training data (X, y), Number of iterations (M), Learning rate (ν)
Ensure: Gradient Boosted Model
1: Initialize $F_0(x)$ // Initial model
2: **for** $m = 1$ to M **do**
3: Compute pseudo residuals $r_i = y_i - F_{m-1}(x_i)$ for each instance i
4: Fit weak learner $h_m(x)$ to r_i
5: Compute optimal multiplier γ_m
6: Update model: $F_m(x) \leftarrow F_{m-1}(x) + \nu \cdot \gamma_m \cdot h_m(x)$
7: **end for**
8: **return** Final model $F_M(x)$

Table 3 compares three machine learning models NB, GNN, and GBM highlighting their respective strengths, weaknesses, and most suitable applications in the prediction of PD. The table outlines the ML models capabilities in capturing complex relationships, handling structured data, and managing feature independence, along with their specific use cases in PD disease diagnosis, such as personalized medicine, gene disease interactions, and disease progression monitoring.

4. DATASET, SIMULATION & PERFORMANCE ANALYSIS

In this section, we discuss the dataset, the tools used to simulate the proposed models, and the methods employed to evaluate the performance of NB, GNN & GBM ML Model for predicting the presence or absence of PD. The performance of these algorithms is assessed to determine their effectiveness in PD prediction [29].

4.1 Dataset

The proposed diagnosis of PD is simulated using the dataset sourced from Kaggle containing voice recordings of patients, demographic information, clinical symptoms, and Unified Parkinson's Disease Rating Scale (UPDRS) scores as shown in table 4 and 5.
The dataset includes various voice features such as base frequency variations, wave variations, jitter, shimmer, and noise ratio, which are commonly used in speech analysis for PD diagnosis. Each feature represents a specific aspect of vocal characteristics, with multiple generated values for each attribute. These features play a crucial role in understanding speech patterns that are indicative of PD. The dataset are suitable for NB and GBM for predicting PD, especially since these algorithms perform well with structured and tabular data. While simulating GNNs the data had been transformed into a graph structure model which develops the relationships between features or samples.

4.2 Simulation

The proposed models for Naive Bayes, Graph Neural Networks and Gradient Boosting Machines are simulated using PyCharm, a powerful Integrated Development Environment (IDE) tailored for Python. PyCharm provides a comprehensive suite of tools, including efficient code editing, debugging, version control, and code completion, which makes it particularly suitable for developing and testing machine learning algorithms. Table 6 summarizes the inputs used for simulating these models in PyCharm, leveraging its robust features to ensure seamless workflow and model evaluation. Key

Table 3. : Comparison of NB, GNN & GBM ML Models

ML Model	Benefits	Limitations
NB	Fast, simple, handles missing data	Assumes feature independence, less accurate for complex data
GNN	Captures complex relationships, good for multi modal data	Computationally expensive, needs large datasets
GBM	High accuracy, good with structured data	Slow training, risk of overfitting

Table 4. : Parkinson Disease Dataset

Feature/Attributes	Description
MDVP:Fo (Hz)	Mean of the base frequency of vowels
MDVP:Fhi (Hz)	Maximum base frequency of vowels
MDVP:Flo (Hz)	Minimum base frequency of vowels
MDVP:Jitter (%)	Basic frequency variations
MDVP:Jitter (Abs)	Basic frequency variations
MDVP:RAP	Basic frequency variations
MDVP:PPQ	Basic frequency variations
Jitter:DDP	Basic frequency variations
MDVP:Shimmer	Basic frequency variations
MDVP:Shimmer (dB)	Wave variations
Shimmer:APQ3	Wave variations
Shimmer:APQ5	Wave variations
MDVP:APQ	Wave variations
Shimmer:DDA	Wave variations
NHR	Noise ratio for tonal components in sound

Table 5. : Sample Dataset

Feature/Attributes	Generated Values
MDVP:Fo (Hz)	106.34, 127.64, 120.47, 112.48, 129.81
MDVP:Fhi (Hz)	132.67, 148.18, 170.13, 171.60, 153.78
MDVP:Flo (Hz)	81.54, 99.09, 87.82, 93.58, 98.57
MDVP:Jitter (%)	0.60, 0.50, 0.54, 0.65, 0.63
MDVP:Jitter (Abs)	0.01, 0.01, 0.01, 0.01, 0.01
MDVP:RAP	0.50, 0.48, 0.45, 0.55, 0.52
MDVP:PPQ	0.40, 0.42, 0.38, 0.43, 0.41
Jitter:DDP	0.50, 0.45, 0.55, 0.48, 0.50
MDVP:Shimmer	0.45, 0.42, 0.46, 0.48, 0.44
MDVP:Shimmer (dB)	30.10, 28.65, 32.57, 29.80, 31.23
Shimmer:APQ3	0.28, 0.24, 0.23, 0.29, 0.23
Shimmer:APQ5	0.35, 0.33, 0.38, 0.32, 0.37
MDVP:APQ	0.40, 0.42, 0.43, 0.41, 0.45
Shimmer:DDA	0.56, 0.58, 0.60, 0.57, 0.61
NHR	0.17, 0.14, 0.15, 0.15, 0.14

parameters such as number of layers, learning rate, and cross validation folds (set to 10) are specified for each model. These inputs are essential for tuning the models to optimize their prediction accuracy.

4.3 Performance Parameters

In this section, key metrics used to evaluate the performance of Naive Bayes, Graph Neural Networks and Gradient Boosting Machines for predicting the presence or absence of Parkinson's disease are discussed.

—**Accuracy:** Measures overall correctness of predictions in Parkinson's disease detection by evaluating both positive (PD) and negative (healthy) classifications as shown in equation (25). It provides an overall measure of model performance.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{25}$$

Where,
—TP = True Positives (correctly predicted PD cases)
—TN = True Negatives (correctly predicted healthy cases)
—FP = False Positives (incorrectly predicted PD cases)
—FN = False Negatives (incorrectly predicted healthy cases)

Table 6. : Simulation Inputs for Parkinson's Disease Prediction

Algorithm	Simulation Inputs
NB	Smoothing Parameter: 0.01-1.0 (Prevents zero probability issues by smoothing probabilities) Cross Validation Folds: 10 (Used to evaluate the model's generalization ability)
GNN	Number of Layers: up to 5 (Controls the depth of the network) Learning Rate: 0.01-0.1 (Determines how much the model weights are updated) Cross Validation Folds: 10 (Number of data splits used for model validation)
GBM	Number of Trees: up to 100 (Number of boosting iterations or trees) Learning Rate: 0.01-0.1 (Controls the contribution of each tree to the final model) Max Depth: up to 10 (Limits the depth of the trees to prevent overfitting) Cross Validation Folds: 10 (Validates the model's performance across different subsets of the data)

—**Precision:** Indicates the proportion of correctly identified PD cases among all predicted PD cases, helping to minimize false positives in early PD detection represented in equation (26).

$$Precision = \frac{TP}{TP + FP} \tag{26}$$

—**F1-Score:** The F1-Score is the harmonic mean of Precision and Recall expressed in equation (27). Balances precision and recall, ensuring effective handling of false negatives and false positives, especially in imbalanced PD datasets.

$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{27}$$

—**Sensitivity (Recall):** Sensitivity, also known as Recall or True Positive Rate (TPR), measures how well the model detects actual PD cases, which is critical for early diagnosis and minimizing missed PD patients as shown in equation (28).

$$Sensitivity(Recall) = \frac{TP}{TP + FN} \tag{28}$$

—**Selectivity (Specificity):** Specificity, also known as Selectivity or True Negative Rate (TNR), evaluates how effectively a model distinguishes healthy individuals, reducing the chances of misclassifying non PD cases as PD shown in equation (29).

$$Selectivity(Specificity) = \frac{TN}{TN + FP} \tag{29}$$

—**ROC-AUC (Receiver Operating Characteristic - Area Under the Curve):** ROC-AUC is a measure of the model's ability to distinguish between Parkinson's disease and healthy cases by integrating sensitivity and specificity. A higher AUC indicates better classification performance across varying thresholds as expressed in equation (30).
ROC-AUC measures a model's ability to distinguish between

$$AUC = \int_0^1 Sensitivity\, d(1 - Specificity) \tag{30}$$

Table 7 summarizes the relevance of performance parameters for evaluating machine learning models in predicting Parkinson's Disease (PD). It includes key metrics like Accuracy, Precision, F1-Score, Sensitivity (Recall), Selectivity (Specificity), and ROC-AUC, each serving to assess different aspects of model performance, such as identifying true PD cases, minimizing false positives, and ensuring robust model distinction between PD and non PD. These metrics are essential for comparing models like GNN, NB, and GBM in the context of PD diagnosis.

4.4 Performance Analysis

Table 8 and fig. (5) presents the performance evaluation of three machine learning models Naive Bayes, Graph Neural Networks, and Gradient Boosting Machines for predicting the presence or absence of Parkinson's Disease (PD).
The table includes key performance metrics such as Accuracy, Precision, F1-Score, Sensitivity (Recall), Selectivity (Specificity), and ROC-AUC, which are used to assess how well each model performs in distinguishing between patients with Parkinson's Disease and healthy individuals. The statistical parameters such as Q1, Q3, IQR, and error (±1.5 × IQR) further reinforce the consistency of GBM, emphasizing its effectiveness in delivering reliable predictions across varied data subsets.

4.5 Computational Delay Analysis

In this section time delay for simulation of models forPD prediction is discussed. When comparing the time delay for simulating machine learning models like Naive Bayes, Graph Neural Networks, and Gradient Boosting Machines in predicting PD, the computational time varies significantly based on the models architecture and complexity.

—**Naive Bayes (NB):**
  —**Time Complexity:** $O(n \cdot m)$, where n is the number of samples and m is the number of features.
  —**Simulation Time:** Naive Bayes is a simple, probabilistic algorithm that assumes feature independence and computes probabilities based on the frequency of features and classes. This makes Naive Bayes the fastest model for simulation, requiring the least computational time compared to the other models.

—**Graph Neural Networks (GNN):**
  —**Time Complexity:** $O(L \cdot N \cdot D)$, where L is the number of layers, N is the number of nodes (data samples), and D is the dimensionality of the node features.
  —**Simulation Time:** GNNs are the most computationally expensive model. They involve graph based computations, such as message passing and adjacency matrix operations, and require substantial training time, particularly with deep models and large datasets. The complexity of graph based learning and the optimization of model parameters make GNN the slowest among the three models.

—**Gradient Boosting Machines (GBM):**

Table 7. : Relevance of Performance Parameters for Parkinson's Disease Prediction

Performance Metric	Relevance for PD Prediction
Accuracy	Provides a general measure of performance but may not fully reflect model effectiveness in imbalanced datasets, which are common in medical prediction tasks.
Precision	Important when false positives (predicting a patient has PD when they do not) must be minimized, avoiding unnecessary treatments or stress in clinical settings.
F1-Score	Offers a balanced evaluation between Precision and Sensitivity, especially valuable when both false positives and false negatives are critical, as in PD diagnosis.
Sensitivity (Recall)	Crucial in medical contexts where missing actual PD cases can have serious consequences. High Sensitivity ensures true PD patients are detected.
Specificity (Selectivity)	Helps minimize false positives, ensuring healthy individuals are not incorrectly diagnosed with Parkinson's Disease.
ROC-AUC	Useful for assessing model performance across thresholds. A high AUC means the model effectively distinguishes between PD and non-PD cases.

Table 8. : Statistical Summary of Model Performance for Parkinson's Disease Prediction

Performance Metric	ML Algorithm	Mean	Median	Q1 25%	Q3 75%	IQR	Error (±1.5*IQR)
Accuracy	NB	89.67	89.5	87.0	91.0	4.0	±6.0
	GNN	94.12	94.0	92.0	96.0	4.0	±6.0
	GBM	96.41	96.5	95.0	98.0	3.0	±4.5
Recall (Sensitivity)	NB	85.30	85.0	83.0	87.0	4.0	±6.0
	GNN	95.45	95.5	94.0	97.0	3.0	±4.5
	GBM	97.12	97.0	96.0	98.0	2.0	±3.0
Precision	NB	87.34	87.0	85.0	89.0	4.0	±6.0
	GNN	93.65	94.0	92.0	95.0	3.0	±4.5
	GBM	95.78	96.0	94.5	97.0	2.5	±3.75
F1-Score	NB	88.50	88.5	86.0	90.0	4.0	±6.0
	GNN	93.89	94.0	92.0	96.0	4.0	±6.0
	GBM	96.15	96.0	95.0	97.0	2.0	±3.0
Specificity	NB	92.15	92.0	90.0	94.0	4.0	±6.0
	GNN	92.98	93.0	91.5	94.5	3.0	±4.5
	GBM	94.35	94.5	93.0	96.0	3.0	±4.5
ROC-AUC	NB	88.70	88.5	87.0	90.0	3.0	±4.5
	GNN	95.80	95.5	94.0	97.5	3.5	±5.25
	GBM	97.50	97.0	96.0	99.0	3.0	±4.5

—**Time Complexity:** $O(M \cdot N \cdot \log(N))$, where M is the number of boosting rounds (trees) and N is the number of samples.
—**Simulation Time:** GBM is an ensemble method that builds multiple decision trees, which are iteratively improved. While it offers high predictive accuracy, it is computationally more expensive than Naive Bayes due to the training of multiple trees and the need for hyperparameter optimization. The training time increases with the number of trees and depth of the trees.

The fig (6) shows the computational delay (in seconds) vs. number of iterations for NB, GNN and GBM in predicting PD as the number of iterations increases during simulation, NB exhibits the lowest computational delay due to its simple probabilistic approach, making it efficient for real-time predictions. In contrast, GNN experiences a highest delay as it captures complex relationships in graph structured data, requiring more computational resources. GBM shows the higher delay, as it involves fitting multiple decision trees in an ensemble, making it computationally intensive but highly accurate. The graph highlights the trade-off between computational efficiency and predictive accuracy, with NB being faster, while GNN and GBM offer more complex models at the cost of higher computational demands.

5. CONCLUSION

The prediction of Parkinson's Disease (PD) using Naive Bayes (NB), Graph Neural Networks (GNN), and Gradient Boosting Machines (GBM) highlights the strengths and trade-offs of each model. GNN is effective in capturing complex relationships in graph based data, though it is computationally expensive. NB is fast and efficient, performing well with conditionally independent features but may not capture complex feature interactions. GBM offers high accuracy and is effective for large datasets, though it requires significant computational resources and careful tuning to prevent overfitting. The choice of algorithm should depend on the specific requirements of Parkinson's disease prediction, such as the desired accuracy, interpretability, and computational efficiency.
Future research should focus on data privacy, model interpretability, and real-world applicability to effectively integrate these algorithms into clinical practice for Parkinson's disease diagnosis. Improving model interpretability will make it easier for clinicians to understand and trust the predictions. Additionally, ensuring the real-world applicability of these models by validating them on diverse, large scale datasets will help integrate them into clinical settings, ultimately improving early diagnosis, optimizing treatment plans, and enhancing patient outcomes in Parkinson's disease care.

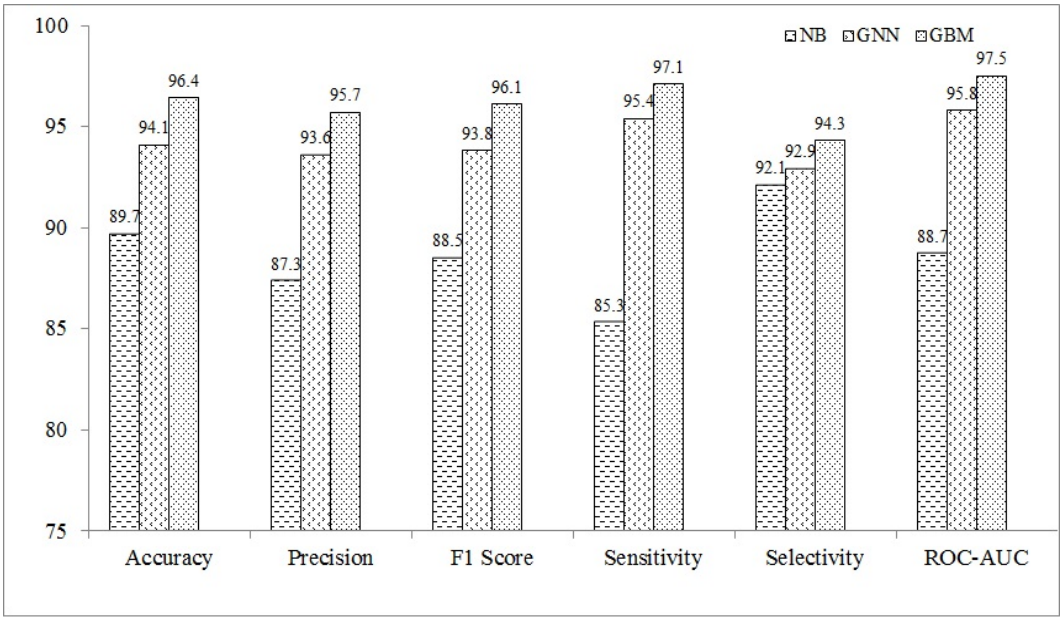

Fig. 5: Performance Parameter (%) of NB, GNN, & GBM in Predicting PD

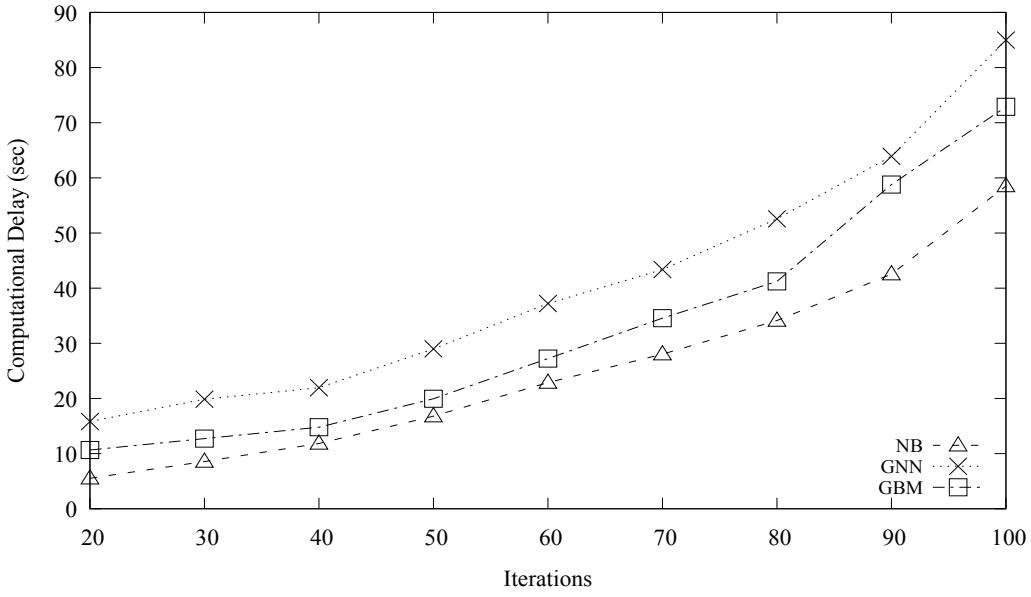

Fig. 6: Computational Delay Vs. Number of Iterations

6. REFERENCES

[1] Parkinson Disease (PD), "World Health Organization (WHO)", https://www.who.int/news-room/fact-sheets/detail/parkinson-disease

[2] Parkinson Disease (PD), "National Institutes of Neurological Disorders and Stroke (NINDS)", https://www.ninds.nih.gov/health-information/disorders/parkinsons-disease

[3] Bastiaan R. Bloem, Michael S. Okun, Christine Klein, "Parkinson's Disease", *The Lancet, Elsevier*, Vol. 97, No. 10291, pp. 2284–2303, 2021.

[4] Kevin Mc Farthing, Gary Rafaloff, Marco Baptista, et. al., "Parkinson's Disease Drug Therapies in the Clinical Trial Pipeline: 2022 Update", *Journal of Parkinson's Disease, IOS Press*, Vol. 12, pp. 1073–1082, 2022.

[5] Fatemeh N. Emamzadeh, Andrei Surguchov, "Parkinson's Disease: Biomarkers, Treatment, and Risk Factors", *Journal of*

Frontiers in Neuroscience, Vol. 12, No. 612, pp. 01–14, 2018.

[6] E. Ray Dorsey, Todd Sherer, Michael S. Okuncn, et. al., "The Emerging Evidence of the Parkinson Pandemic", *Journal of Parkinson's Disease, IOS Press*, Vol. 08, No. S1, pp. S3–S8, 2018.

[7] Fabrizio Stocchi, Daniele Bravi, Aron Emmi, et. al., "Parkinson Disease Therapy: Current Strategies and Future Research Priorities", *Nature Reviews Neurology, Springer Nature*, Vol. 20, pp. 695–7072, 2024.

[8] Surekha Tadse, Muskan Jain, Pankaj Chandankhede, "Parkinson's Detection Using Machine Learning", *Proc. of the IEEE* 5^{th} *International Conference on Intelligent Computing and Control Systems*, Madurai, India, 2021.

[9] Sura Mahmood Abdullah, Thekra Abbas, Munzir Hubiba Bashir, et. al.,"Deep Transfer Learning Based Parkinson's Disease Detection Using Optimized Feature Selection", *IEEE Access*, Vol. 11, pp. 3511–3524, 2023.

[10] Johann Faouzi, Olivier Colliot, Jean Christophe Corvol, "Machine Learning for Parkinson's Disease and Related Disorders", *Machine Learning for Brain Disorders, Springer*, Chapter No. 26, pp. 847-877, 2023.

[11] M. Sharmila Begum, A. Balajee, S. Kulothungan, D. Santhakumar, et. al., "Parkinson's Disease Prediction and Drug Personalization using Machine Learning Techniques", *Soft Computing, Springer*, Vol. 27, pp. 12669–12675, 2023.

[12] Linlin Yuan, Yao Liu, Hsuan-Ming Feng, "Parkinson Disease Prediction using Machine Learning based Features from Speech Signal", *Service Oriented Computing and Applications, Springer*, Vol. 18, pp. 101–107, 2024.

[13] Shawki Saleh, Bouchaib Cherradi, Oussama El Gannour, et. al., "Predicting Patients with Parkinson's Disease using Machine Learning and Ensemble Voting Technique", *Multimedia Tools and Applications, Springer*, Vol. 83, pp. 33207-33234, 2023.

[14] Saravanan Srinivasan, Parthasarathy Ramadass, Sandeep Kumar Mathivanan, et. al., "Detection of Parkinson Disease using Multiclass Machine Learning Approach", *Scientific Reports, Springer*, Vol. 14, No. 13813, pp. 01–17, 2024.

[15] Sawera Tariq, Madiha Qadeer, Attique Ur Rehman, et. al.,"An Intelligent Technique for The Effective Prediction of Parkinson Disease", *Proc. of the IEEE International Conference on Emerging Trends in Networks and Computer Communications*, Namibia, 2024.

[16] Nithiyameenatchi N., Sujatha R., Konguvel E., Adithya Narayan, et. al.,"Analysis of Parkinson's and Alzheimers Disease Using Various Machine Learning Based Algorithm", *Proc. of the IEEE* 3^{rd} *International Conference on Artificial Intelligence For Internet of Things*, Vellore, India, 2024.

[17] C. R. Dhivyaa, K. Nithya, S. Anbukkarasi, et. al.,"Enhancing Parkinson's Disease Detection and Diagnosis: A Survey of Integrative Approaches Across Diverse Modalities", *IEEE Access*, Vol. 12, pp. 158999–159024, 2024.

[18] S. Saravanan, Kannan Ramkumar, K. Narasimhan, et. al.,"Explainable Artificial Intelligence (EXAI) Models for Early Prediction of Parkinson's Disease Based on Spiral and Wave Drawings", *IEEE Access*, Vol. 11, pp. 68366–68378, 2023.

[19] Beslin Pajila, Gracelin Sheena, Gayathri, et. al.,"A Comprehensive Survey on Naive Bayes Algorithm: Advantages, Limitations and Applications", *Proc. of the IEEE* 4^{th} *International Conference on Smart Electronics and Communication*, Trichy, India, 2023.

[20] Mostafa Langarizadeh, Fateme Moghbeli, "Applying Naive Bayesian Networks to Disease Prediction: A Systematic Review", *Journal of Academy of Medical Sciences of B & H, Acta Informatica Medica*, Vol. 24, No. 05, pp. 364–369, 2016.

[21] Kristina Polotskaya, Carlos S. Muñoz Valencia, Alejandro Rabasa, et. al., "Bayesian Networks for the Diagnosis and Prognosis of Diseases: A Scoping Review", *Machine Learning and Knowledge Extraction, MDPI*, Vol. 24, No. 06, pp. 1243–1262, 2024.

[22] Zonghan Wu, Shirui Pan, Fengwen Chen, et. al., "A Comprehensive Survey on Graph Neural Networks", *IEEE Transactions on Neural Networks and Learning Systems*, Vol. 32, pp. 04–24, 2020.

[23] Xiangyan Meng, Tonghui Zou, "Clinical Applications of Graph Neural Networks in Computational Histopathology: A Review", *Computers in Biology and Medicine, Elsevier*, Vol. 32, pp. 01–31, 2023.

[24] Xiao Meng Zhang, Li Liangu, et. al., "Graph Neural Networks and Their Current Applications in Bioinformatics", *Frontiers in Genetics*, Vol. 12, pp. 01–22, 2021.

[25] David Ahmedt Aristizabal, Mohammad Ali Armin, Simon Denman, et. al., "Graph based Deep Learning for Medical Diagnosis and Analysis: Past, Present and Future", *Sensors, MDPI*, Vol. 21, No. 14, pp. 01–48, 2021.

[26] Norshakirah Aziz, Emelia Akashah Patah Akhir, Izzatdin Abdul Aziz, et. al., "A Study on Gradient Boosting Algorithms for Development of AI Monitoring and Prediction Systems", *Proc. of the IEEE International Conference on Computational Intelligence*, Malaysia, 2020.

[27] Farzana Z. Ali, Kenneth Wengler, Xiang He, et. al., "Gradient Boosting Decision Tree based Algorithm with Neuroimaging for Personalized Treatment in Depression", *Neuroscience Informatics, Elsevier*, Vol. 02, No. 04, pp. 01–06, 2022.

[28] A. Y. Yildiz, Asli Kalayci, "Gradient Boosting Decision Trees on Medical Diagnosis over Tabular Data", *Computer Science & Machine Learning, Cornell University*, pp. 01–08, 2024.

[29] Supreet Hiremath, Tarun S. J., Gururaj S. Kori, et. al.,"Effective Prognosis of Heart Attack Using Supervised Machine Learning Algorithms", *Proc. of the IEEE* 1^{st} *International Conference on Software, Systems and Information Technology*, India, 2024.