International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Performance Analysis and Evaluation of Supervised Machine Learning Algorithms for the Detection of Parkinson Disease

Sani Isiyaku[1], Sani Abba[2], S. Boukari[3]

[1,2,3]Department of Mathematical Sciences, Abubakar Tafawa Balewa University, Bauchi, Nigeria

ABSTRACT

. Disease detection and prediction has been one of the often uses of machine learning models in the last decade, as a result, Parkinson's disease would not be an exception. It is common for people in old age to have one form of disease or the other, relating to the circumstances they found themselves. This thesis research work performed analysis and evaluation of various supervised machine learning models to detect the presence of Parkinson's disease in aged people. An extreme gradient boosting model along with other models such as the Naïve Bayes (NB), Random Forest (RF), K-Nearest Neighbor (KNN), and Support Vector Machine (SVM) was implemented to determine which would give the best outcome. eXtreme Gradient Boosting (XGBoost) model is regarded as a powerful boosting machine learning technique, outstandingly in terms of speed and accuracy. Results obtained showed that the XGBoost model which is an optimized version of the other boosting models is capable of projecting the presence of Parkinson's disease with a very high accuracy compared to other algorithms. The implemented classification algorithm extreme gradient boosting method achieved approximately 98% accuracy, 97.5% precision, 100.0% recall, and 98.7% f1-score which is of high performance compared to the results found in the existing literature.

Keywords: XGBoost model; Parkinson Disease; K-Nearest Neighbor (KNN); Support Vector Machine (SVM); Machine Learning Models.

1. Introduction

Parkinson's disease, a neurological condition, causes tremors, stiffness, and coordination difficulties (Chintalapudi & Battineni, 2022). Early detection is crucial for effective treatment, reducing costs, and saving lives. Current methods often identify the disease in advanced stages, affecting dopamine levels in the basal ganglia. With an annual incidence of 10 to 18 per hundred thousand persons, Parkinson's is the second most common neurodegenerative condition after Alzheimer's (Hoq et al., 2021). Age is a major risk factor, with symptoms such as walking and speaking difficulties worsening over time (Borzi et al., 2021). Mental, behavioral changes, sleep issues, depression, and memory problems also accompany the disease (Kurmi et al., 2022). Parkinson's affects both genders, with age being a significant risk factor. While most cases develop around age fifty-five, five to ten percent occur before fifty. Hereditary factors and specific genetic mutations play a role in the disease (Hoq et al., 2021). Key symptoms include limb and trunk stiffness, trembling, slow movements, balance issues, depression, eating difficulties, bladder problems, skin issues, and sleep disruptions (Borzi et al., 2021). Existing studies fall short of a 96% detection rate, motivating the development of an improved Parkinson's detection model using advanced gradient boost algorithms.

2.0 Statement of the Problem

Remote monitoring of Parkinson's disease through vocal tasks has shown high accuracy in trials. However, current studies often rely on limited voice recordings in controlled settings, hindering scalability (Pramanik et al., 2021). Parkinson's, a progressive neurological condition, lacks a definitive diagnostic method, complicating early detection (Aich et al., 2020). Machine learning algorithms improve disease prediction precision, particularly for late-stage onset in individuals above sixty (Pramanik et al., 2021). Early identification is crucial, as delayed detection leads to chronic conditions in the elderly (Fujita et al., 2021). Challenges in obtaining accurate data from Parkinson's patients, due to age and individual variations, led to the development of technology for medication monitoring using supervised machine learning. Despite advancements, Parkinson's detection accuracy is not perfect, prompting this study to enhance it with an extreme gradient boosting algorithm (Chintalapudi & Battineni, 2022).

3. Aim and Objectives

The aim is to develop an XGBoost machine learning model for precise detection of Parkinson's disease, with the following objectives:

i. Create a machine learning framework utilizing boosting algorithms, specifically XGBoost, for the accurate detection of Parkinson's disease.

ii. Investigate and emphasize the impact of Parkinson's disease (PD) on supervised machine learning models employing boosting techniques.

iii. Assess the performance of the models in Parkinson's disease detection using evaluation metrics, including those from Chintalapudi & Battineni (2022) and others like Hoq et al. (2021).

4.0 Brief Literature Review

There are several kinds of research found in the literature that was carried out to detect Parkinson's disease using various techniques; these include machine learning, deep learning, and so on but among all of them applied to the existing research, boosting methods were least explored in the detection of Parkinson’s disease. (Fujita et al., 2021) conducted a study that detects Parkinson’s disease (PD) through the construction of a detection model that combines Recurrent Neural Network (RNN) and Convolutional Neural Network (CNN), the study focuses on the fact that patients with Parkinson’s disease (PD) have motor speech disorders.

The study conducted compared the performance of the RNN to that of the long short-term memory (LSTM) and gated recurrent unit (GRU) with conventional gates (Hoq et al., 2021). According to their study, they used graphene-based biosensors to detect dopamine for effective Parkinson’s disease diagnostics. The biosensors are being turned into smartphone-connected disease management systems. However, they recommended that emphasized that clinical utility should take precedence over analytical and technical performance.

In a similar study by (Hoq et al., 2021), for the detection of PD using machine learning classifiers. The probing results showed that the proposed Sparse Autoencoder - Support Vector Machine (SAE-SVM) model outperformed not only the previous model of the Principal Component Analysis - Support Vector Machine (PCA-SVM) and other standard models such as Multilayer Perceptron (MLP), Extreme Gradient Boosting (XGBoost), K-Nearest Neighbor (KNN), and Random Forest (RF), but also two recent studies using the same dataset according to (Hoq et al., 2021).

To detect the presence of Parkinson’s disease, vocal tasks for remote detection and monitoring of Parkinson's disease are highly accurate in numerous trials (of PD). Most of these studies, on the other hand, publish results based on a small number of voice recordings, which are often taken under acoustically controlled conditions, and so cannot be scaled up without specialized equipment (Chintalapudi et al., 2021).

Finding accurate biomarkers that allow the development of medical-decision support tools is now a major research effort in healthcare biometrics. These instruments aid in the detection and monitoring of disorders such as Parkinson's disease (Chintalapudi et al., 2019).

Using a convolutional neural network (CNN) to diagnose Parkinson's disease (PD) from drawing movements. Feature extraction and classification are the two aspects of this CNN. The greatest results obtained in their study were 96.5 percent accuracy, 97.7 percent F1-score, and 99.2 percent area under the curve (Byeon, 2020).

The difficulty in getting accurate data from Parkinson’s disease patients may be because of old age and characteristics fluctuation from one patient to another. Technology was designed that could automatically monitor the medication state of patients using the supervised techniques of machine learning such as the random forest, support vector machine, k-nearest neighbor, and naïve Bayes. Random forest outperformed other models in performance according to (Ozkan, 2016). Patients with Parkinson's disease (PD) are more likely to experience fluctuations in their motor symptoms. This unavoidable trait may have an impact on the sufferers' quality of life. However, utilizing self-reported data from PD, it is challenging to gather precise information on the fluctuation features. Based on the outcome of the research (Aich et al., 2020) concluded that the possibility of employing inertial sensors incorporated in commercial smartphones was proven in this work, which also presented a simple technique for accurate postural instability grading. This tool can be used to detect early indicators of Parkinson's disease, monitor the disease, and prevent falls. In their study, (Aljalal et al., 2022) proposed an automated Parkinson's disease detection using deep learning. For the study, EEG recordings from 16 healthy controls and 15 PD patients were used. EEG recordings were transformed into spectrograms using the Gabor transform, which was then utilized to train the proposed two-dimensional convolutional neural network (2D-CNN) model. The proposed model achieved a high classification accuracy. However, some diseases would have made great damage already before they were discovered.

5.0 METHODOLOGY

5.1 Analysis of the existing model

To assess the efficacy, loss, precision, recall, and F1-score of these three ML systems in Parkinson's disease (PD) identification, the study (Chintalapudi & Battineni, 2022) divided their dataset into training and testing groups. The number of samples from each target class in each subset of this unbalanced dataset has been determined using class labels. Each subset in a stratified design has roughly the same number of samples from each target class as the entire dataset. 67% of the data were used for training, while 33% were used for testing.

By contrasting the effectiveness of the suggested algorithm in conjunction with the four aforementioned classifiers, 20 PD subjects with definite motor fluctuations have been analyzed in total. It was discovered that random forest performed better than the other classifiers, with an accuracy, recall, and precision of 96.72%, 97.35%, and 96.20%, respectively. Patients with Parkinson's disease (PD) are more likely to experience fluctuations in their motor symptoms. This unavoidable trait may have an impact on the sufferers' quality of life. However, utilizing self-reported data from PD, it is challenging to gather precise information on the fluctuation features (Aich et al., 2020).

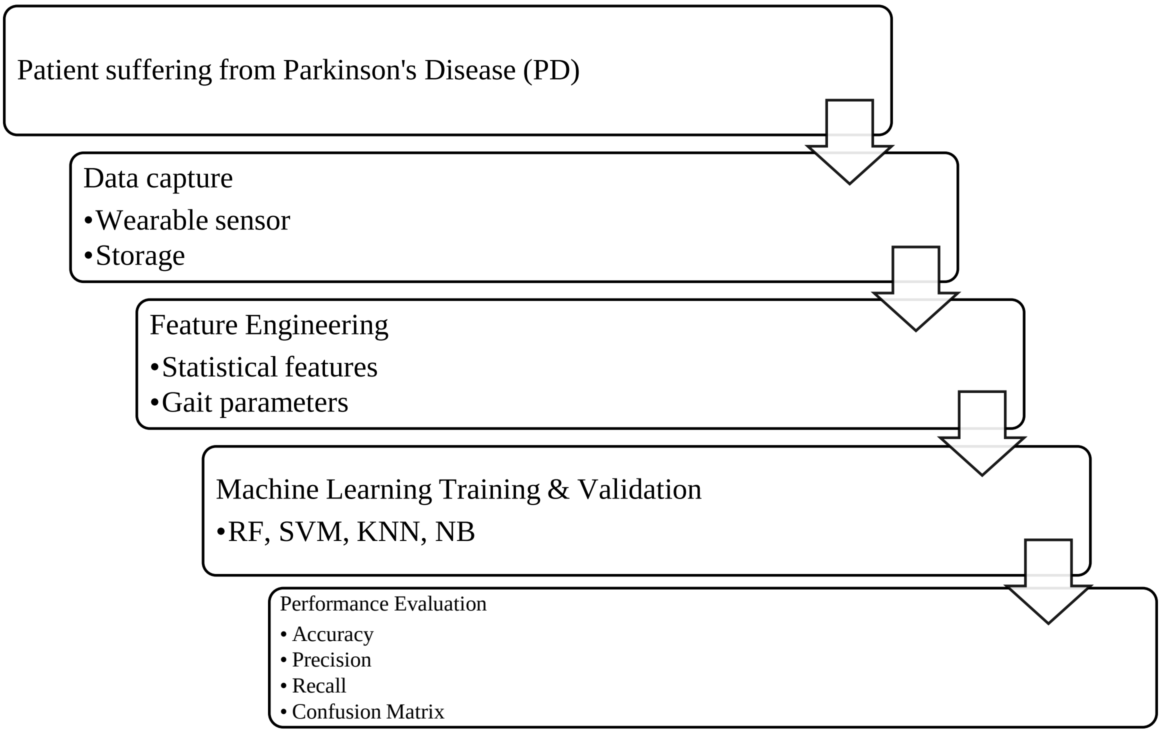

Figure 1: Architecture of the existing model *Source: (Aich et al., 2020)*

The test dataset is used to predict cases with and without Parkinson's disease (PD) using the trained model after the training dataset has been used to train the machine learning (ML) model. With cross-entropy as a loss, deep learning models were trained in tiny batches of 16 samples using the stochastic gradient descent (SGD) technique. Despite the overwhelming indication supporting the need to detect Parkinson's disease at an early stage still, the disease has continued to be of greatest concern globally because it affects the important organs of humans which is brain disorder and forgetfulness.

5.2 Weakness of the existing system

As earlier pointed out, Parkinson's disease (PD) is a neurological condition that worsens over time and is marked by tremors, stiffness, and rigidity in addition to motor impairment. Some Parkinsonians also exhibit non-motor symptoms, such as hyposmia, constipation, urinary dysfunction, orthostatic hypotension, memory loss, depression, pain, and sleep difficulties, in addition to the characteristic motor symptomatology. Since there is no established, objective method for diagnosing PD, it cannot be done easily. The precision of disease predictions has increased since machine learning (ML) algorithms have been used in medical diagnoses. It is a late-stage disease with no noticeable symptoms at first instance, mostly found in elderly people above sixty years of age (Pramanik et al., 2021). Untimely detection of Parkinson's disease among aged persons often results in a chronic condition of the victims (Fujita et al., 2021). The accuracy of Parkinson's disease identification is not perfect.

5.3 The Proposed Model

The figure below described how the research will be conducted. The research work is expected to use different datasets to determine the best-performing model and for ease of evaluation and reporting of results. The datasets were downloaded from the UCI machine learning repository and Kaggle repository. Data manipulation techniques such as data preparation and feature selection were performed. Subsequently, the dataset will be divided into two; training and testing. Models built will be subjected to the training data to learn from its pattern and evaluated on the testing data to determine accuracy, precision, recall, and f1-score. Therefore, this study would deploy a relatively new method for improved detection of the disease using the eXtreme Gradient Boosting algorithm.

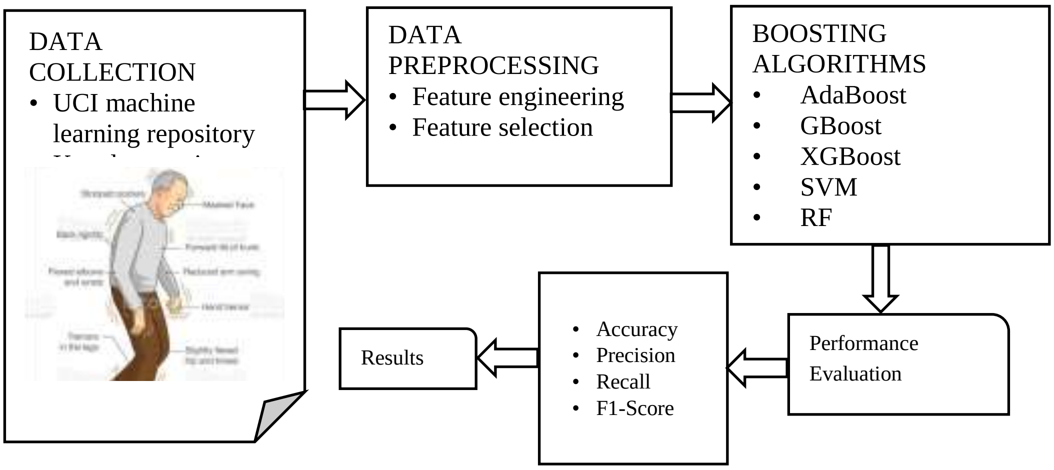

Figure 2: Block Diagram of the Proposed Model Framework

5.4 Working principle of the model

The principle of the proposed machine learning boosting models as described in Figure 1 above; states all the steps needed to achieve the desired improved or boosted prediction performance of Parkinson's disease. Two different birthplaces of PD datasets were downloaded and loaded. The datasets have undergone preprocessing techniques. After working on the data, the dataset will be divided into two; training and testing. The developed models built will be subjected to the training data to learn from it and evaluated on the dataset reserved for testing. The results obtained will be evaluated on performance evaluation metrics for classification to further determine and investigate hidden results.

5.5. Parkinson's Disease Dataset

The Parkinson's disease dataset utilized in this investigation was obtained from publicly available sources; thus including, the UCI Machine Learning and Kaggle repositories. This dataset contains a variety of biological voice measurements from 31 individuals, 23 of whom have Parkinson's disease (PD). Each column in the table refers to a specific voice measure, and each row corresponds to one of the 195 voice recordings made by these people. The primary goal of the data is to distinguish healthy persons from those with Parkinson's disease using the "status" column, which is set to 0 for healthy people and 1 for those with PD (Gil-mart & Montero, 2019).

Additionally, the data for this study came from 188 individuals with Parkinson's disease at Istanbul University's Cariappa Faculty of Medicine, with ages ranging from 33 to 87. The control group is made up of 64 healthy people (including men and women) ranging in age from 41 to 82. The microphone was adjusted to 44.1 KHz during the data collection process, and each subject's sustained phonation of the vowel /a/ was taken three times after the physician's examination.

5.5.1 Data Pre-processing

Data preprocessing entails converting raw data into well-formed data sets to use data mining methods (Aljalal et al., 2021). Raw data is frequently incomplete and formatted inconsistently. The success of every project involving data analytics is directly related to the adequacy or inadequacy of data preparation. Data validation and data imputation are both parts of the preprocessing process. Data validation aims to determine whether the data is comprehensive and accurate. The purpose of data imputation is to fill in missing numbers and fix errors (Mei et al., 2021).

5.5.2Feature Extraction

As the name suggests, in the feature extraction technique, one can extract and create new features that are in a linear mixture of the current dataset features. When compared to the original feature values, the new set of features will have different values. The fundamental goal is to require fewer features to gather the same amount of data. We would believe that selecting fewer features will result in underfitting, but in the case of the Feature Extraction approach, the excess data is usually noise (Demir et al., 2022).

5.6 Forecasting Tools and Methods

Python programming language will be used throughout because it is a language that is generally considered for machine learning and artificial intelligence. It has thousands of packages that can be used to execute programs. For example, Scikit Learns, pandas, numpy, matplotlib, and so on could be leveraged for this research (Demir et al, 2022).

5.6.1 Python

It is a scripting language that is high-level, interpreted, interactive, and object-oriented. Python is intended to be a very understandable language. It usually employs English vocabulary, whereas other languages rely on punctuation. In comparison to other languages, it features fewer syntactical structures. Python is a scripting language that is interpreted, interactive, object-oriented, and simple to use (python.org).

5.6.2 Sci-kit learn

Built on Numerical Python (NumPy), SciPy, and Matplotlib, libraries provide simple and efficient tools for predictive and investigative data analysis that are accessible to everyone and reusable in a variety of situations. They are open-source and commercially useable under the BSD license (scikit-learn.org).

5.6.3 Numerical Python (NumPy)

Numerical Python or for short NumPy (https://numpy.org) has a wide range of mathematical functions, as well as random number generators, linear algebra algorithms, and Fourier transforms. NumPy works well with distributed, GPU, and sparse array libraries and supports a wide range of hardware and processing systems.

3.6.4 Pandas

It is a fast, powerful, flexible, and easy-to-use open-source data analysis and manipulation tool, built on top of the Python programming language (pandas.pydata.org).

5.6.5 Jupyter Notebook

This tool called Jupyter notebook is one of the recent interactive development environments for project development and implementation for notebooks, code, and data that can be accessed over the web. It provides an easy way for users to be able to create and arrange workflows in data science, scientific computing, computational journalism, and machine learning using its versatile interface. Extensions to enhance and enrich functionality are encouraged by a modular architecture as indicated on their page (https://jupyter.org).

5.6.6 Colab notebooks

It is used to execute code on Google's cloud servers, more importantly, the power of the Google hardware, including GPUs and TPUs can be leveraged, regardless of the power of your machine being used (colab.research.google.com).

5.7 System Requirement

The study would be conducted on a specific computer with Intel Dual Core(R) Pentium(R), CPU at 16.64GHz, RAM 16.00 GB, and x64-based processor. Windows 10 operating system version to be used.

5.8 Performance Evaluation

It is superlative to evaluate the result of the model to determine whether it is what is needed and that it is accurate and could be deployed to serve the purpose for which it was developed. Any machine learning model's ultimate goal is to learn from examples in such a way that the model can apply what it has learned to new situations it has not encountered before. At the most basic level, you should train on a portion of the complete dataset, leaving the rest for evaluation to determine the model's capacity to generalize - in other words, how well will the model perform on data that it has not directly learned from during training.

5.8.1 Choice of Evaluation Metrics

The detection problem is a classification problem, therefore to evaluate the performance of the proposed model; we are going to use a confusion matrix. It is made up of accuracy, precision, recall, and f1 scores will be used. These evaluation metrics were used to evaluate Parkinson's disease detection models (Pardoel et al., 2019). It is calculated as shown below:

5.8.2 Confusion Matrix

A confusion matrix is a summary of classification problem prediction outcomes. (Kurmi et al., 2022) The number of rights and unsuccessful predictions is totaled and broken down by class using count values. The confusion matrix depicts the various ways in which your classification model becomes perplexed when making predictions. It informs you not only about the faults made by your classifier but also about the types of errors that are being made. This breakdown addresses the drawback of relying solely on categorization accuracy. The number of rights and unsuccessful predictions is totaled and broken down by class using count values. The confusion matrix depicts the various ways in which your detection or classification model becomes perplexed when making predictions. It informs you not only about the errors made by your classifier but also about the types of errors that are being made (Borzì et al., 2020).

Event	No-event	
Event	True Positive (TP)	False Positive (FP)
No-event	False Negative (TN)	True Negative (TN)

5.8.2.1 True Positive (TP)

These are successfully predicted positive values, indicating that the value of the real class is yes, as well as the value of the anticipated class. For example, if the actual class value indicates that this passenger survived and the anticipated class also suggests that this passenger survived (Chintalapudi & Battineni, 2022).

5.8.2.2 False Positive (FP)

When the expected class is yes and the actual class is no. For example, if the actual class indicates that this passenger did not survive, but the forecasting class indicates that this passenger will (Pianpanit et al., 2021).

5.8.2.3 False Negative (TN)

When the actual class is positive but the anticipated class is negative. For example, if the passenger's actual class value indicates that he or she survived, the predicted class value implies that the person would die. We can calculate Accuracy, Precision, Recall, and F1 score once we understand these four characteristics (Demir et al., 2022).

5.8.2.4 True Negative (TN)

These are accurately predicted negative values, indicating that the value of the real class is zero and the value of the projected class is zero as well. For example, if the real class states the passenger did not survive and the forecasting class says the same. False positives and false negatives happen when your actual class differs from the projected class (Kurmi et al., 2022).

5.9 Accuracy

The simplest intuitive performance metric is accuracy, which is just the ratio of properly predicted observations to all observations. One would believe that if our model is accurate, it is the best. Yes, accuracy is a useful statistic, but only when the datasets are symmetric and the values of false positives and false negatives are almost equal. As a result, other parameters must be considered while evaluating your model’s performance (Hoq et al., 2021).

$$Accuracy = \frac{tp + tn}{tp + tn + fp + fn} \quad - \quad - \quad - \qquad (1)$$

5.10 Precision

The ratio of accurately predicted positive observations to total expected positive observations is known as precision. The question that this measure answer is how many of the passengers who were identified as having survived did. The low false positive rate is related to high precision (Kurmi et al., 2022).

$$Precision = \frac{tp}{tp + fp} \quad - \quad - \quad - \qquad (2)$$

5.11 Recall

Yes, recall is defined as the proportion of accurately predicted positive observations to all observations in the class. Many of the passengers who truly survived were labeled, according to the question. The recall is also known as sensitivity (Chintalapudi et al., 2021).

$$Recall = \frac{tp}{tp + fn} \qquad - \quad - \quad - \qquad (3)$$

5.12 F1-Score

Precision and Recall are weighted and averaged. As a result, this score considers both false positives and false negatives. Although it is not as intuitive as accuracy, F1 is frequently more useful than accuracy, especially if the class distribution is unequal. When false positives and false negatives have equivalent costs, accuracy works well. It's best to look at both Precision and Recall if the cost of false positives and false negatives is considerably different (Hoq et al., 2021).

$$F1 - Score = 2\left(\frac{precision * recall}{precision + recall}\right) - \quad - \quad - \qquad (4)$$

Where the meaning of the parameters in equations (1, 2, 3, and 4) is as follows:

tp = True Positive

tn = True Negative

fp = False Positive

f = False Negative

6.0 ANALYSIS AND DISCUSSION

In this section, we categorically explained the stages involved from acquiring data to the experimental procedure and the presentation of results. Data manipulation methods used in this work were also stated. The evaluation, analysis, and discussion of the results obtained were also included in this section.

6.1 Data Acquisition and Experimental Procedure

6.1.1. Dataset

A dataset is a collection of raw statistics and data gathered during a research project (Williamson et al., 2021). A dataset is central to study work like this because it tells us the history and behavior of certain quantities, measurements, and diseases. Parkinson's disease dataset has been generated and found on a popular repository for machine learning research (Pramanik et al., 2021). The dataset used here is described below.

6.1.2. Parkinson's Data Set

The dataset used for this research work was collected from the two popular machine learning repositories, UCI Machine Learning repository, and Kaggle, https://archive.ics.uci.edu/ml/datasets and https://www.kaggle.com/datasets respectively. The Max Little of the University of Oxford generated the dataset in partnership with the National Center for Voice and Speech in Denver, Colorado, which recorded the speech signals. The data is downloaded in comma-separated value (CSV) format.

6.1.3. Dataset Information

This dataset contains a variety of biological voice measurements from thirty-one (31) individuals, twenty-three (23) of whom have Parkinson's disease (PD). Each column in the table refers to a specific voice measure, and each row corresponds to one of the one hundred and ninety-five (195) voice recordings made by these persons. The primary goal of the data is to distinguish healthy persons from those with Parkinson's disease using the "status" column, which is set to 0 for healthy people and 1 for those with PD (Gil-mart & Montero, 2019).

Table 1: Parkinson's Disease Dataset Features and Descriptions of the Original Dataset.

FEATURES	DEFINITIONS
MDVP: Fo (Hz)	Average vocal fundamental frequency
MDVP: Fhi (Hz)	Max. vocal fundamental frequency
MDVP: Flo (Hz)	Min. vocal fundamental frequency
MDVP: Jitter (%)	Jitter as a percentage
MDVP: Jitter (Abs)	Absolute jitter in microseconds
MDVP: RAP	Relative amplitude perturbation
MDVP: PPQ5	Five-point period perturbation quotient
MDVP: Shimmer	local shimmer
MDVP: Shimmer (dB)	the local shimmer in decibels

MDVP: APQ	11-point amplitude perturbation quotient
Shimmer: APQ3	Three-point amplitude perturbation quotient
Shimmer: DDA	The average absolute difference between consecutive differences between the amplitudes of consecutive periods
Shimmer: APQ5	Five-point amplitude perturbation quotient
Jitter: DDP	Average absolute difference of differences between cycles, divided by the average period
NHR	Noise-to-harmonics ratio
Status (1/0)	Active/Inactive
HNR	Harmonics-to-noise ratio
RPDE	Recurrence period density entropy
DFA	Signal fractal scaling exponent
D2	Correlation dimension
POPE	Pitch period entropy
Spread1	The fundamental frequency of two nonlinear actions
Spread2	Variant

Source: (Ozkan, 2016)

6.2. Experimental Setup Procedure

First of all, it was ensured that all necessary programs, tools, and techniques that would be needed to perform this experiment were downloaded and installed to obtain a good result. The Python libraries used for various operations and functions are also installed. These include NumPy which is used for numerical Python; pandas are used for data loading and analysis and are also acquired and installed to set the environment ready. Jupyter Notebook was used because it can present codes and data very well. Scikit-learn python machine learning library is used for designing, building, and developing machine learning classifiers like extreme gradient boost is also set in place.

6.2.1. Data gathering

This is the first step for most if not all detection machine learning projects. Data is the first thing we need to get or acquire to start developing and running machine learning projects. The quality of the data is central to obtaining a good and accurate detection at the end of the experiment. As a result, the data used in this work as stated above was obtained from a reputable machine learning repository which is known as the UCI machine learning repository. The data obtained is in comma-separated value (.csv) format.

4.2.2. Data Preparation

After importing the necessary supporting programs, files, and datasets for the research work. Data preparation is paramount which is used to set the data ready to go for a machine learning project. The data collected was loaded into the google colab notebook using a panda's command read data as follows.

```
#Import the data

from Google.colab import drive

drive. mount('/mntDrive')
```

It is common knowledge that most data collected or downloaded must be prepared or preprocessed to make it fit the proposed model to obtain an accurate and even dependable result. Therefore, the data obtained has undergone data preprocessing, feature extraction, and feature engineering as briefly explain below.

6.2.2.1. Data Preprocessing

This is a crucial step in a machine learning detection or classification project like this. It is a process that is used to enhance the quality dataset to promote the extraction of meaningful and accurate insights from the dataset. It is a tool and technique used in machine learning projects for cleaning and organizing data to make it suitable for building, training, and testing machine learning models (Alzubaidi et al., 2021). It is applied in this work to transform raw data into an understandable and readable format.

6.2.2.2. Feature extraction

This feature extraction step is a process of dimensionality reduction process by which an initial set of data is reduced by identifying only the most relevant key features (Hoq et al., 2021) from the dataset that affects the detection machine learning model. In this study, feature extraction was used to select necessary relevant features for Parkinson's disease detection.

6.2.2.3. Feature engineering

This feature is also used primarily to improve or enhance model detection performance accuracy. It is a technique of machine learning that leverages data to create new variables that are not in the training dataset (Aich et al., 2022). It generally produces new features for both types of machine learning projects, supervised and unsupervised learning. It is used for simplifying and increasing the speed of dataset transformation and manipulation aside from improving precision, recall, f1-score, and accuracy of model performance (Cocco et al., 2021).

6.2.3. Model application

There are a lot of already developed machine learning algorithms that could be out rightly adopted and apply them directly for detection or prediction purposes but for this study work, the extreme boost algorithm was developed to fit the proposed model for this work. Hence, the modified extreme boost algorithm model has been applied to the already preprocessed or prepared dataset for the detection of Parkinson's disease.

6.2.4. Train-Test Split

The bulk of machine learning algorithm research work used to be in training the model because the accuracy of the machine learning model mostly depends on the model training on the dataset. After the model has been developed and trained then testing is necessary to measure how accurate the detection performance of the model is at this stage, the dataset was divided into two sets, sixty-five and thirty-five percent; the first one for training and the former for testing, this was done to evaluate the model performance on the dataset that is not known to the model.

6.2.5. Performance Evaluation

It is superlative to evaluate the result of the model to determine whether it is what is needed and that it is accurate and could be deployed to serve the purpose for which it was developed. Any machine learning model's ultimate goal is to learn from examples in such a way that the model can apply what it's learned to new situations it hasn't encountered before. At the most basic level, you should train on a portion of the complete dataset, leaving the rest for evaluation to determine the model's capacity to generalize - in other words, how well will the model perform on data that it has not directly learned from during training (Milano et al., 2021).

6.2.6. Parameter tuning

Before building an XGBoost model, we must examine a variety of parameters and their values. Therefore, to increase and completely use the advantages of the XGBoost model over competing methods, parameter adjustment is required. Various parameter combination was performed and tested until a better performance is finally discovered.

6.3. Results

The evaluation parameters used to assess the machine learning algorithms were accuracy, average recall, average precision, and average f1 score. Correctness in this to comprehend the total accuracy in the performance, a specific task is used. The recall is determined for each class in the present scope of the task and indicates the number of data samples that the model properly identified for a given class.

The average percentage of accurate predictions (True Positive and True Negative) in each class in this study was computed using the average recall. Also, the accuracy of the forecasts has been calculated, which establishes confidence in a specific forecast. Moreover, average precision was computed for evaluation reasons, allowing us to determine the average of all precision ratings for each class. Finally, the f1-score was calculated, which offers a weighted average of each class's precision and recall. Penalizes the score for each sample in a class that was incorrectly predicted.

The generalizability of the generated model for incoming unseen data is another issue that needs to be addressed since it was employed in the production environment.

Table 3, below provides an overview of experimental outcomes. All the classifiers were run under the same platform and condition. Five models which include extreme gradient boosting, random forest, support vector machine, k-nearest neighbor, and naïve Bayes were applied. The result obtained is shown below with the extreme gradient boosting method having the highest performance accuracy and precision. It also has a recall of a hundred percent and f1-score. In essence, the boosting classifier has out rightly outperformed all other models used in the previous studies.

Table 2: A comparative analysis of multiple classifiers

Classifier	Accuracy	Precision	Recall	F1 Score
XGBoost	97.5	97	100	98.5
RF	96	96	97	97
SVM	93	93	92	93
KNN	86	85	84	85
NB	88	86	85	86

6.3.1 Data Description

The Google colab support various frameworks for data description and analysis, several pieces of information could be derived from the data. The data shape, information, correlation, and so on were explored to understand clearly the dependent is an independent variable and their relationships. Figure 3 below displayed the relationship between the various columns in the datasets. It is obtained using a Python library called seaborn. From the black color to the white color indicate a very low correlation to a very high correlation, in other words from minus one to plus one. The places with black color show no relationship at all whereas the cells with white color describe an excellent correlation.

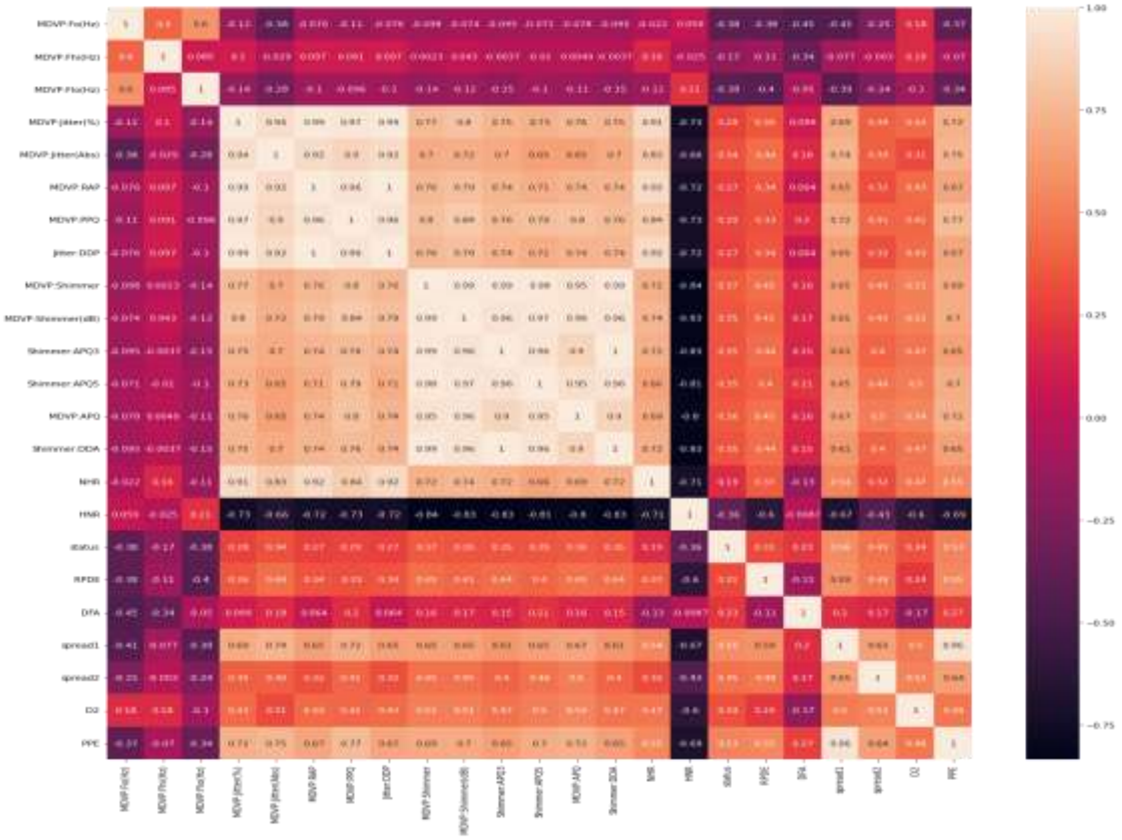

Figure 3: Correlation matrix of the dataset

In Figure 4, figure 5, and Figure 6, the line kind of plot was used to plot the actual versus the predicted values. The blue lines indicate the actual values of the dataset while the orange color represents the prediction results. Figure 4, naïve Bayes prediction results were compared with the actual data, hence the plot below. The line is not so much good because we can see the difference. For the random forest in Figure 5, the difference between the actual and the predicted is not quite clear as the one in naïve Bayes but it has some way to go to get the actual prediction.

The extreme gradient boost in Figure 6 as stated earlier outperformed all models in performance accuracy and other evaluation metrics. It has a nearly perfect prediction, the difference between the actual and prediction is very minimal hence needing some simple adjustment to achieve a perfect output. The two lines in this plot, figure 6 looks like one of the accuracies achieved by the model. Figures 7 and 8 display the actual and predicted performance of the support vector machine and k-nearest neighbor.

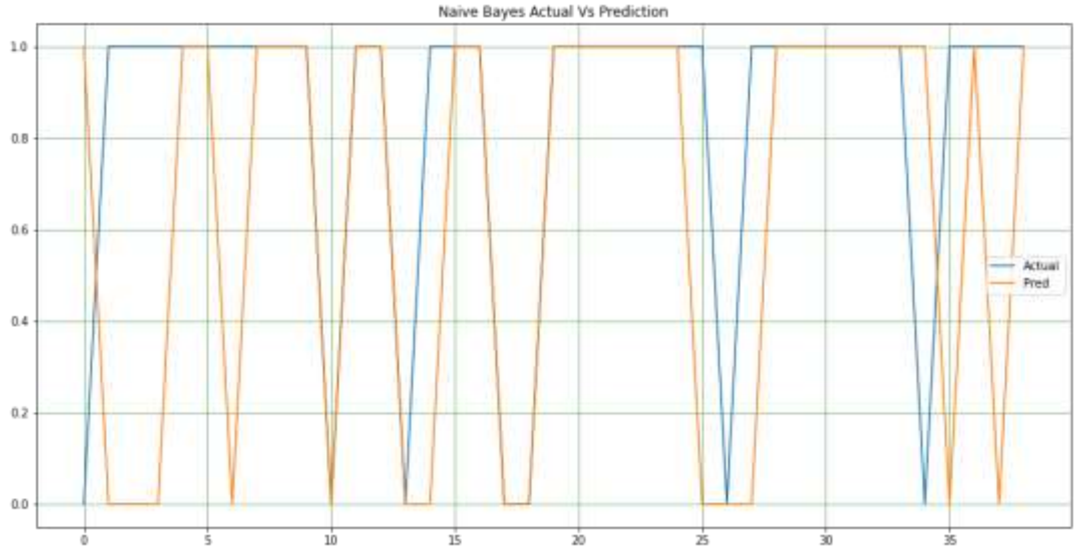

Figure 4: Naïve Bayes Actual Verses Predicted values

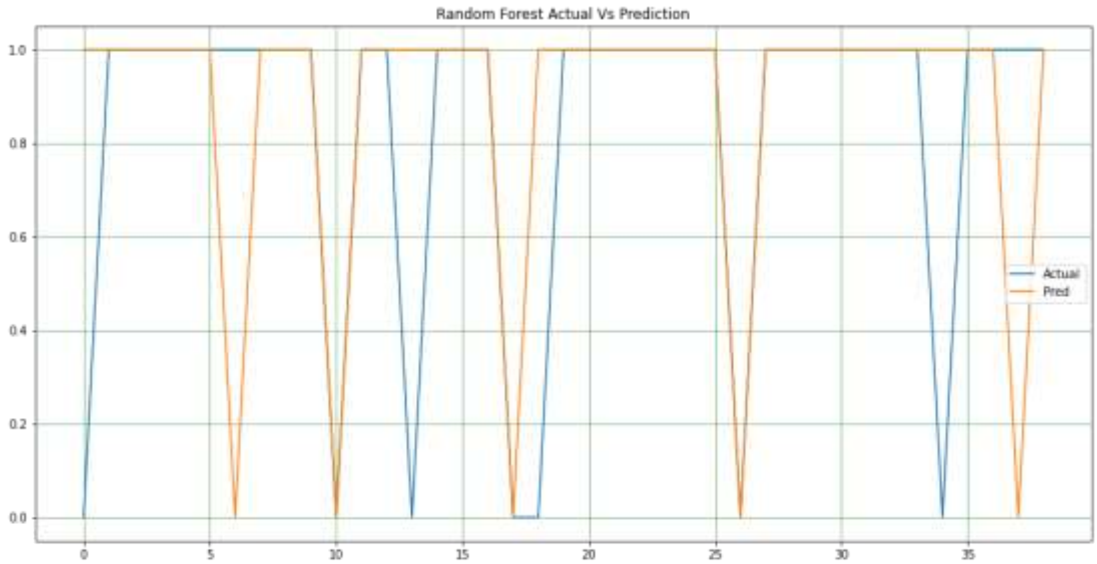

Figure 5: Random Forest Actual Verses Predicted values

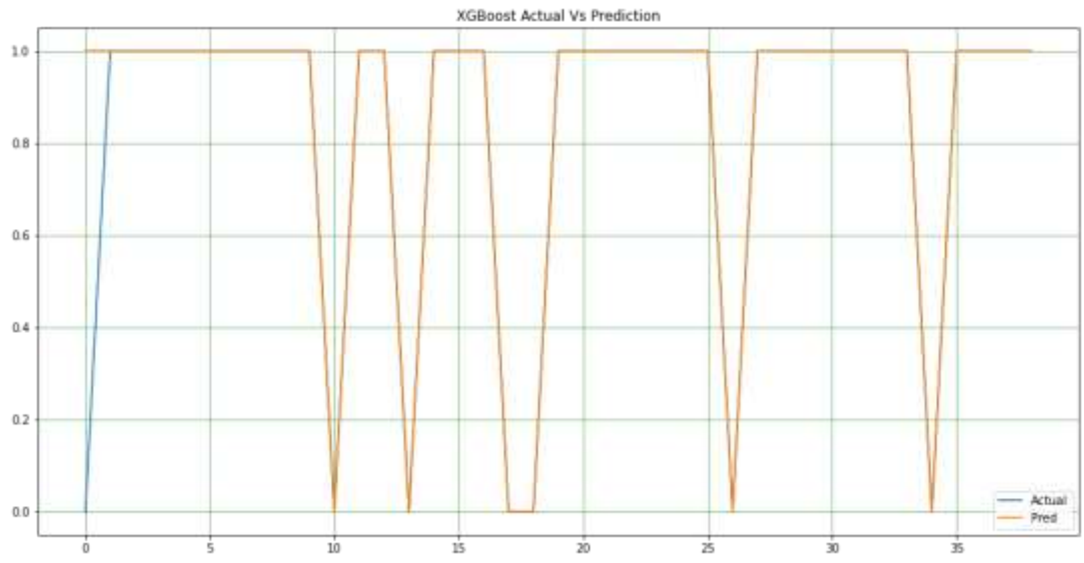

Figure 6: Extreme Gradient Boosting Actual Verses Predicted values

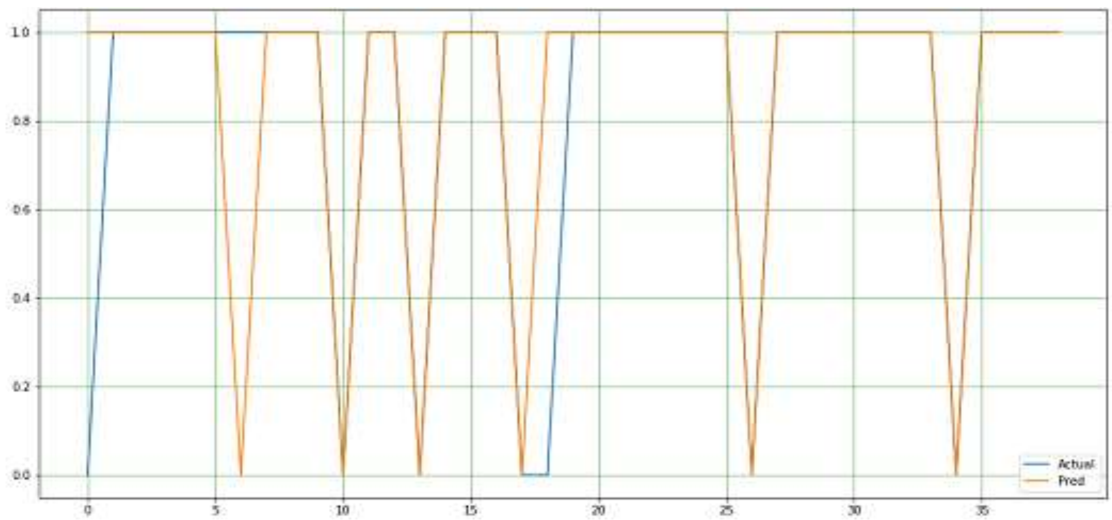

Figure 7: SVM Actual Verses Predicted values

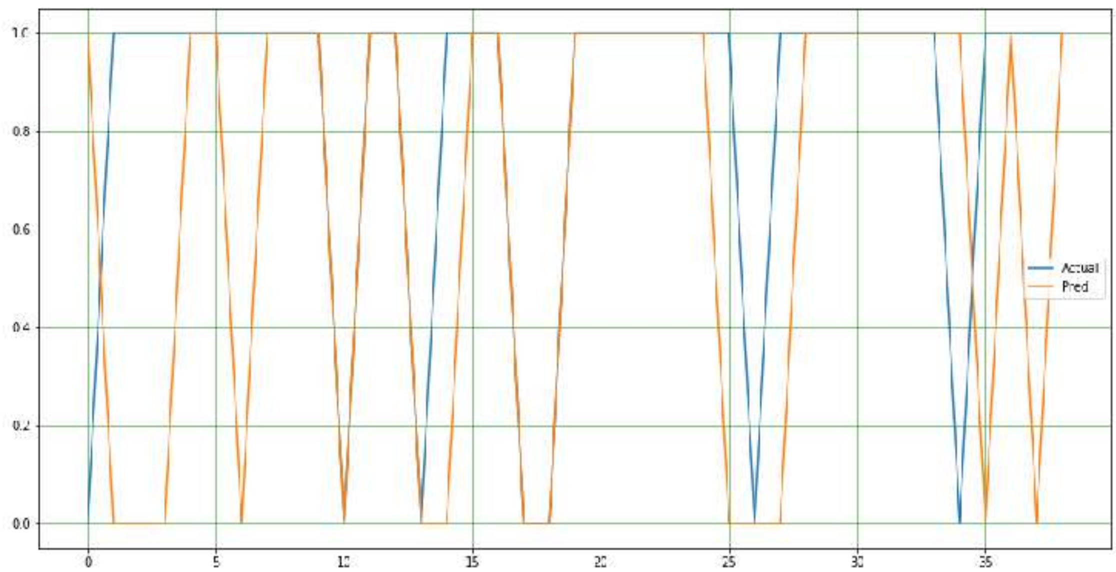

Figure 8: KNN Actual Verses Predicted values

In all the figures x-axis represents the predicted value and the y-axis represents the period. Figures 9, 10, 11, and 12 below present the outcome of the evaluation metrics used to further evaluate the performance of the models. As can be seen, our proposed model has outperformed all other models used in the previous study. In Figure 7, we have the highest accuracy score, precision also highest and hundred percent recall, and an excellent f1 score in Figure 10. The percentage of accurate predictions in each class in this study was computed using the average recall. Also, the accuracy of the forecasts has been calculated, which establishes confidence in a specific forecast. Moreover, average precision was computed for evaluation reasons, allowing us to determine the average of all precision ratings for each class. Finally, the f1-score was calculated, which offers a weighted average of each class's precision and recall. Penalizes the score for each sample in a class that was incorrectly predicted.

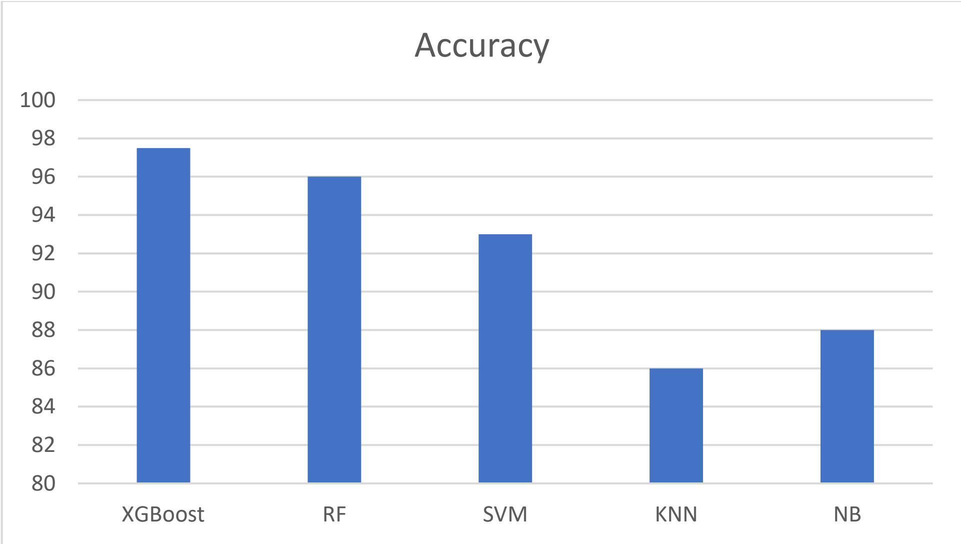

Figure 9: Accuracy

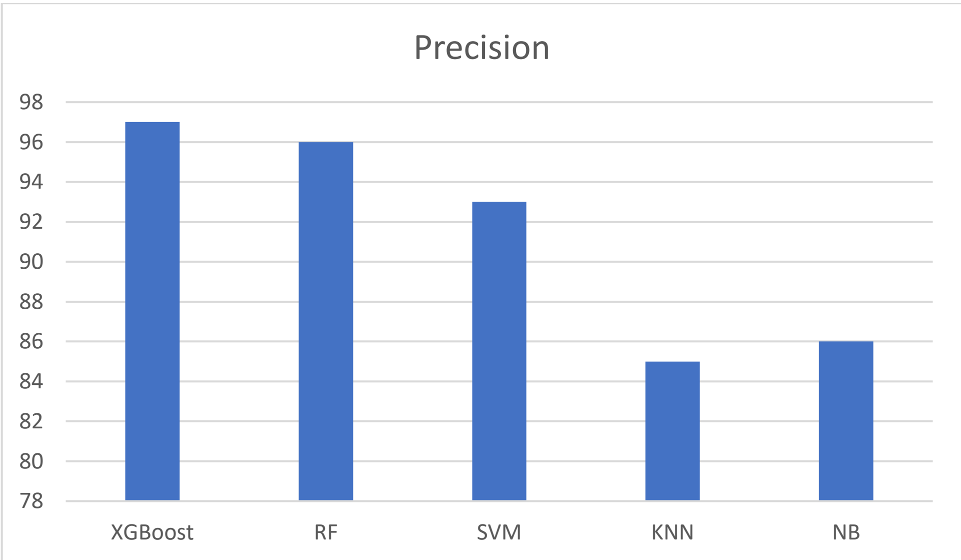

Figure 10: Precision

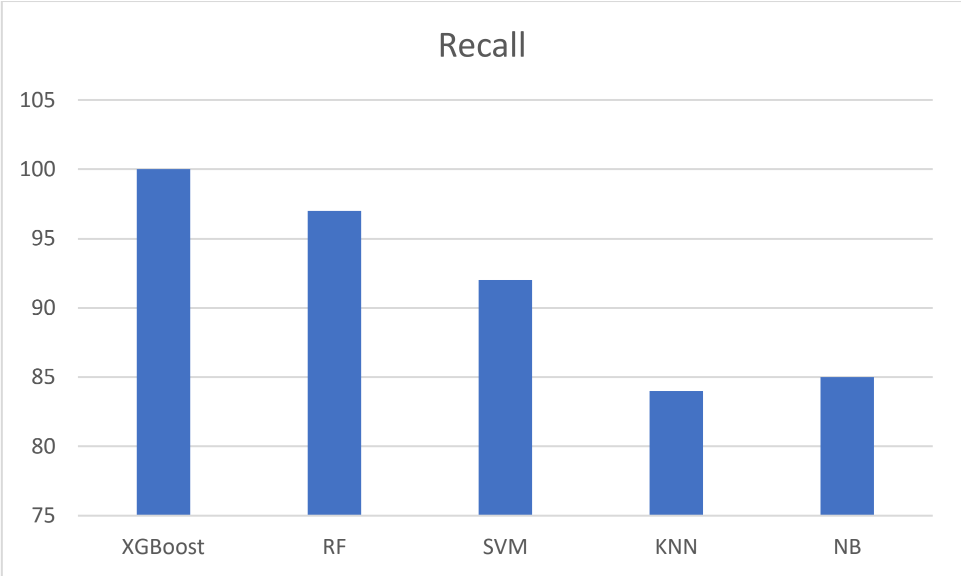

Figure 11: Recall

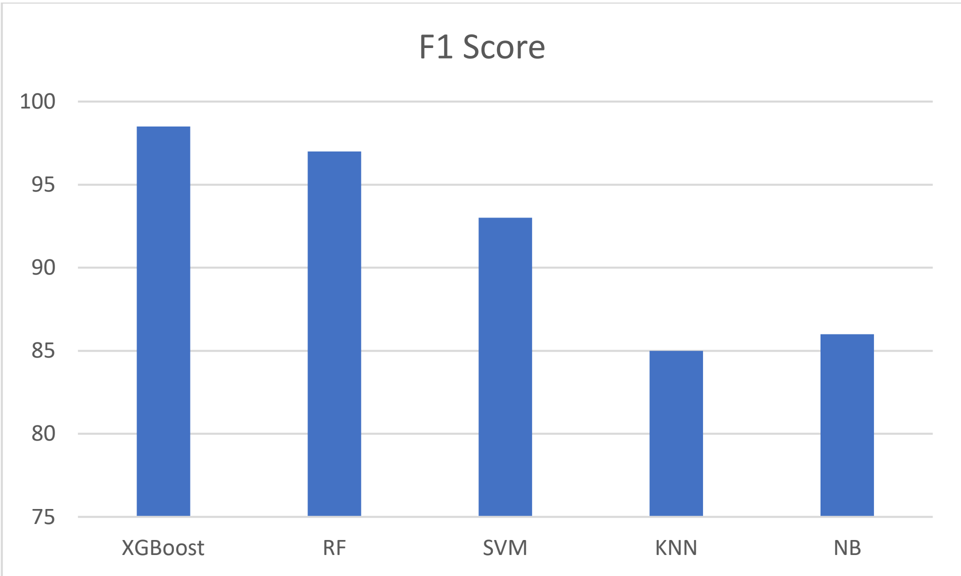

Figure 12: F1 Score

6.3.2 Kaggle Parkinson's Disease Dataset

To further evaluate the performance of the model, a similar dataset was obtained from the Kaggle machine learning dataset repository to ensure the accuracy of the detection model. The same experimental procedure was followed like the UCI dataset. Even though the algorithms do not perform better on this dataset despite following the process of pre-processing and feature extraction. The models perform better on the UCI dataset than the dataset provided by Kaggle. But one thing that remains certain is that the individual model maintains their position, for example, XGBoost has the best performance in both the datasets even though it was 98% and 92% in UCI and Kaggle datasets respectively. Likewise, other models.

Table 4: A Comparative Analysis of Multiple Classifiers on Kaggle Dataset

Classifier	**Accuracy**	**Precision**	**Recall**	**F1 Score**
XGBoost	92.3	94	97	94
RF	87	89	97	93
SVM	84	85	91	88

KNN	70	91	71	80
NB	72	92	72	81

In Figure 14, figure 15, and Figure 16, the line kind of plot was used to plot the actual versus the predicted values. The blue lines indicate the actual values of the dataset while the orange color represents the prediction results. Figure 14, naïve Bayes prediction results were compared with the actual data, hence the plot below. The line is not so much good because we can see the difference. For the random forest in Figure 15, the difference between the actual and the predicted is not quite clear as the one in naïve Bayes but it has some way to go to get the actual prediction.

The extreme gradient boost in Figure 16 as stated earlier outperformed all models in performance accuracy and other evaluation metrics. It has a nearly perfect prediction, the difference between the actual and prediction is very minimal hence needing some simple adjustment to achieve a perfect output. The two lines in this plot, figure 16 looks like one of the accuracies achieved by the model. Figures 17 and 18 display the actual and predicted performance of support vector machine and k-nearest neighbor.

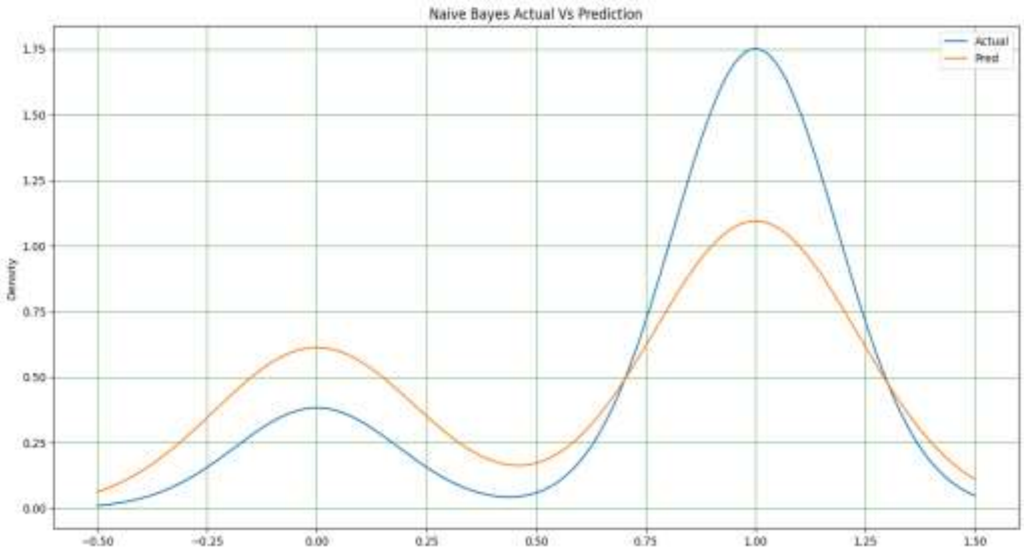

Figure 14: Naïve Bayes Actual Verses Predicted values

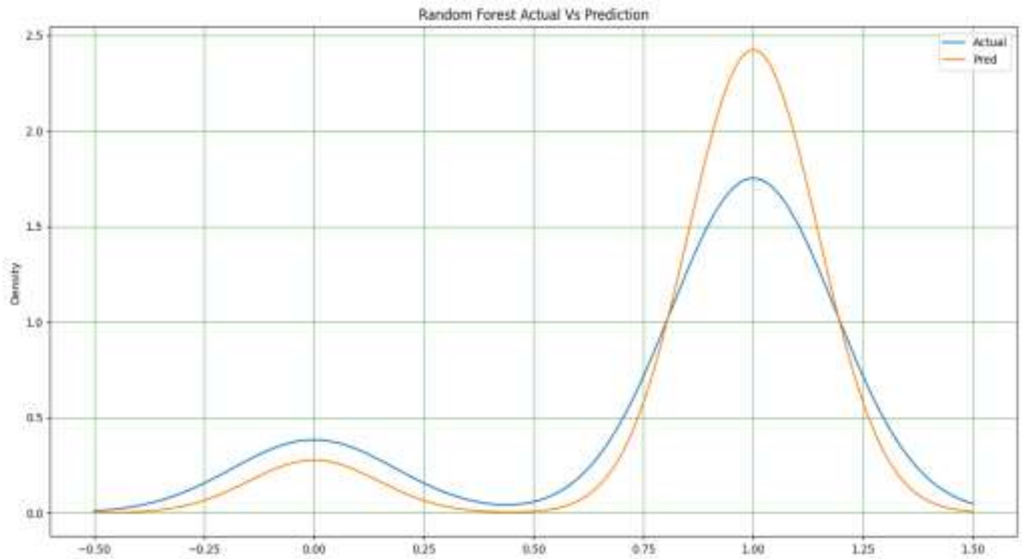

Figure 15: Random Forest Actual Verses Predicted values

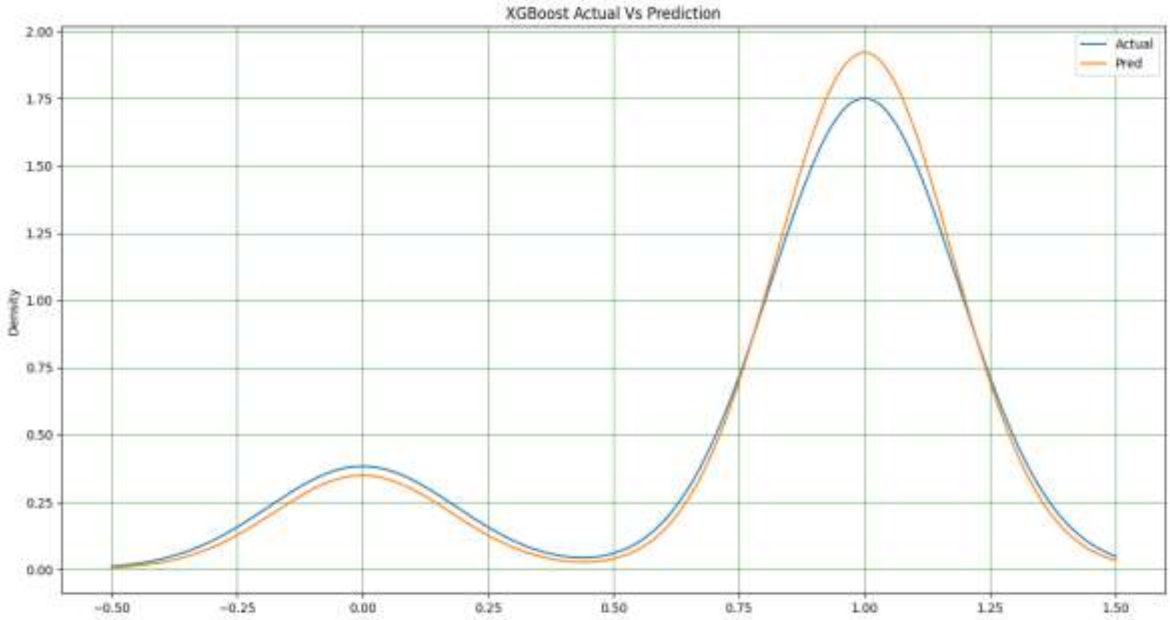

Figure 16: Extreme Gradient Boosting Actual Verses Predicted values

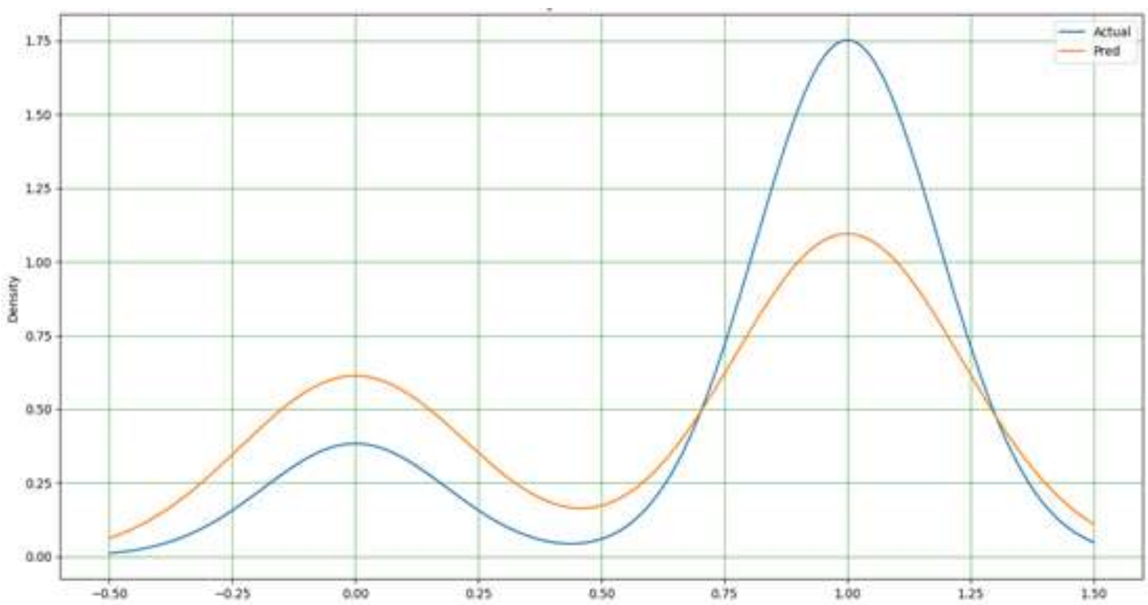

Figure 17: SVM Actual Verses Predicted values

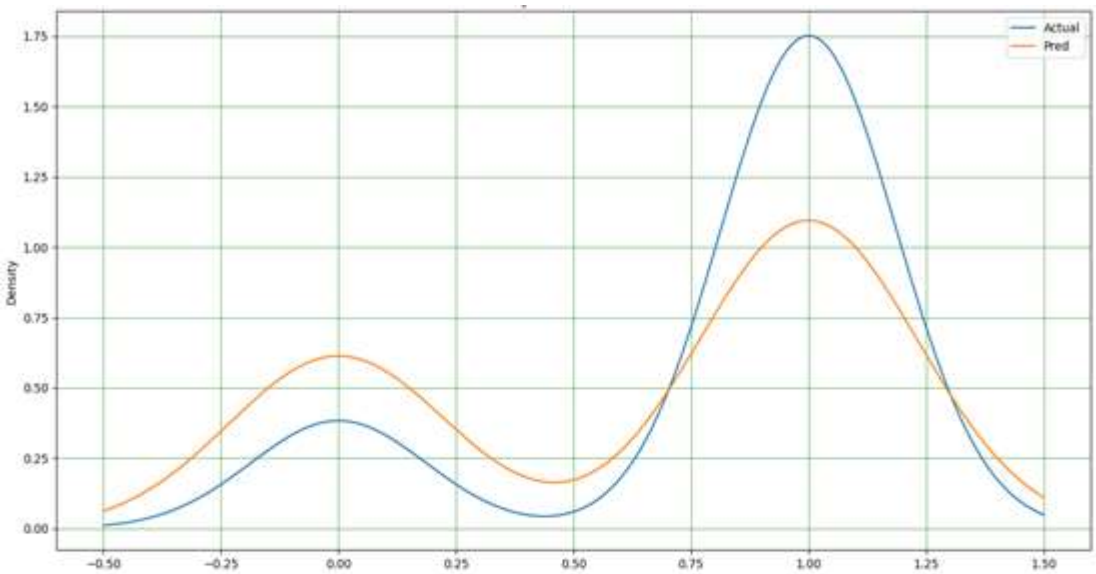

Figure 18: KNN Actual Verses Predicted values

In all the figures above, the x-axis represents the predicted value and the y-axis represents the period. Figures 19, 20, 21, and 22 below present the outcome of the evaluation metrics used to further evaluate the performance of the models. As can be seen, our proposed model has outperformed all other models used in the previous study. In Figure 17, we have the highest accuracy score, precision also highest and hundred percent recall, and an excellent f1 score in Figure 20. The percentage of accurate predictions in each class in this study was computed using the average recall. Also, the accuracy of the forecasts has been calculated, which establishes confidence in a specific forecast. Moreover, average precision was computed for evaluation reasons, allowing us to determine the average of all precision ratings for each class. Finally, the f1-score was calculated, which offers a weighted average of each class's precision and recall. Penalizes the score for each sample in a class that was incorrectly predicted.

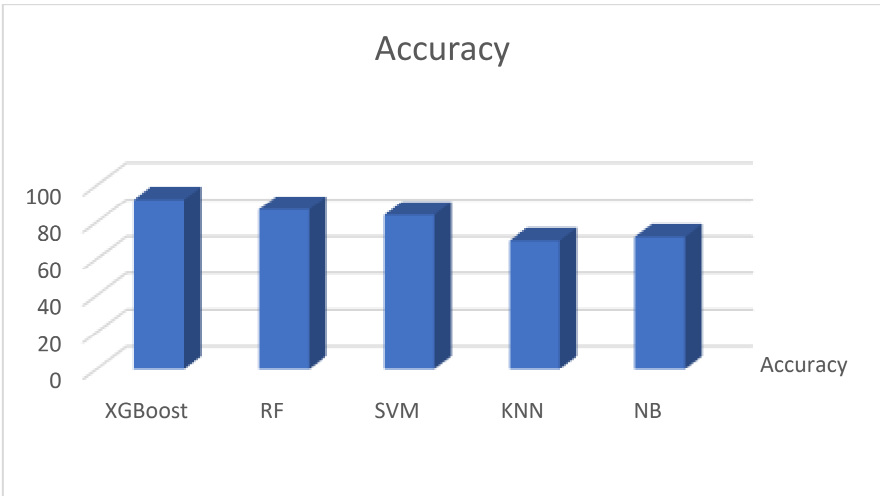

Figure 19: Accuracy

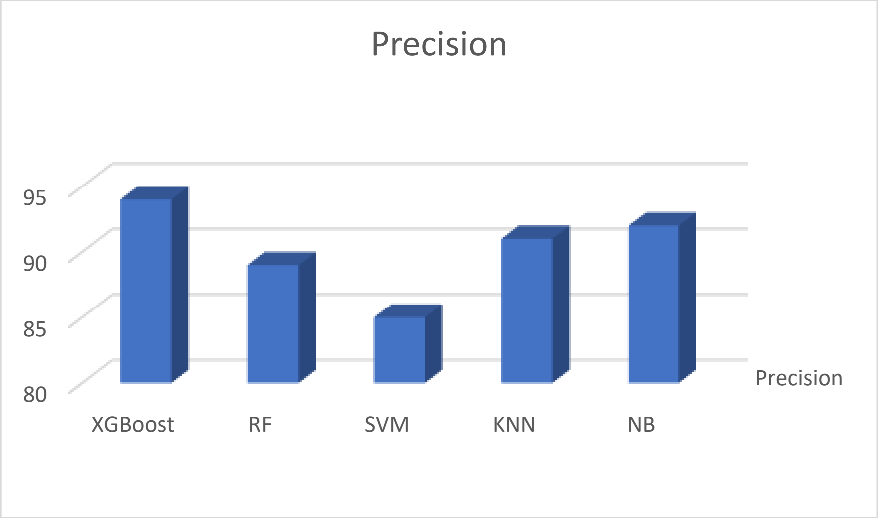

Figure 20: Precision

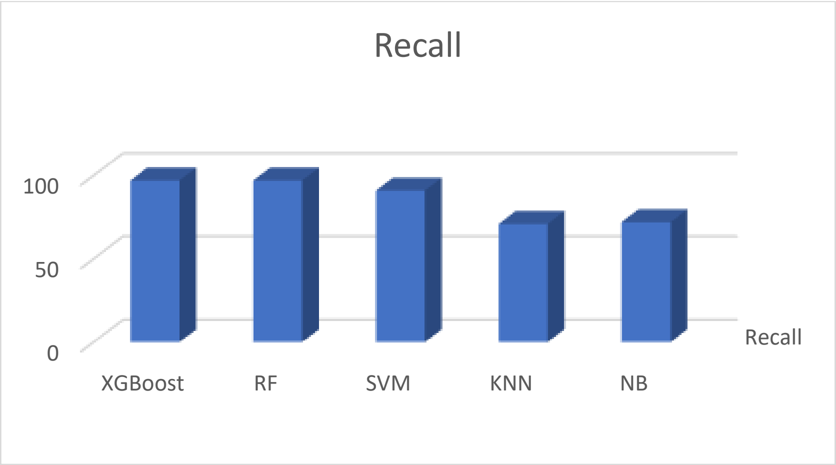

Figure 21: Recall

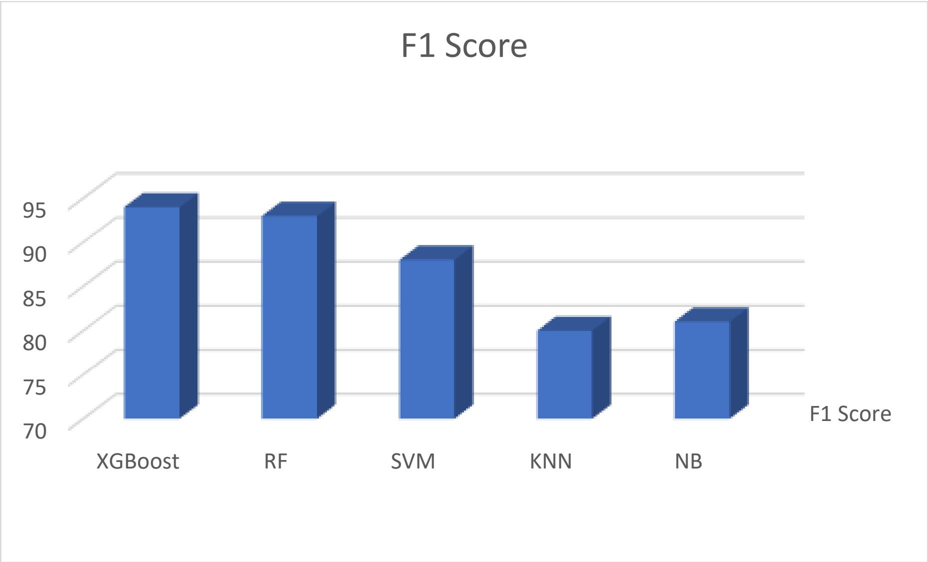

Figure 22: F1 Score

Figures 19 to 22 represent the individual accuracies of the boosting algorithms used for the study on the Kaggle dataset. The testing dataset accuracy produced almost 92% accuracy with the eXtreme gradient boosting model. Random Forest (RF) and Support vector machine (SVM) algorithms generated

87% and 87% accuracy respectively. Figure 22 shows the detailed results of combined-machine learning algorithms models with precision, recall, and f1-score.

7.0 CONCLUSION

This research project performed analysis and evaluation of various supervised machine learning models to project the presence of Parkinson's disease among individuals. An extreme gradient boosting model along with other models such as the naïve Bayes, random forest, k-nearest neighbor, and support vector machine was implemented to determine which would give the best outcome. eXtreme Gradient Boosting model is regarded as a powerful boosting machine learning technique, outstandingly in terms of speed and accuracy. Results obtained showed that the XGBoost model which is an optimized version of the other boosting models is capable of projecting the presence of Parkinson's disease with relatively almost a very high accuracy compared to other algorithms. The implemented classification algorithm extreme gradient boosting method achieved approximately 98% accuracy, 97.5% precision, 100.0% recall, and 98.7% f1-score which is of high performance compared to (Milano et al, 2021) classification accuracy of about 93% using a K-NN classification algorithm and (Aich et al., 2020) that applied naïve bayes, k-nearest neighbor, support vector machine and random forest classifiers. It is recommended that the boosting algorithms should be further explored to conduct research on various range of areas or experiment as it has the potential to outperform other popular and commonly used machine learning classifiers.

7.1 CONTRIBUTION

First of all, this study work would provide insight into what is Parkinson's disease and what its symptoms are, and how to prevent its further spread in society especially among old age people, men, and women. This study has introduced and unveiled the capabilities of boosting machine learning algorithms, extreme gradient boosting (XGBoost). The study presents state-of-the-art research finding to the academic databases which will be used for reference and both supplement and advance studies that would continue to bring out reasons for defeating the disease to a minimum if not eliminated. Though similar studies were being performed using various methods including the machine learning models, the extreme gradient boosting algorithm should need to be further studied to reveal hidden or unexplored areas to be able to obtain a perfect result. Even though the output we got is nice but there is a need to get a better and the easiest way to do it and get the very best result.

REFERENCES

Aich, S., Youn, J., Chakraborty, S., Pradhan, P. M., Park, J., Park, S., and Park, J. (2020). A Supervised Machine Learning Approach to Detect the On/O State in Parkinson's Disease Using Wearable Based Gait Signals. Diagnostics 2020, 10, 421; doi:10.3390/diagnostics10060421 www.mdpi.com/journal/diagnostics

Aljalal, M., Aldosari, S.A., AlSharabi, K., Abdurraqeeb, A.M., Alturki, F.A. (2022) Parkinson's Disease Detection from Resting-State EEG Signals Using Common Spatial Pattern, Entropy, and Machine Learning Techniques. Diagnostics 2022, 12, 1033. https://doi.org/10.3390/diagnostics12051033

Alzubaidi, M.S., Shah, U., Dhia Zubaydi, H., Dolaat, K., Abd-Alrazaq, A.A., Ahmed, A., Househ, M. (2021). The Role of Neural Network for the Detection of Parkinson's Disease: A Scoping Review. Healthcare, 9, 740. https://doi.org/10.3390/healthcare9060740

Bae, D.-J., Kwon, B.-S., Song, K.-B. (2022). XGBoost-BasedDay-Ahead Load Forecasting Algorithm Considering Behind-the-Meter Solar PV Generation. Energies, 15, 128. https://doi.org/10.3390/en15010128

Borzì, L., Mazzetta, I., Zampogna, A., Suppa, A.; Olmo, G., Irrera, F. (2021) Prediction of Freezing of Gait in Parkinson's Disease Using Wearables and Machine Learning. Sensors 2021, 21, 614. https://doi.org/10.3390/s21020614

Chintalapudi, N., Battineni, G., Hossain, M. A., and Amenta, F. (2022). Cascaded Deep Learning Frameworks in Contribution to the Detection of Parkinson's Disease. Bioengineering 2022, 9, 116. https://doi.org/10.3390/bioengineering9030116

Fujita, T., Luo, Z., Quan, C., Mori, K., Cao, S. (2021). Performance Evaluation of RNN with Hyperbolic Secant in Gate Structure through Application of Parkinson's Disease Detection. Appl. Sci., 11, 4361. https://doi.org/10.3390/app11104361

Gil-Martín, M., Montero, J. M., and San-Segundo, R., (2019) Parkinson's Disease Detection from Drawing Movements Using Convolutional Neural Networks, Electronics 2019, 8, 907; doi:10.3390/electronics8080907 www.mdpi.com/journal/electronics

How, M., Uddin, M.N., Park, S.-B. (2021). Vocal Feature Extraction-Based Artificial Intelligent Model for Parkinson's Disease Detection. Diagnostics 2021, 11, 1076. https://doi.org/10.3390/diagnostics11061076

Hoq, M.; Uddin, M.N.; Park, S.-B. (2021). Vocal Feature Extraction-Based Artificial Intelligent Model for Parkinson's Disease Detection. Diagnostics, 11, 1076. https://doi.org/10.3390/diagnostics11061076

Kurmi, A., Biswas, S., Sen, S., Sinitca, A., Kaplun, D., and Sarkar, R. (2022). An Ensemble of CNN Models for Parkinson's Disease Detection Using DaTscan Images. An Ensemble of CNN Models for Parkinson's Disease Detection Using DaTscan Images. Diagnostics 2022, 12, 1173. https://doi.org/10.3390/diagnostics12051173

Ozkan, H., (2016). A Comparison of Classification Methods for Telediagnosis of Parkinson's Disease. Entropy 2016, 18, 115; doi:10.3390/e18040115

Mei J, Desrosiers C and Frasnelli J. (2021) Machine Learning for the Diagnosis of Parkinson's Disease: A Review of Literature. Front. Aging Neurosci. 13:633752. DOI: 10.3389/fnagi.2021.633752

Kurmi, A., Biswas, S., Sen, S., Sinitca, A., Kaplun, D., Sarkar, R. An Ensemble of CNN Models for Parkinson's Disease Detection Using DaTscan Images. Diagnostics 2022, 12, 1173. https://doi.org/10.3390/diagnostics12051173

Demir, F., Siddique, K., Alswaitti, M., Demir, K.; Sengur, A. A. (2022). Simple and Effective Approach Based on a Multi-Level Feature Selection for Automated Parkinson's Disease Detection. J. Pers. Med. 2022, 12, 55. https://doi.org/10.3390/jpm12010055

Pardoel, S., Kofman, J., Nantel, J., and Lemaire, E. D. (2019). Wearable-Sensor-Based Detection and Prediction of Freezing of Gait in Parkinson's Disease: A Review. Sensors 2019, 19, 5141; doi:10.3390/s19235141

Pianpanit, T., Lolak, S., Sawangjai, P., Sudhawiyangkul, T. and Wilaiprasitporn, T. (2021). Parkinson's Disease Recognition Using SPECT Image and Interpretable AI: A Tutorial. IEEE Sensors Journal.

Pramanik, M., Pradhan, R., Nandy, P., & Bhoi, A. K. (2021). Applied Sciences Machine Learning Methods With Decision Forests For Parkinson ' S Detection. Appl. Sci. 2021, 11, 581. https://doi.org/10.3390/app11020581

Rich, S., Youn, J., Chakraborty, S., Pradhan, P. M., Park, J., Park, S., and Park J. (2020). A Supervised Machine Learning Approach to Detect the On/O State in Parkinson's Disease Using Wearable Based Gait Signals. Diagnostics 2020, 10, 421; doi:10.3390/diagnostics10060421.

Byeon, H. (2020). Application of Machine Learning Technique to Distinguish Parkinson's Disease Dementia and Alzheimer's Dementia: Predictive Power of Parkinson's Disease-Related Non-Motor Symptoms and Neuropsychological Profile. J. Pers. Med. 2020, 10, 31; doi:10.3390/jpm10020031

Williamson, J.R., Telfer, B., Mullany, R., Friedl, K.E. (2021). Detecting Parkinson's Disease from Wrist-Worn Accelerometry in the U.K. Biobank. Sensors, 21, 2047. https://doi.org/10.3390/s21062047

https://jupyter.org/ accessed on 4th October, 2022

https://scikit-learn.org accessed on 4th October, 2022

https://numpy.org accessed on 4th October, 2022

https://jupyter.org accessed on 4th October, 2022

https://pandas.pydata.org accessed on 4th October, 2022