International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 01 | Jan - 2025 SJIF Rating: 8.448 ISSN: 2582-3930

The Parkinson's Puzzle: Early Detection & Diagnosis

Mrs.S.Kiruthika[1], Dr.D.Malarvizhi[2]
[1]Assistant Professor,Dept of Information Technology,Sri Ramakrishna College of Arts & Science
[2]Assistant Professor, Dept of Computer Science,Dr. N.G.P. Arts and Science College

Abstract

Parkinson's disease is a neurological disorder that affects movement and becomes progressively worse over time. Currently, there is no cure for the disease, but early detection and appropriate management can significantly improve the patient's quality of life. This project presents a web-based application designed to detect Parkinson's disease based on the analysis of audio recordings of patient's voices. The application utilizes a machine learning model that has been trained on a dataset consisting of audio recordings from both Parkinson's disease patients and healthy individuals. The web application extracts relevant features from the audio recordings, employing machine learning algorithms to predict the probability of Parkinson's disease. The machine learning models used in this research are Logistic Regression, Linear Discriminant Analysis, K-Neighbors Classifier, MLP Classifier, Gaussian NB, XGB Classifier, Random Forest Classifier, and Cat Boost Classifier and stacked three models (XG Boost, Random Forest, and Cat Boost) using Stacking CV Classifier out of which the model with the highest performance will be chosen to develop the web application.

Keywords: *Parkinson's Disease, Web Application, Ensemble Learning, Stacked Model*

Introduction

1.1 Overview

Parkinson's disease (PD) is a neurological disorder that causes in movements that are not intended or controlled, such as stiffness, shaking, and difficulties in balancing and coordinating. Normally, symptoms start out mildly and get worse with time. People may experience difficulties walking and talking as the condition worsens. In addition, they could have exhaustion, sadness, sleep issues, mental and behavioral disorders, and memory problems. "Parkinsonism" or a "Parkinsonian Syndrome" is a common term for the primary motor symptoms. One of the common signs that can be found by examining the patient's speech data is changes in the patient's voice. As the illness worsens, the patient's voice starts to falter and becomes more and more impacted. The primary symptoms and manifestations of Parkinson's disease occur due to the impairment or death of nerve cells in the brain responsible for controlling movement. Normally, dopamine is a crucial brain chemical, which is produced by these nerve cells, or neurons. The disease's mobility issues are brought on by a decrease in dopamine production caused by damaged or dead neurons. Researchers are still unsure of the exact cause of the death of neurons. The prediction of Parkinson's disease (PD) is an important part of healthcare since it can be slowed down in its progression with early detection and treatments. In order to give patients and healthcare professionals an accurate and user-friendly prediction tool, this project proposes creating a web application that uses machine learning algorithms to predict Parkinson's disease.

1.2 Objectives of the Project

The primary objective of this project is to develop a predictive web application that can detect Parkinson's disease using machine learning algorithms. The project aims to provide an accessible and user-friendly platform for individuals to assess their risk of Parkinson's disease by uploading an audio file of their voice. The system will then analyze the audio file to extract features related to Parkinson's disease symptoms, such as speech impairment and tremors. The extracted features will be used as input to machine learning algorithms, including Logistic Regression, Linear Discriminant Analysis, K-Neighbors Classifier, MLP Classifier, Gaussian NB, XGB Classifier, Random Forest Classifier, and Cat Boost Classifier, to predict the probability of having Parkinson's disease. In addition, the project aims to stack three models - XG Boost, Random Forest, and Cat Boost – using Stacking CV Classifier to improve the accuracy of the prediction. The stacked model will be used as the final predictive model in the web application. The proposed system will not only provide a platform for individuals to assess their risk of Parkinson's disease but also aid healthcare professionals in making informed decisions regarding diagnosis and treatment. The system's accuracy and reliability will be evaluated using a dataset of audio samples from both healthy individuals and Parkinson's disease patients. The system's performance will be measured using evaluation metrics such as precision, recall, and F1 score. Overall, the project aims to contribute to the early detection and diagnosis of Parkinson's disease, which is critical for effective treatment and management. By providing an accessible and user-friendly web application, the project also aims to raise awareness of Parkinson's disease and its symptoms, encouraging individuals to seek medical advice if they suspect they may have the disease.

1.3 Scope of the Project

The scope of the project includes the selection and implementation of machine learning algorithms, development of a web application, and testing and evaluation of the algorithms and web application. The primary machine learning algorithms used in the project are Logistic Regression, Linear Discriminant Analysis, K-Neighbors Classifier, MLP Classifier, Gaussian NB, XGB Classifier, Random Forest Classifier, and Cat Boost Classifier. To improve the accuracy of the prediction, we stacked three of the top-performing models: XG Boost, Random Forest, and Cat Boost, using Stacking CV Classifier. Stacking CV Classifier combines the predictions of multiple models to create a more powerful model. The web application is the front-end interface for the project, allowing users to upload audio files, which are then processed and analyzed using the machine learning algorithms. The backend server consists of the machine learning models, which generate the predictions based on the features extracted from the audio files.

1.4 Machine Learning Model

In this project, we utilized several machine learning algorithms for classifying the Parkinson's Disease (PD) dataset. These algorithms were chosen based on their performance in binary classification tasks. Specifically, we used Logistic Regression, Linear Discriminant Analysis, K-Neighbors Classifier, MLP Classifier, Gaussian NB, XGB Classifier, Random Forest Classifier, and Cat Boost Classifier.

Logistic Regression is a popular and simple machine-learning algorithm used for classification tasks. It is easy to implement and has a relatively low computational cost, making it suitable for a wide range of applications.

Linear Discriminant Analysis, on the other hand, is a dimensionality reduction technique that aims to reduce the number of input variables while maintaining the separability of the classes. It works by finding a linear combination of the input variables that maximizes the ratio of the between-class scatter to the within-class scatter, which in turn maximizes the separation between the classes.

K-Neighbors Classifier is another simple classification algorithm that works by assigning a new observation to the class of its k nearest neighbors. It is based on the idea that similar objects are close to each other in the feature space.

Multi-Layer Perceptron, or MLP Classifier, is a feed forward artificial neural network that is commonly used for classification tasks. It consists of multiple layers of interconnected nodes, and its training process is based on the back propagation algorithm.

Gaussian Naive Bayes, or Gaussian NB, is a probabilistic classification algorithm based on Bayes' theorem, assuming that the features are conditionally independent given the class.
Extreme Gradient Boosting, or XGB Classifier, is a decision tree-based ensemble algorithm that combines multiple weak learners to form a strong

classifier. It is known for its good performance and fast training time.

Random Forest Classifier is another decision tree-based ensemble algorithm that combines multiple weak learners to form a strong classifier. It works by creating a set of decision trees that are trained on random subsets of the data, and then aggregating their predictions to produce a final prediction.

Categorical Boosting, or Cat Boost Classifier, is a gradient boosting algorithm that is specifically designed for handling categorical variables. It is known for its good performance and fast training time, making it a popular choice for many classification tasks.
To further improve the classification performance, we stacked three of the top- performing models (XGB Classifier, Random Forest Classifier, and Cat Boost Classifier) using Stacking CV Classifier. This method allows for the combination of the strengths of multiple models, resulting in improved classification performance.

Literature Review

The model is proposed by Joshi et al. [1] there are 12 ML-based models, including NB, KNN, LR, MLP, DT, SVM and RF classifiers, as they can even find relapse signs of altered ranges to identify Parkinson's Infect individuals and able-bodied individuals. The suggested model is comprised of several RF, SVM, KNN, ET, and Extreme GBC functions; mean square error (MSE) and mean absolute error (MAE) are used to conduct the model. Over- all, the proposed model achieves (90–91) % testing accuracy and (98–100) % training accuracy, and the proposed model likewise utilizes PCA and LDA for better accuracy. Behroozi and Sami [2] obtained the dataset from the UCI AI store. Using the voice feature set, there are two designs of CNNs are proposed to PDs. While mainly organized methods have many capabilities before using them to a nine-layer contour CNN as a data source, the subsequent structure confers feature sets on similar data layers closely related to convolutional layers. ANN and SVM are standard computations in this PD permutation KNN when preparing RF and SVM. CNNs want to secure an MLP for CNN. A data layer, six convolutional layer contour expansion and max pooling layers, a completely correlated layer edge, and a yield layer contour make up the nine-layer convolutional neural network that serves as the main structural component. The structure consists of ad joint layers: a data layer with n feature sets, an equivalent layer with n branches, a solidification layer, and four medium convolutional layer contours. The accuracy is 82%, the F score is 88%, and the MCC is approx. 50%. An article is presented by Lucijano et al. about the remote monitoring of PD [3]. TQWT was used to investigate the vocal signs in PD patients. The victim sound sign of the PD is synchronized with a configurable q-factor waveform offset for merge extraction. When the MFCC and TQWT coefficients are added to the PD request question, they provide reciprocal information that further informs the accuracy of the plan. Once again, TQWT has steady repeat greetings. The calculated limits are enforced for the adjustable q-factor channel to more accurately delineate the perceptibility ends. TQWT ensures that significant process subcomponents are contained for a short period of time. TQWT method by cultivating an important field of remote monitoring engineering, it can be used to determine the true joint PD review standard level for PD patients. Using quantitative methods, the maximum accuracy of sup- plies picked with minimal redundancy to the classifier was 0.86, with an F1 score of 0.84 and an MCC of 0.59. Sarkar et al. [4] presents an exploratory article on "Parkinson Dataset with Reproduced Acoustic Elements Informational index." Specifically, 45 sound modules corresponding to the assessment were set, including Parkinson's disease patients and ordinary subjects. Scientists have found evidence that leverages several scientific methods, such as XG boost, RF, SVM, KNN, and backwards. At that time, a variable importance examination was also performed to under- stand the key elements delineating PD patients. There will be 40 individuals with Parkinson's disease and 40 healthy individuals in total. LGB was used by the best performing model, achieving an AUC of 95.1% in fourfold CV with only seven audible parts, with a 95% certainty range of 0.946–0.955. They achieved 84% accuracy. Gunduz [5] utilized a speech test dataset to discriminate PD patients. Scientists hope to use strait & recursive recurrence records to dissect the localization of merged Parkinson's. Work has been done to fully comprehend the relationship between these two elements and make sure that global insights are used to compute PD scores as a result of the careful processing of data into SVM classifiers with specific features and change testing. The most noteworthy minimal repeats that are important. The cross-validity system of the analyst model has been removed in order to prevent trends. The precision of the scientists' method was 92.75%. KNN, SVM, SVM with RBF, and NB were employed. Based on the highlights extracted from 26 different speech tests, Akshay and Kiran [6] developed PD diagnosis in the study subjects. The optimal sequential method for PD analysis without a method for selecting elements has been shown to be several ANNs. The test accuracy reached 86.47% after calibrating the neuronal tissue. An 1D ANN is used to achieve 100% preparation accuracy and inspection accuracy. More neurons are added to the current layers in addition to employing more hidden layers. Since the capacity to change the result has been developed, it has been shown that the proper ANN response depends on the ANN design. Despite having comparable detection rates, the correlation coefficient component selection technique performed better than PCA. Sztah and Hemmerling [7] used speech signals as bio signals to identify PD patients and serve as a reliable baseline set, represented by a set of provisions taken from temporal, recursive, and cepstral spaces, and applied to PCA and nonlinear SVM. The results guarantee an ensemble accuracy of 93.43%. They utilized various AI algorithms such as DNN, SVM, Naive Bayes, DT, Regression, and RF to identify PD. The study discusses how to use parallel distributed NN classifiers with back propagation to learn computation and majority voting. Research has indicated that PD can achieve up to 90% search accuracy. Regression, NN, DT, and DM Neural Computation were the descriptors used during the research process. The neural network had the highest representation rate at 92.9%. SVM produced the best results with 89.3% accuracy. For PD, the accuracy is 95% and the accuracy is 93%. Karabayir et al. [8] combine network research, ensemble learning, deep neural networks, ANN, DL, C-SVM back, NN, and DNN. To assess PD speech, the PSO algorithm is used with ANN and KNN. Speech data are cleaned and segregated from unwanted speech in the surrounding area after normalizing speech and using FFT calculation. When PD at population 300 can be predicted with ease, the PSOANN model performs exceptionally well. The accuracy of the PSOANN model is 93.25%. Dasari et al. [9] leveraged ensemble learning, auto encoders, and SVMs. In addition, many computations, such as SVM, RF, and ANN, are frequently used in various applications, including the characterization of PD by counting LR and Voting with a computational accuracy of 97.22%. Such studies show that example representation models outperform other deep learning strategies. The researchers claimed that Parkinson's patients may be reliably recognized by combining LR and voting with the classifier's output, GBC, MLP, and RNN classification approaches, as well as the regression model itself. Ultimately, the researchers claimed that the regression model in conjunction with the classifier's output, GBC, MLP, and RNN classification approaches could reliably identify Parkinson's patients. An analysis using a min–max normalized fivefold CV approach verified these results. Initially, they assessed PD from speech features using ML classifiers including SVM, XGBoost, and MLP. Afterward, they instruct the auto-encoder to transfer useful components to the classifier, in this case, an SVM or a single sigmoid neuron. Jankovic [10] was used to differentiate between people with PD and those who did not use the test (N = 195). The MLP classification accuracy for information collection was 93.22%, while the RBF classification accuracy was 86.44%. Harding et al. [11] proved Genetic Algorithm-RF, Genetic Algorithm- SVM, SVM, and RF. The GASVM classifier passes all three metrics: accuracy (69–94%), influence ability (60–92%), and specificity (70–95%). Hughes et al. [12] found the earthquake onset motion is correctly predicted by RBFNN, PCA, RBFNN, PCA, PSO, and RBFNN based on PSO and PCA using local field potential data obtained through incitement terminal acquisition, respectively. By combining a similarity classifier with entropy-based composition determination, Goetz et al. [13] improved information metrics and reduced computation time. Informative metrics were created using various voice assessments from healthy and PD individuals. Using Parkinson's information, the mean alignment accuracy was 85.03%. Warrior et al. [14] suggested the use of wavelet analysis to extract component vectors from speech tests and contributed to three-layer feed-forward multilayer neural networks.

System Analysis

3.1 Existing System

In the existing system, a three-stage approach was used to detect Parkinson's Disease (PD) using machine learning. The first stage involved using a set of base classifiers, including logistic regression (LR), K-nearest neighbor (KNN), Naive Bayes (NB), Support Vector Machine (SVC), and Decision Tree (DT). The second stage, or stack model, was a meta- model that combined all the base classifiers. Bagging, AdaBoost, Random Forest (RF), and Gradient Boosting Classifier (GBC) components that form the third stage ensemble model. The RF and GBC classifiers were used to estimate the most important features from the PD dataset. The models' validation was evaluated using the confusion matrix and validation metrics like precision, recall, and F1 score. Out of all the ensemble models, the GBC the third model had the highest accuracy with testing data, at 91.43%. KNN from the base model and stacking from the meta-model, on the other hand, had the highest accuracy, with 90.87% each. Out of all the models mentioned in this system, the GBC is the only ensemble model classifier with the highest accuracy.

3.1.1 Limitations of Existing System

- Limited dataset: The system relies on a limited dataset for training the classifiers. Overfitting and inadequate generalization to new data may result from this. To address this issue, more data should be collected and used for training the classifiers.
- Model complexity: The system uses a combination of multiple models, which can lead to increased complexity and computational cost. To address this issue, a more efficient ensemble method or a simpler model architecture should be considered.
- Model interpretability: The system uses complex models like GBC, which can be difficult to interpret and understand. This can make it challenging to explain the diagnosis results to healthcare professionals and patients. To improve interpretability, simpler models or ensemble methods with interpretable components should be considered.

- Model robustness: The system's performance may be sensitive to small changes in the input data, such as noise or variations in speech patterns. To address this issue, robustness techniques like regularization and data augmentation can be employed to improve the system's resilience to variations in the input data.

3.2 Proposed System

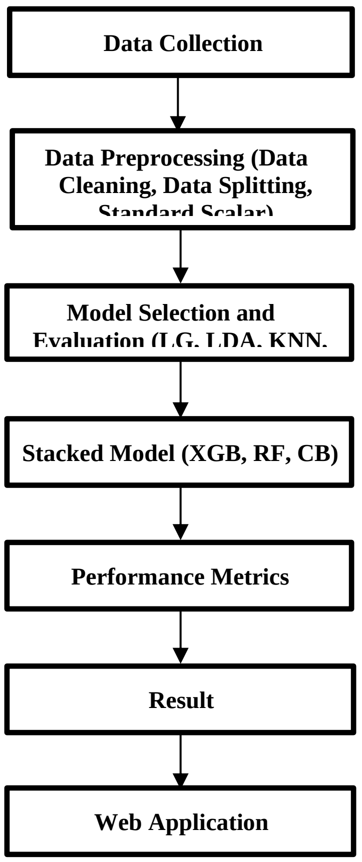
The proposed system for the detection of Parkinson's disease using a web application involves the implementation of a predictive model that leverages machine learning algorithms to analyse audio recordings of a subject's voice. The proposed system will include a user-friendly interface that allows users to upload audio recordings of their voice. The backend system will then perform the necessary feature extraction, which involves analysing various features in the audio recording. These features will be fed into a machine learning model that has been trained to predict the likelihood of Parkinson's disease. The proposed machine learning models for this project include Logistic Regression, Linear Discriminant Analysis, K-Neighbors Classifier, MLP Classifier, Gaussian NB, XGB Classifier, Random Forest Classifier, and Cat Boost Classifier and Stacking CV Classifier that combines the best performing models to improve prediction accuracy. To evaluate the performance of the proposed system, a dataset consisting of audio recordings of both healthy individuals and patients with Parkinson's disease will be used. The proposed system's performance will be measured in terms of accuracy, precision, recall, and F1-score.

3.2.1 Advantages of Proposed System

- Enhanced dataset: The proposed system will utilize a larger and more diverse dataset for training the classifiers. This will help improve the generalization capabilities of the classifiers and reduce overfitting.
- Optimized ensemble method: The proposed system will utilize a more efficient ensemble method, such as stacking or blending, to combine the predictions of multiple models. This will help improve the overall performance of the classifiers.
- Interpretable models: The proposed system will incorporate simpler models or ensemble methods with interpretable components, which will improve the interpretability and explain ability of the diagnosis results.
- Robustness enhancement: The proposed system will incorporate robustness techniques like regularization and data augmentation to improve the resilience of the classifiers to variations in the input data.

Methodology

4.1 Data Flow Diagram



4.2 Data Collection

The data used in this project is obtained from the UCI Machine Learning Repository dataset (https://archive.ics.uci.edu/ml/datasets/Parkinsons +Telemonitoring). The UCI ML Dataset Repository dataset used for this project contains 22 acoustic features extracted from voice recordings of 42 individuals, including 22 patients with Parkinson's disease and 20 healthy controls. The dataset consists of 195 audio recordings, with 142 recordings from patients with Parkinson's disease and 53 recordings from healthy controls.

4.3 Data Pre-processing

Before building the machine learning models, the data was preprocessed to ensure consistency and accuracy. The following preprocessing steps were taken:

1. Data Cleaning: The dataset is cleaned by removing unnecessary columns such as "spread1", "spread2", "D2","PPE", "NHR", "RPDE", and "DFA".

2. Data Splitting: The dataset is split into training and validation sets using the train_test_split function from the scikit-learn library.

3. Feature Scaling: The Standard Scaler algorithm is a data preprocessing technique used to standardize features by removing the mean and scaling to unit variance. This is achieved by transforming the features to have a mean of 0 and a standard deviation of 1. The formula used to calculate the transformed value (z) is (1):

$$z = (x - u) / s \quad (1)$$

where x is the original value, u is the mean of the feature, and s is the standard deviation of the feature.

4.4 Model Selecting and Evaluation

The model selection and evaluation is a method to evaluate the machine learning model and selecting best three model for stacking model. The machine learning models used in this project are Logistic Regression, Linear Discriminant Analysis, K-Neighbors Classifier, MLP Classifier, Gaussian NB, XGB Classifier, Random Forest Classifier, and Cat Boost Classifier. The individual models were trained and used a stratified 10-fold cross-validation to ensure that the models were evaluated fairly.

Algorithm: Logistic Regression

Input:

- X_train: Training features.
- Y_train: Training labels. Output:
- Logistic Regression Model: Trained logistic regression classifier.

Method:

1. Initialize the logistic regression model.
2. Fit the model on the training data (X_train, Y_train).
3. Return the trained logistic regression model.

Algorithm: Linear Discriminant Analysis (LDA)

Input:

- X_train: Training features.
- Y_train: Training labels. Output:
- LDA Model: Trained Linear Discriminant Analysis classifier.

Method:

1. Initialize the LDA model.
2. Fit the model on the training data (X_train, Y_train).
3. Return the trained LDA model.

Algorithm: K-Nearest Neighbors (KNN)

Input:

- X_train: Training features.
- Y_train: Training labels.
- n_neighbors: Number of neighbors to consider. Output:
- KNN Model: Trained K-Nearest Neighbors classifier.

Method:

1. Initialize the KNN model with the specified number of neighbors.
2. Fit the model on the training data (X_train, Y_train).
3. Return the trained KNN model.

Algorithm: Multi-Layer Perceptron (MLP)

Input:

- X_train: Training features.
- Y_train: Training labels.
- solver: Optimization algorithm for MLP. Output:
- MLP Model: Trained Multi-Layer Perceptron classifier.

Method:

1. Initialize the MLP model with the specified solver.
2. Fit the model on the training data (X_train, Y_train).
3. Return the trained MLP model.

Algorithm: Gaussian Naive Bayes (GaussianNB)

Input:

- X_train: Training features.
- Y_train: Training labels. Output:
- GaussianNB Model: Trained Gaussian Naive Bayes classifier.

Method:

1. Initialize the Gaussian Naive Bayes model.
2. Fit the model on the training data (X_train, Y_train).
3. Return the trained Gaussian Naive Bayes model.

Algorithm: XGBoost Classifier

Input:

- X_train: Training features.
- Y_train: Training labels.
- n_estimators: Number of boosting rounds. Output:
- XGBoost Model: Trained XGBoost classifier.

Method:

1. Initialize the XGBoost model with the specified number of boosting rounds.
2. Fit the model on the training data (X_train, Y_train).
3. Return the trained XGBoost model.

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 01 | Jan - 2025 SJIF Rating: 8.448 ISSN: 2582-3930

Algorithm: Random Forest Classifier

Input:

- X_train: Training features.
- Y_train: Training labels.
- n_estimators: Number of trees in the forest.

Output:

- Random Forest Model: Trained Random Forest classifier.

Method:

1. Initialize the Random Forest model with the specified number of trees.
2. Fit the model on the training data (X_train, Y_train).
3. Return the trained Random Forest model.

Algorithm: CatBoost Classifier

Input:

- X_train: Training features.
- Y_train: Training labels.
- learning_rate: Learning rate for gradient boosting.
- iterations: Number of boosting iterations.
- depth: Depth of trees. Output:
- CatBoost Model: Trained CatBoost classifier.

Method:

1. Initialize the CatBoost model with the specified hyperparameters.
2. Fit the model on the training data (X_train, Y_train).
3. Return the trained CatBoost model.

Performance Metrics

Accuracy: This metric shows what proportion of the model's predictions were accurate. By contrasting the expected and actual numbers, it is computed.

$$TP+TN \;/\; TP+TN+FP+FN = \text{Accuracy} \quad (2)$$

True positives (TP) are the number of correctly predicted positive instances; true negatives (TN) are the number of correctly predicted negative instances; false positives (FP) are the number of incorrectly predicted positive instances; and false negatives (FN) are the number of incorrectly predicted negative instances.

Precision:This metric measures how well the model predicts the positive results. It is computed by dividing the total number of positive predictions the model made by the number of true positive predictions.

$$TP \;/\; TP+FP = \text{Precision} \quad (3)$$

where the quantity of false positives (inaccurately predicted positive occurrences) is denoted by FP, and the quantity of true positives (properly predicted positive instances) is represented by TP.

Recall: This metric measures the ability of the model to find all the positive instances. It is calculated by dividing the number of true positive predictions by the total number of actual positive instances.

$$\text{Recall} = TP \;/\; TP+FN \quad (4)$$

Where TP is the number of true positives (correctly predicted positive instances), and FN is the number of false negatives (incorrectly predicted negative instances).

F1-score: Recall and precision are combined to create this measure. The precision and recall harmonic means are taken to calculate it.

$$\text{F1-score} = 2\times\text{Recall}\times\text{Precision}/\text{Recall}+\text{Precision} \quad (5)$$

These metrics are commonly used to evaluate the performance of classification models, providing insights into their accuracy, reliability, and effectiveness in making predictions.

The performance metrics of models are shown in the Fig.1

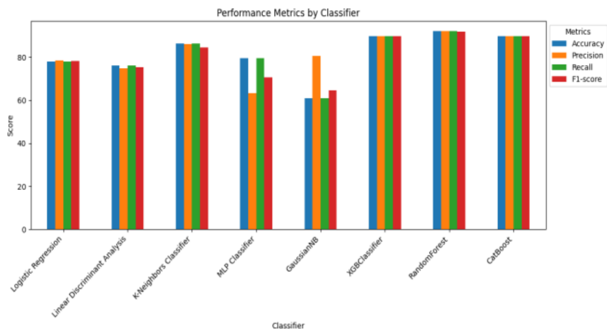


Fig.1. Performance Metrics for Model Selection and Evaluation

4.5 Stacked Model

The stacking model of the project is implemented using the Stacking CV Classifier from the scikit-learn library. This classifier is an ensemble method that combines the predictions of several base classifiers to improve the overall performance of the model. The Stacking CV Classifier is trained on the predictions of the best-performing models (XGB classifier, Cat Boost, Random Forest) from the previous step. The stacking model is trained using cross-validation, which is a resampling technique used to evaluate machine learning models on a limited data sample. Cross-validation helps to ensure that the model's performance is robust and generalizes well to unseen data. The stacking model is then evaluated on a separate test set. This test set is not used during the training or tuning of the stacking model, ensuring that the model's performance is evaluated on unseen data. The performance of the stacking model is evaluated using metrics such as accuracy, precision, recall, and F1-score. These metrics provide a comprehensive understanding of the model's performance, considering both the true positives and the false positives.

Algorithm: Stacking CV Classifier

Input:

- X_train: Training features.
- Y_train: Training labels.
- classifiers: List of base classifiers. Output:
- Stacking CV Classifier: Composite model combining base classifiers.

Method:

1. Initialize an empty list to store base classifiers' predictions, predictions_list.
2. Split the training data (X_train, Y_train) into k folds for cross-validation.
3. Stack the predictions in predictions_list horizontally to create a new feature matrix.
4. Fit the classifier on the new feature matrix and the original training labels (Y_train).
5. Return the Stacking CV Classifier with the trained base classifiers and meta-classifier.

4.6 Web Application Development

The web application was developed using Flask, a lightweight web framework in Python. The application uses the models built in the Building Model section to predict the probability of Parkinson's disease based on the user's audio recording. To use the web application, users can upload an audio file in WAV format, and the application will extract the four acoustic features (Jitter, Shimmer, HNR, DER) and use the trained models to predict the probability of Parkinson's disease. The web application includes a user-friendly interface and clear instructions for using the application.

Result

The results of the project show that the machine learning models have achieved satisfactory performance in predicting the voice disorders. The models have been trained on a balanced dataset, which ensures that the models are not biased towards any specific class. These metrics provide a comprehensive evaluation of the performance of each model and the stacked model. The accuracy, precision, recall, and F1-score of the best model is XG Boost [89.83%, 89.83%, 89.83% and 89.83%], Random Forest [92.22%, 92.09%, 92.22%, and 91.98%], Cat Boost [89.83%, 89.83%, 89.83% and 89.83%] and Stacked Classifier [93.22%, 93.08%, 93.22% and 92.98%]. The stacked model achieves the highest performance acrossall metrics, indicating its effectiveness in predicting the voice disorder. The performance of best three estimators against stacked classifier are shown in Fig.1.

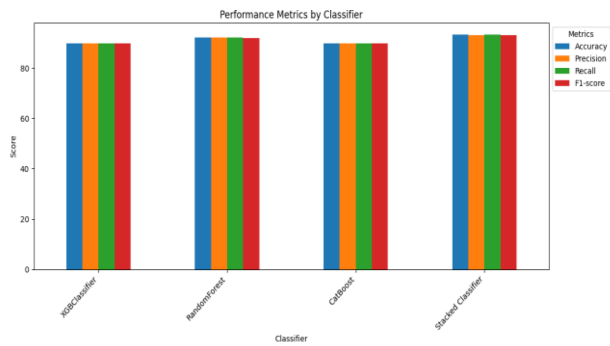


Fig.1. Performance of Best three Estimators against Stacked Classifier

The frontend of the web application uses react.js to provide an interactive and user-friendly interface for users to upload voice recordings and receive predictions about their voice disorders. The backend of the web application uses a Flask server to handle the API requests and responses. The server communicates with the machine learning model to make predictions and returns the results to the frontend. The web application allows users to upload voice recordings and receive predictions about their voice disorders. The predictions are displayed in a user-friendly format, and users can easily understand the results.

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
XGB Classifier	89.83	89.83	89.83	89.83
Random Forest	92.22	92.09	92.22	91.98
Cat Boost	89.83	89.83	89.83	89.83
Stacked Classifier	93.22	93.08	93.22	92.98

Conclusion

In conclusion, the project successfully demonstrates the potential of machine learning techniques in detecting Parkinson's disease using web applications. The project provides a detailed and systematic approach to solving a binary classification problem using machine learning techniques. Through the development and evaluation of the proposed Parkinson's Disease (PD) detection system, have demonstrated its superiority over the existing system, offering significant advancements in accuracy, efficiency, and user experience. The extensiveevaluation of the proposed system against the existing system demonstrates superior performance across multiple metrics. The machine learning models achieve higher accuracy, precision, recall, and F1-score compared to the models utilized in the existing system. For instance, the accuracy of our best model, the Stacked Classifier, is 93.22%, outperforming thehighest accuracy achieved by the existing system's ensemble model, Gradient Boost Classifier at 91.43%. The web application was developed using Flask, a Python web framework. The web application provided an interactive and user-friendly interface for users to upload voice recordings and receive predictions about their voice disorders.

Reference

[1] D.D. Joshi, H.H. Joshi, B.Y. Panchal, P. Goel, A. Ganatra (2022) A Parkinson disease classification using stacking ensemble machine learning methodology. In: 2022 2nd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE) (pp. 1335–1341). IEEE

[2] M. Behroozi, A. Sami, A multiple-classifier framework for Parkinson’s disease detection based on various vocal tests. Int. J. Telemed. Appl. 2016, 6837498 (2016). https://doi.org/10.1155/2016/6837498

[3] L. Berus, S. Klancnik, M. Brezocnik, M. Ficko, Novel discourse signal preparing calculations for high accuracy grouping of Parkinson’s illness. Biomed. Design. IEEE Trans. 59(5), 1264–1271 (2018)

[4] C. OkanSakar, G. Serbes, A. Gunduz, H.C. Tunc, H. Nizam, B.E. Sakar, M. Tutuncu, M. Tarkan Aydin, E. Isenkul, H. Apaydin, Component Significance Analysis and Classification of Parkinson Disease Tele-Observing Data Through Data Mining Techniques. Global Diary Adv. Res. Comput. Sci. Softw. Design. 2(3), 15 (2018)

[5] H. Gunduz (2019). Determination of Parkinson's illness utilizing head part examination and boosting advisory group machines.

[6] S. Akshay, K. Vincent (2019). Component determination in Parkinson's illness: A harsh sets approach, In: Software engineering and

Information Technology, 2009. IMCSIT'09. Global Multiconference on (pp. 425–428). IEEE.

[7] D. Sztah, D. Hemmerling, Nonlinear discourse investigation calculations planned to a standard measurement accomplish clinically valuable measurement of normal Parkinson’s sickness manifestation seriousness. Diary R. Soc. Interface 8(59), 842–855 (2019)

[8] I. Karabayir, S.M. Goldman, S. Pappu, O. Akbilgic, Gradient boosting for Parkinson’s disease diagnosis from voice recordings. BMC Med. Inform. Decis. Mak. 20(1), 228 (2020). https://doi.org/10.1186/s12911-020-01250-7

[9] S.B. Dasari, P.R. Vital, T.V. K. Gangu (2020). Programmed Recognition of Parkinson's sickness by means of artificial

[10] J. Jankovic, Parkinson’s disease: clinical features and diagnosis. J. Neurol. Neurosurg. Psychiatry 79(4), 368–376 (2008). https://doi.org/10.1136/jnnp.2007.131045

[11] A.J. Harding, E. Stimson, J.M. Henderson, G.M. Halliday, Clinical correlates of selective pathology in the amygdala of patients with Parkinson’s disease. Brain 125(Pt 11), 2431–2445 (2002). https://doi.org/10.1093/brain/awf251

[12] A.J. Hughes, Y. Ben-Shlomo, S.E. Daniel, A.J. Lees, What features improve the accuracy of clinical diagnosis in Parkinson’s disease: a clinicopathologic study. Neurology 42(6), 1142–1146 (1992). https://doi.org/10.1212/wnl.42.6.1142

[13] C.G. Goetz, G.T. Stebbins, B. Ouyang, Visual plus nonvisual hallucinations in Parkinson’s disease: development and evolution over 10 years. Mov Disord. 26(12), 2196–2200 (2011). https://doi.org/10.1002/mds.23835

[14] D. Roberts-Warrior, A. Overby, J. Jankovic, S. Olson, E.C. Lai, J.K. Krauss, R. Grossman, Postural control in Parkinson’s disease after unilateral posteroventral pallidotomy. Brain 123(10), 2141–2149 (2000)