UNIABUJA Journal of Engineering and Technology

https://ujet.uniabuja.edu.ng/

ISSN: 2714-3236 (Online); 2714-3228 (Print)

Volume 2, Issue 2, 2025; 127-134

Enhancing Prediction of Parkinson's Disease Using Stacked Ensemble Learning Algorithm

Bolaji A. OMODUNBI[1], Afeez A. SOLADOYE[2], Florence O. AWE[3], Mutiu B. FALADE[4], Tolulope O. OMODUNBI[5], Olubunmi J. JOODA[6], Adedayo A. SOBOWALE[7]

[1,2,7]*Department of Computer Engineering, Federal University, Oye-Ekiti, Nigeria*
[3]*Department of Computer Engineering, Federal University of Technology, Akure, Nigeria*
[4]*Department of Computer Engineering, Federal University, Wukari, Nigeria*
[5]*Department of Family Medicine, Oluyoro Catholic Hospital, Ibadan, Oyo State, Nigeria*
[6]*Department of Computer Engineering, Redeemers University, Ede, Osun State, Nigeria*

[1]*Bolaji.omodunbi@fuoye.edu.ng*, [2]*Afeez.soladoye@fuoye.edu.ng*, [3]*omosighoflorence@gmail.com*, [4]*falade@fuwukari.edu.ng*, [5]*tolulopeomodunbi22@gmail.com*, [6]*joodaj@run.edu.ng*, [7]*adedayo.sobowale@fuoye.edu.ng*

Abstract

Accurate and early diagnosis of Parkinson's disease (PD) is still a challenge. In this work, stacked ensemble learning is explored for enhanced PD prediction from voice data. The "Parkinson's" dataset consisting of 195 instances from 22 recordings of voices (features) was downloaded from Kaggle. Preprocessing of the data included resampling through Synthetic Minority Oversampling Techniques to balance against possible class imbalance, as well as normalization through Min-Max scaling. Gain Ratio was used for feature ranking, and experiments were done using the top 5 and top 10 ranked features. Four machine learning algorithms – K-Nearest Neighbor, Logistic Regression, Random Forest, and a Stacked Ensemble (with SVM, KNN, and Random Forest as base learners and Logistic Regression as the meta learner) – were compared using a hold-out evaluation strategy with accuracy, precision, recall, and F1-score as measures of evaluation. It was found that Stacked Ensemble worked the best, particularly when the top 10 features were implemented to train (Accuracy: 95.7%, Precision: 95.0%, F1-Score: 95.0%, Recall: 95.0%) and outperformed all the individual models as well as what was discovered when the top 5 features only were used. By this study, it is concluded that stack ensemble learning coupled with effective feature selection is an effective approach to enhance Parkinson's disease prediction from voice data.

Keywords: *Parkinson disease, gain ratio, stacked ensemble, voice recording.*

1.0 Introduction

Parkinson's disease (PD) is a slowly progressive neurodegenerative disorder affecting over 10 million patients worldwide, leading to tremor, rigidity, and voice disturbance (Poewe et al., 2022). It necessitates early diagnosis, but clinical examination remains highly subjective and detects PD only after the disease reaches an advanced stage. Previous work has explored voice analysis as a low-cost, pain-free diagnostic tool because PD patients exhibit distinctive vocal behaviors such as reduced pitch variability, breathiness, and disorganized articulatory movements (Sakar et al., 2019). Machine learning (ML) has been promising to automatize PD diagnosis from voice features, but the application of singlemodel based approaches has resulted in inconsistent performance and weak generalizability (Ali et al., 2021).

Past studies have predominantly employed isolated algorithms for PD voice classification. For instance, Support Vector Machines (SVM) achieved a success rate of 88% in discriminating PD patients from controls (Benba et al., 2020), whereas Random Forests (RFs) and k-Nearest Neighbors (KNN) achieved 85– 90% accuracy in such tasks (Rusz et al., 2021). However, they have three major limitations: Dominant feature preference, overlooking the subtle vocal cues; Sensitivity to dataset imbalance, as PD voice datasets typically have unbalanced class distributions; and Inability to extract complementary patterns that can improve diagnostic robustness (Tsanas et al., 2022).

Moreover, Support Vector Machines (SVM) (Gozem et al., 2019), K-Nearest Neighbors (KNN) (Ozturk & Ozturk, 2020), and Random Forest (RF) (Pereira et al., 2018) are some of the algorithms that have shown high accuracy in classifying individuals with and without PD from their voice features. These researches have pointed towards the promise of ML to yield an objective and automated method of screening and diagnosis of PD. Relying solely on a single ML algorithm, however, can be restrictive. Every algorithm is susceptible to containing inherent bias and possesses specific strengths. Their effectiveness can be inconsistent depending on the exact character of the dataset in addition to the complexity of the underlying patterns. Individual

Received: 06-04-2025 / Accepted: 11-06-2025 / Published: 20-06-2025

https://doi.org/10.70118/ujet.2025.0202.13

models may fail to adequately represent the entire variety of small voice modifications that exist with PD, and these can lead to suboptimal precision in prediction and generalization (Polikar, 2012).

As a response to these challenges, this study proposes a stacked ensemble learning approach combining SVM, KNN, and RF as base learners and LR as a metalearner. Stacking leverages the benefits of diverse algorithms: SVM's speed in high-dimensional spaces, KNN's locality-sensitive vocal cues, and RF's resistance to noise. The meta-learner then adjusts their collective predictions towards improved overall accuracy and generalizability (Wolpert, 2022). Our approach is evaluated on the widely used University of California Irvine (UCI) PD voice dataset, which includes 756 voice recordings from 131 patients and 64 controls, with 22 acoustic features (e.g., jitter, shimmer, harmonic-to-noise ratio).

2.0 Methodology

The methodology employed in this study to predict Parkinson's disease from a voice dataset is explained in this chapter. The research process entailed data collection, preprocessing, feature selection, application of various machine learning models, and critical performance evaluation.

2.1 Data Acquisition

The data employed here in this research is an open access data set named "parkinsons" readily available on both Kaggle and GitHub. The data set includes 22 varied voice recordings from individuals with and without Parkinson's disease. All the recordings have been mapped as instances in the data set, totaling 195 instances. The data set has a variety of features extracted from these voice recordings, which capture different aspects of vocal characteristics. Some of these features are: MDVP:Fo(Hz)- Average vocal fundamental frequency, MDVP:Fhi(Hz)- Maximum vocal fundamental frequency, Jitter:DDP- Difference of differences between cycles, divided by the average period, Shimmer:APQ3- 3 Point Amplitude Perturbation Quotient, NHR- Noise to Harmonic Ratio and PPE- Pitch Period Entropy among others

2.2 Data Preprocessing

Prior to the training of machine learning models, the data that was collected was put through several critical preprocessing steps to enhance its quality and suitability for analysis:

Resampling of Data with SMOTE: To reverse any class imbalance in the data (i.e., an excess of instances of one class over the other), the Synthetic Minority Over-sampling Technique (SMOTE) was employed.
SMOTE accomplishes this by creating synthetic examples of the minority class by interpolation between true minority class samples. By doing this, the class distribution is equalized, and the machine learning algorithms are prevented from becoming biased in the majority class (Chawla et al., 2002). The resampling was employed before the train-test split, so as to enable even distribution of the resampled instanced during splitting when the label was stratified.
Normalization with Min-Max Scaling: To avoid features with different scales dominating the performance of machine learning models, Min-Max scaling was applied. This normalization technique scales all feature values between 0 and 1 using the following formula in Equation 1:

$$X_{normalized} = \frac{X - X_{min}}{X_{max} - X_{min}} \qquad (1)$$

where (X) is the original feature value, (X_{min}) is the minimum feature value across the dataset, and (X_{max}) is the maximum feature value across the dataset.

2.3 Machine Learning Models for Parkinson's Disease Prediction

This study investigated the performance of five machine learning models, unique in their nature, for Parkinson's disease prediction based on preprocessed voice data:

i. **K-Nearest Neighbor (KNN):** KNN is a non-parametric instance-based learning technique that categorizes new points on the basis of the majority class among their (k) closest neighbors in feature space. The reason why KNN was applied is because of its ease, good performance when dealing with intricate decision boundaries, and proficiency when dealing well with datasets containing an irregular decision boundary (Ozturk & Ozturk, 2020). A (k=3) value was utilized in this study.

ii. **Logistic Regression (LR)**: LR is a linear classifier that makes predictions for a binary outcome (the presence or absence of Parkinson's disease, for instance) on the basis of a sigmoid function. LR is selected as the baseline (meta) model as it is explainable, fast, and has the ability to capture the linear relationship between the features and the target variable.

iii. **Support Vector Machine (SVM)**: SVM is a fast supervised machine learning algorithm for finding the best hyperplane with the maximum class margin that individually delineates the

Received: 06-04-2025 / Accepted: 11-06-2025 / Published: 20-06-2025

https://doi.org/10.70118/ujet.2025.0202.13

points of different classes in an arbitrary high-dimensional feature space. SVM was chosen on the merit of its ability to handle high-dimensional data as well as providing a good prediction from unseen data when the boundary is nonlinear through kernel functions (Gozem et al., 2019). A radial basis function (RBF) kernel was used for the SVM model in this study.

iv. **Random Forest (RF):** Random Forest is an ensemble method that creates many decision trees during training and makes the mode of the classes (for classification) or the mean prediction (for regression) of the trees. RF was selected due to its resistance to overfitting, its high ability to cope with the number of features, and its ability to detect complex non-linear data relationships (Pereira et al., 2018).

v. **Stacked Ensemble Learning**: Stacked ensemble learning is a meta-learning technique where the output of multiple base models is combined by using a meta-learner. A stacked ensemble was constructed in this study by using KNN, SVM, and Random Forest as base models, and Logistic Regression as the meta-learner. The base models were first trained using the training set, and their predictions on the validation set were then used as input features to train the Logistic Regression meta-learner.

Stacked ensemble learning was used to leverage the differential strengths of the individual base models and potentially enhance prediction accuracy and reliability by learning the optimal way to blend their predictions (Wolpert, 1992). The algorithmic implementation of the stacked ensemble learning approach is represented in Algorithm 1 where step by step approach of implementing this approach for prediction of Parkinson disease were carefully highlighted.

This whole implementation of the model and feature selection was done using Python 3.9v in Google Collab environment due to its sophisticated cloud computing functionality.

```
Algorithm 1: Stacked Ensemble Learning for Parkinson's Prediction
Input:
  D: Original Dataset (Features X, Target Y)
  Base_Models_Types: [SVM, Random Forest, K-Nearest Neighbors]
  Meta_Model_Type: Logistic Regression
  Train_Test_Split_Ratio: Ratio for initial train/test split (e.g., 0.8 for training)
  Meta_Train_Split_Ratio: Ratio for splitting the main training set to create meta-model training data (e.g., 0.7 for base model training)
Output:
  Final_Predictions: Predicted labels for the Final Test Set (X_test_final)
1.  Initial Data Split (Overall Hold-out):
    Split the original dataset D into two main sets:
    (X_train_main, Y_train_main)  // Used for training all models (base and meta)
    (X_test_final, Y_test_final)  // The final unseen test set for overall evaluation
    (Ensure this split is stratified to preserve target class proportions.)
2.  Initialize Meta-Feature Matrices:
    Initialize Meta_Features_For_Meta_Train: An empty matrix to store predictions from base models for training the meta-model. Its dimensions will be (len(X_train_meta_input) x num_base_models).
    Initialize Meta_Features_For_Final_Test: An empty matrix to store predictions from base models for predicting on the final test set. Its dimensions will be (len(X_test_final) x num_base_models).
3.  Generate Meta-Features for Meta-Model Training (Level 0 - Internal Hold-out):
    // This step creates "in-fold" predictions for the meta-model's training.
    // It prevents data leakage by ensuring base models predict on data they didn't train on.
    Split (X_train_main, Y_train_main) further into two subsets:
    (X_base_train, Y_base_train)   // For training base models to generate meta-features
    (X_meta_train_input, Y_meta_train_target) // Data for which meta-features will be generated
    (Ensure this split is stratified.)
    For each base_model_type in Base_Models_Types:
        Create a new instance of base_model_type.
        Train this base model on (X_base_train, Y_base_train).
        Generate predictions (probabilities or raw outputs) for X_meta_train_input.
        Add these predictions as a new column to Meta_Features_For_Meta_Train.
```

Received: 06-04-2025 / Accepted: 11-06-2025 / Published: 20-06-2025

https://doi.org/10.70118/ujet.2025.0202.13

```
4.  Generate Meta-Features for Final Testing (Level 0 - Full Training Set Predictions):
    // This step generates predictions for the ultimate unseen test set (X_test_final).
    // Base models are trained on the full main training set (X_train_main).
    For each base_model_type in Base_Models_Types:
        Create a new instance of base_model_type.
        Train this base model on the *entire* (X_train_main, Y_train_main).
        Generate predictions (probabilities or raw outputs) for X_test_final.
        Add these predictions as a new column to Meta_Features_For_Final_Test.
5.  Train Meta-Model (Level 1):
    Create an instance of Meta_Model_Type (Logistic Regression).
    Train the Meta_Model on Meta_Features_For_Meta_Train, with Y_meta_train_target as its true labels.
6.  Generate Final Predictions:
    Use the trained Meta_Model to generate Final_Predictions from Meta_Features_For_Final_Test.
```

2.4 Gain ration for Feature Selection

Effective feature selection is a critical step in building robust and efficient machine learning models, especially for complex diseases like Parkinson's where a large number of telemonitoring vocal features might be collected. Not all features contribute equally to the prediction task, and the presence of irrelevant or redundant features can introduce noise, increase computational cost, and potentially lead to overfitting. In this study, **Gain Ratio** was employed as the primary metric for feature ranking, aiming to identify the most discriminative vocal features for Parkinson's disease prediction. The stepwise approach in employing gain ratio is represented with Algorithm 2.

```
Algorithm 2: Feature Selection using Gain Ratio Ranking
Input:
  D: Dataset (Features X, Target Y, where Y indicates Parkinson's presence)
Output:
  Ranked_Features: A list of features sorted by their Gain Ratio in descending order.
1.  Calculate Global Entropy of Target Variable:
    Calculate H_Y = Entropy(Y)
    (Using the formula: H(Y) = - sum(p_i * log2(p_i)) for all classes i in Y)
2.  Initialize Feature_Gain_Ratios:
    Create an empty list to store tuples of (feature_name, gain_ratio).
3.  For each Feature 'A' in X (the set of all features):
    a.  Initialize Split_Values_Counts: A dictionary to store counts for each unique value of Feature 'A'.
    b.  Initialize Sub_Dataset_Entropies: A dictionary to store entropy for each subset.
    c.  For each unique_value 'v' in Feature 'A':
        i.   Create Subset S_v where Feature 'A' == 'v'.
        ii.  Calculate H_S_v = Entropy(Y for S_v).
        iii. Store (v, H_S_v) in Sub_Dataset_Entropies.
        iv.  Count occurrences of 'v' in Feature 'A' and store in Split_Values_Counts.
    d.  Calculate Weighted_Avg_Entropy:
        Weighted_Avg_Entropy = 0
        For each unique_value 'v' in Feature 'A':
            proportion_S_v = Split_Values_Counts[v] / len(D)
            Weighted_Avg_Entropy += proportion_S_v * Sub_Dataset_Entropies[v]
    e.  Calculate Information Gain (IG_A):
        IG_A = H_Y - Weighted_Avg_Entropy
    f.  Calculate Split Information (SplitInfo_A):
        SplitInfo_A = 0
        For each unique_value 'v' in Feature 'A':
            proportion_S_v = Split_Values_Counts[v] / len(D)
            // Handle log2(0) case: if proportion_S_v is 0, this term is 0.
            If proportion_S_v > 0:
                SplitInfo_A -= proportion_S_v * log2(proportion_S_v)
    g.  Calculate Gain Ratio (GR_A):
        If SplitInfo_A == 0:
```

```
      GR_A = 0  // Avoid division by zero, or assign a very small value if IG_A > 0
      If IG_A > 0:
        GR_A = IG_A // In some implementations, if SplitInfo is 0 but IG is positive, GR is IG.
    Else:
      GR_A = IG_A / SplitInfo_A
  h.  Add (Feature_A_Name, GR_A) to Feature_Gain_Ratios.
4.  Rank Features:
  Sort Feature_Gain_Ratios in descending order based on their Gain Ratio values.
5.  Print Ranked Features:
  Output the sorted list of features with their corresponding Gain Ratio scores.
```

2.5 Performance Evaluation

The performance of all the machine learning models was evaluated using the hold-out evaluation technique. Using this technique, the dataset was separated into two distinct sets: a training set (70%) to train the models and a test set (30%) to evaluate their performance on unseen data. The data was separated so that the models were tested on data they were not trained on, providing a better estimate of their generalization capability. The performance of the models was evaluated using the following standard classification metrics:

i. Accuracy: The ratio of the instances correctly classified over the total number of instances.

ii. Precision: The ratio of positive predictions that were true over all positive predictions.

iii. Recall: The ratio of positive predictions that were true over all actual positive instances.

iv. F1-Score: The harmonic mean of precision and recall, which provides a well-balanced estimation of the model's performance, particularly when classes are imbalanced.

These were calculated for each model on the held-out test set to quantify their capacity for predicting Parkinson's disease from the selected voice features.

3.0 Results and Discussion

This chapter presents the findings of the experimental analysis conducted in this study, encompassing the performance of the K-Nearest Neighbor, Logistic Regression, Random Forest, and Stacked Ensemble machine learning algorithms in predicting Parkinson’s disease based on voice data. The results are presented through quantitative metrics, followed by a detailed discussion and interpretation of the findings in accordance with the research objectives and literature. The features ranked by the Gain ratio is presented in Table 1 for easier representation so as to show the raking of the features in the dataset used in this study. Only the top 10 features were considered in this study as the other features were not considered as they did not have major contribution to the prediction of Parkinson’s disease. The first top 5 features as ranked by Gain ratio was experimented with and the experimental result is presented in Table 2.

Table 1: Features as ranked by Gain Ratio

Ranking	Gain Ratio (Features)
1	'MDVP:Flo(Hz)'
2	'spread1'
3	'MDVP:APQ'
4	PPE'
5	NHR'
6	'spread2',
7	'MDVP:Fhi(Hz)'
8	'MDVP:RAP'
9	'Jitter:DDP',
10	'MDVP:Shimmer'

The K-Nearest Neighbor (k=3) algorithm takes the lead with a very high average accuracy, precision, F1-score, and recall of 91.0%. This means that with the top 5 features chosen by Gain Ratio, the points for individuals with and without Parkinson's disease would likely be well separated in the feature space.

KNN as a non-parametric algorithm can readily learn complex decision boundaries if such separations exist based on the chosen features. The consistently high scores on all measures indicate an even performance with minimal false positives and false negatives. Conversely, Logistic Regression had a relatively much lower overall average performance across all the measures at around 74.0-74.2%. Logistic Regression is a linear model, and its relatively poorer performance shows that the relationship between the top 5 features and the presence of Parkinson's disease might not be purely linear. Although still producing a reasonable accuracy in prediction, its failure to pick up potentially non-linear relationships in the voice data shows itself when compared to KNN.

The Random Forest algorithm had a significant superiority over Logistic Regression with an average accuracy of 87.6%, precision of 88.0%, F1-score of 88.0%, and recall of 88.0%. Random Forest is an ensemble decision tree learning method that can model complicated non-linear relationships and possess high resistance to overfitting. The superior performance compared to Logistic Regression supports the potential that non-linear associations between the selected features are crucial in predicting Parkinson's disease. Interestingly, the Stacked Ensemble model, the focal point of this research, achieved a performance remarkably close to the

Random Forest model, with average accuracy of 87.6%, precision of 88.0%, F1-score of 88.0%, and recall of 88.0%. While in ensemble methods, particularly in stacking, there is an expectation that improved performance can be achieved through combining the strengths of different base models, in this specific experimental setting where there were only the top 5 features under Gain Ratio ranking, the Stacked Ensemble failed to deliver much of an improvement over the Random Forest.

There are a number of reasons for this discovery. Firstly, the Gain Ratio's top 5 chosen features may already contain enough discriminatory information for one strong model such as Random Forest to successfully exploit. The extra complexity of the Stacked Ensemble, which is likely aggregating the predictions of several base learners (including perhaps Random Forest itself or models with the same underlying dynamics), may not have captured much more sophisticated patterns using this restricted feature set. Perhaps the base models within the ensemble were already performing at a near-optimal level using the information contained within these five features, and as such, the potential for the meta-learner within the stacked ensemble to provide substantial added value was limited.

Further, the choice of base learners for the Stacked Ensemble and how the meta-learner is trained are also very critical. The specific choice of models in the ensemble or training parameters may not have utilized the individual strengths of the base learners to the best in the framework of these selected features.

The consistent good performance of KNN and Random Forest/Stacked Ensemble suggests that Gain Ratio feature selection was effective in choosing a subset of highly relevant features from the voice dataset for the prediction of Parkinson's disease. The observation that there is little gain with Stacked Ensemble over Random Forest informs us that exploration is still required. Generally, although the top 5 selected features by the gain ratio had supported effective prediction of Parkinson's disease through KNN and Random Forest algorithms, the Stacked Ensemble in this instance was unable to display much significant gain. This is a testament to the importance of best feature choice and the fine nuanced nature of ensemble learning, where the potential of stacking could be larger using a stronger set of features or different combination of base models.

Similarly, the Top 10 features as ranked by Gain ratio was further experimented with to compare their performance as presented in Table 3

Table 3: Experimental results of the top 10 features as ranked by Gain ratio

S/N	Algorithms	Avg. Accuracy (%)	Avg. Precision (%)	Avg. F1-Score (%)	Avg. Recall (%)
1	K-Nearest Neighbour (k=3)	94.4	95.0	94.0	94.0
2	Logistic Regression	78.7	79.0	79.0	79.0
3	Random Forest	91.0	91.0	91.0	91.0
4	Stacked Ensemble	95.7	95	96	96

K-Nearest Neighbor (k=3) classifier also showed a huge performance improvement with a mean accuracy of 94.4%, precision of 95.0%, F1-score of 94.0%, and recall of 94.0%. This demonstrates that the 5 additional features, when considered by the KNN classifier, further enhance the separability between the classes within the feature space, thereby leading to more accurate classifications. The slight improvement in precision (95.0%) also suggests fewer false positives. Logistic Regression also had a boost in its performance metrics, with a mean accuracy of 78.7%, precision of 79.0%, F1-score of 79.0%, and recall of 79.0%. While still the worst performing algorithm of the ones tried, the improvement from approximately 74% to near 79% suggests that the additional features provided more linearly separable information or helped the linear model better approximate the relationship between the voice features and Parkinson's disease.

The Random Forest algorithm's performance also improved, achieving an average accuracy of 91.0%, precision of 91.0%, F1-score of 91.0%, and recall of 91.0%. This indicates that the inclusion of more of the top-ranked features allowed the ensemble of decision trees to learn a better and more accurate model, one capable of detecting more complex patterns in the data. That all the metrics consistently improved is a testament to the effectiveness of Random Forest for this classification task. Most remarkably, the Stacked Ensemble model had a tremendous performance increase when trained with the top 10 features. In particular, it achieved an average accuracy of 95.7%, precision of 95.0%, F1-score of 96.0%, and recall of 96.0%. This result is a clear outperformance of all the base models explored during this study, including the Random Forest that had earlier recorded comparable performance.

As is evident from the table, increasing the number of top features from 5 to 10 led to the enhancement of the average accuracy for every algorithm. The most improvement (8.1%) was exhibited by the Stacked Ensemble, jumping from the level of Random Forest performance to the highest accuracy among all models attempted. This strongly suggests that the remaining five features contained helpful, complementary information that was indeed leveraged by the Stacked Ensemble via its collective learning process. The meta learner of the ensemble likely benefited from the denser feature space to distinguish better among the predictions of the base models and hence achieve greater overall performance. The across-the-board boost for the Stacked Ensemble with the top 10 features indicates a more robust, stable prediction model with the top 10 features. This addresses the importance of feature selection and the idea that while the top 5 features were a good starting point, incorporating more of the highest-ranked features by Gain Ratio adds significant value to the model's ability to distinguish between those with and without Parkinson's disease based on their voice features.

These findings point to the potential of the Stacked Ensemble learning technique for Parkinson's disease prediction based on voice data, particularly if coupled with an effective feature selection technique like Gain Ratio that chooses a sufficient number of relevant features. The dramatic improvement with the top 10 features invites closer inspection of the optimal number of features and the respective contributions of the features to the Stacked Ensemble model's enhanced predictive power. Future research can explore even larger sets of top-ranked features and more closely analyze the interactions and relative contributions of the individual features within the ensemble framework.

4.0 Conclusion

This research study investigated the efficacy of a stacked ensemble learning approach in Parkinson's disease prediction using a voice dataset. The primary aim was to achieve the highest prediction accuracy by combining the strengths of different base machine learning models, such as K-Nearest Neighbor, Support Vector Machine, and Random Forest, with Logistic Regression as the meta-learner. The effect of feature selection, employing Gain Ratio to determine the 5 and 10 most prominent features, was also tested. The experimental results unequivocally prove the efficacy of the approach. The Stacked Ensemble learning model universally performed better than the individual machine learning models in all test metrics when trained on the 10 features chosen by Gain Ratio, recording the highest mean accuracy, precision, F1score, and recall. Furthermore, the study discovered that applying the top 10 features resulted in a significant improvement of prediction performance for all algorithms used, such as the Stacked Ensemble, compared to applying the top 5 features. This points to the importance of selecting an appropriate number of significant features in sound Parkinson's disease prediction.

The better performance of the stacked ensemble suggests that by making smart use of heterogeneous base model predictions, the meta-learner had managed to discern more subtle patterns and nuances of meaning in the voice data typical of Parkinson's disease than any single individual model was capable of under its own independent initiatives. Such a finding highlights the application of ensemble learning techniques, with stacking being pre-eminent among them, for enhancing the accuracy and reliability of vocal biomarker-based automated PD diagnostic systems. This research contributes to current literature on the application of machine learning to early and non-invasive detection of Parkinson's disease. By the demonstration of concept of a specific stacked ensemble architecture and the positive impact of using a greater number of highly applicable features derived with Gain Ratio, this research provides valuable information for future development and research in this context. The subsequent work would further investigate the influence of including even more top-ranking features, and studying various base learner-meta-learner combinations under the stacked ensemble. Further investigations into other innovative feature selection algorithms are also on the agenda. It would be a key milestone towards developing practicable and credible tools for the screening and diagnosis of Parkinson's disease to extend the assessment of the proposed framework's generalizability on alternative voice datasets as well as under real-world clinical conditions.

References

Ali, L., Zhu, C., Golilarz, N. A., et al. (2021). Reliable Parkinson's disease detection by analyzing handwritten drawings: Construction of an unbiased cascaded learning system. *Computers in Biology and Medicine, 138*, 104835. https://doi.org/10.1016/j.compbiomed.2021.104835

Benba, A., Jilbab, A., & Hammouch, A. (2020). Voice analysis for detecting patients with Parkinson's disease using the hybridization of the best acoustic features. *International Journal of Speech Technology, 23*(4), 753–763. https://doi.org/10.1007/s10772-020-09747-2

Little, M. A., McSharry, P. E., Roberts, S. J., et al. (2023). Suitability of dysphonia measurements for telemonitoring of Parkinson's disease. *IEEE Transactions on Biomedical Engineering, 56*(4), 1015–1022. https://doi.org/10.1109/TBME.2008.2005954

Ozturk, N., & Ozturk, F. (2020). Parkinson's disease diagnosis using KNN algorithm with different distance metrics. *International Journal of Medical Informatics*, *134*, 104038.

Pereira, R. S., Pereira, J. C. R., Weber, S. A. T., Hook, C., & da Silva, R. F. (2018). Detecting Parkinson's disease with Random Forest algorithm. *Journal of Healthcare Engineering*, *2018*, 1-7.

Poewe, W., Seppi, K., Tanner, C. M., et al. (2022). Parkinson disease. *Nature Reviews Disease Primers, 3*(1), 17013. https://doi.org/10.1038/nrdp.2017.13

Polikar, R. (2012). Ensemble learning. In *Wiley encyclopedia of electrical and electronics engineering* (pp. 1-14). John Wiley & Sons, Inc.

Rusz, J., Hlavnička, J., Tykalová, T., et al. (2021). Automated speech analysis in early untreated Parkinson's disease: Relation to gender and dopaminergic transporter imaging. *European Journal of Neurology, 28*(3), 884–891. https://doi.org/10.1111/ene.14622

Received: 06-04-2025 / Accepted: 11-06-2025 / Published: 20-06-2025

https://doi.org/10.70118/ujet.2025.0202.13