Journal of Neonatal Surgery
ISSN(Online): 2226-0439
Vol. 14, Issue 17s (2025)
https://www.jneonatalsurg.com

OPEN ACCESS

A Hybrid Wrnn and Dbn-Based Approach (Hymod) For Parkinson's Disease Detection Using Voice Data

Ramya N[1], Dr. S. Devi Suganya[2]

[1]Ph. D (Research Scholar), Department of Computer Science, Vellalar College for Women (Autonomous), Affiliated to Bharathiar University, Erode, Tamil Nadu, India.

Email ID: nramyainfotech@gmail.com

[2]Assistant Professor, Department of Computer Science, Vellalar College for Women (Autonomous), Affiliated to Bharathiar University, Erode, Tamil Nadu, India

Email ID: sdevisuganya@gmail.com

Cite this paper as: Ramya N, Dr. S. Devi Suganya, (2025) A Hybrid Wrnn and Dbn-Based Approach (Hymod) For Parkinson's Disease Detection Using Voice Data. *Journal of Neonatal Surgery*, 14 (17s), 398-412.

ABSTRACT

Parkinson's Disease (PD) is a progressive neurodegenerative disorder that requires early and accurate detection for effective treatment. Existing approaches for PD detection have utilized deep learning models, incorporating techniques like Synthetic Minority Oversampling Technique (SMOTE) for data balancing and min-max normalization for feature scaling. However, min-max normalization can be sensitive to outliers, potentially skewing the model's performance. Additionally, traditional classifiers may struggle with feature selection, leading to suboptimal results and increased risk of overfitting. The current models, while effective, face challenges with generalization and the accurate detection of PD, particularly when working with sequential voice data, where temporal dynamics are crucial for diagnosis. This study suggests an approach for addressing these problems improved hybrid model (HYMOD) combining a Weighted Recurrent Neural Network (WRNN) and Deep Belief Network (DBN). The proposed method applies SMOTE for data balancing, but replaces min-max normalization with Z-Score normalization to mitigate the impact of outliers and ensure more stable model convergence. An entropy-based butterfly optimization the feature selection process uses an algorithm, improving model efficiency and focusing on the most relevant features, reducing noise and redundant data. By leveraging the sequential processing capability of WRNN and the deep feature extraction of DBN, the hybrid model significantly outperforms existing methods, achieving superior accuracy, precision, recall, and F1-scores for early PD detection. This enhanced model, through its innovative integration of advanced pre-processing, feature selection, and classification techniques, offers a more robust solution for reliable PD diagnosis and timely intervention..

Keywords: *Parkinson's Disease (PD), Deep Learning, Weighted Recurrent Neural Network (WRNN), Deep Belief Network (DBN), Synthetic Minority Oversampling Technique (SMOTE), Z-Score Normalization, Entropy-Based Butterfly Optimization Algorithm.*

1. INTRODUCTION

Parkinson's disease (PD) is one of the most prevalent neurological diseases, affecting millions individuals worldwide [1,2]. It presents with motor-related symptoms like tremors, muscle stiffness, and slowed movements, as well as non-motor issues such as cognitive deterioration and speech difficulties. Detecting PD at an early stage is vital for effective treatment and management, as prompt medical intervention can greatly enhance the quality of life for those affected. However, traditional diagnostic methods are largely subjective, relying on clinical assessments that can be prone to human error and variability. Consequently, the need for automated, objective methods for early PD detection [3].

The development of machine learning algorithms has enabled the detection of subtle variations in voice patterns that may indicate Parkinson's Disease (PD). By analysing acoustic features from voice recordings, these models are able to differentiate between people with PD and healthy individuals [4], providing a scalable and cost-effective diagnostic solution. This method offers the potential for continuous monitoring of disease progression, which is essential for optimizing treatment strategies over time. The use of voice data for PD detection is a significant innovation in healthcare, allowing for earlier and more accurate diagnoses while reducing reliance on invasive and expensive testing methods. Furthermore, it holds promise for improving patient quality of life through more personalized care. Medical diagnosis has showed potential due to recent developments in deep learning and machine learning, particularly with the use of voice data for identifying PD [5]. Voice data is especially useful in PD detection; as vocal changes are often among the earliest symptoms of the disease. Deep learning models, including SVM, ANN, RNN and CNN, have been widely employed to analyse voice patterns and detect

early signs of PD. While these models have demonstrated strong predictive power, challenges remain. Existing methods often suffer from data imbalance issues, inefficient feature selection, and sensitivity to outliers, all of which can negatively impact the generalizability and performance of the model.

In prior research, data balancing has been addressed using the SMOTE, while feature scaling was handled through min-max normalization. However, min-max normalization is highly sensitive to outliers, which can distort the range of the features and degrade model accuracy. Additionally, the most important characteristics may not be identified through conventional feature selection methods, leading to overfitting or under fitting of the model. These constraints limit the development of a reliable and effective system for PD detection, especially when working with sequential voice data, where the temporal relationships between data points are critical.

To overcome these issues, this paper proposes a HYMOD combining a WRNN and DBN. The WRNN is effective in processing sequential data and capturing the temporal dependencies in voice signals, while the DBN offers powerful deep feature extraction capabilities. In this approach, SMOTE is used to address data imbalance, but min-max normalization is replaced with Z-Score normalization, which is less sensitive to outliers and provides better convergence during training. Feature selection is performed using an entropy-based butterfly optimization algorithm, which ensures that the selection of the most relevant characteristics enhances the classification model's accuracy and efficiency.

The following are the paper's primary contributions:

A novel hybrid WRNN-DBN model that integrates sequential processing and deep feature extraction for robust PD detection (HYMOD).

The implementation of an entropy-based butterfly optimization algorithm for efficient feature selection, reducing redundancy and enhancing classifier performance.

The use of SMOTE for data balancing and Z-Score normalization to handle outliers and improve model stability

The proposed approach is anticipated to surpass current models more reliable and reliable accuracy, precision, recall, and F1-score of a method for the early detection of Parkinson's disease. The paper proceeds as follows: Section 2 provides a comprehensive review of pertinent literature, while Section 3 explains the recommended methodology. The experiments carried out for performance and outcome assessment are described in Section 4. Finally, Section 5 concludes up the analysis and talks about the suggested approach., offering insights for future work.

2. RELATED WORK

Almasoud et al [2022] [6] In including a recurrent neural network (RNN) into the GLSTM's batch normalizing layer and refining the network's hidden layer using the Adaptive Moment Estimation (ADAM) technique, the proposed categorization model has received enhancements. To demonstrate the importance of feature engineering, the suggested system uses a Sparse Auto-Encoder (SAE) to extract dynamic speech features and Linear Discriminant Analysis (LDA) to reduce dimensions. Energy transitions from voiced to unvoiced segments (offset) and from unvoiced to voiced segments (onset) are analyzed to extract these dynamic properties. To analyze the PD datasets, 10-fold cross-validation is used, ensuring that no samples overlap.

Abd El Aal et al [2021] [7] is a technique for early PD patient identification that uses speech features and an RNN together with long short-term memory (LSTM). This model employs the ADAM optimizer after the hidden layers and a batch normalization layer to increase classification accuracy. The method is tested on two benchmark speech feature datasets, include data from both healthy individuals and PD patients.

Al-Fatlawi et al [2016] [8] Parkinson's disease is classified using a Deep Belief Network (DBN), which consists of an output layer after two layered Restricted Boltzmann Machines (RBMs). A two-phase learning procedure is needed to optimize the network's parameters. Unsupervised learning using RBMs is used in the first phase to address problems caused by randomly initialized weight values. The model is improved in the second stage by using the backpropagation algorithm as a supervised learning method. The effectiveness of the suggested strategy is shown by contrasting its performance with that of many other approaches and related research. The suggested methodology surpasses all other methods assessed, in an overall testing accuracy of 94%.

Qasim et al [2021] [9] Unbalanced datasets, such those related to Parkinson's disease (PD), were handled using a hybrid feature selection approach. Class imbalance was addressed using the SMOTE. Recursive feature elimination (RFE) and principal component analysis (PCA) were used to get eliminate of feature conflicts and reduced processing time. Materials and Methods: Classification models such as Bagging, K-Nearest Neighbor (KNN), Multilayer Perceptron, and Support Vector Machine (SVM) were constructed using acoustic datasets from PD patients as well as data from healthy control participants.

Pramanik and Sarker [2021] [10] Various data pre-processing techniques were applied, including data standardization using

Z-Score, multicollinearity analysis, and techniques for reducing dimensionality to enhance data. PD was classified using K-Nearest Neighbour, Support Vector Machine, Random Forest, AdaBoost, and Logistic Regression. Hyperparameter optimization is necessary to maximize classifier performance and preserve class balance in the unbalanced dataset, k-fold cross-validation, and grid search were conducted. The suggested model surpassed earlier research on the same dataset by around 8%, reaching a peak accuracy of 94.10%.

Khaskhoussy and Ayed [2022] [11] suggested categorizing data through the use of machine learning methods like Support Vector Machines (SVM). The database utilized for the research included voice recordings from both PD sufferers and healthy persons. The research investigated three types of features. First, they examined Mel Frequency Cepstral Coefficients (MFCC). Second, they utilized deep features extracted through an AutoEncoder (AE). Additionally, they developed novel features using the Gaussian Mixture Models-Universal Background Model (GMM-UBM) framework to derive MFCC-GMM features. The combination of AE-generated deep features and MFCC-GMM features consistently delivered the highest accuracy in detection performance.

3. PROPOSED METHODOLOGY

The proposed methodology involves several steps, including feature selection, categorization, and data pre-processing. The proposed approach starts with Data Pre-processing, in which a more balanced dataset is produced by using the SMOTE approach to create artificial entities for the underserved class to rectify the class imbalance. Following this, using Z-Score normalization, the characteristics are standardized, aiding in better model convergence.

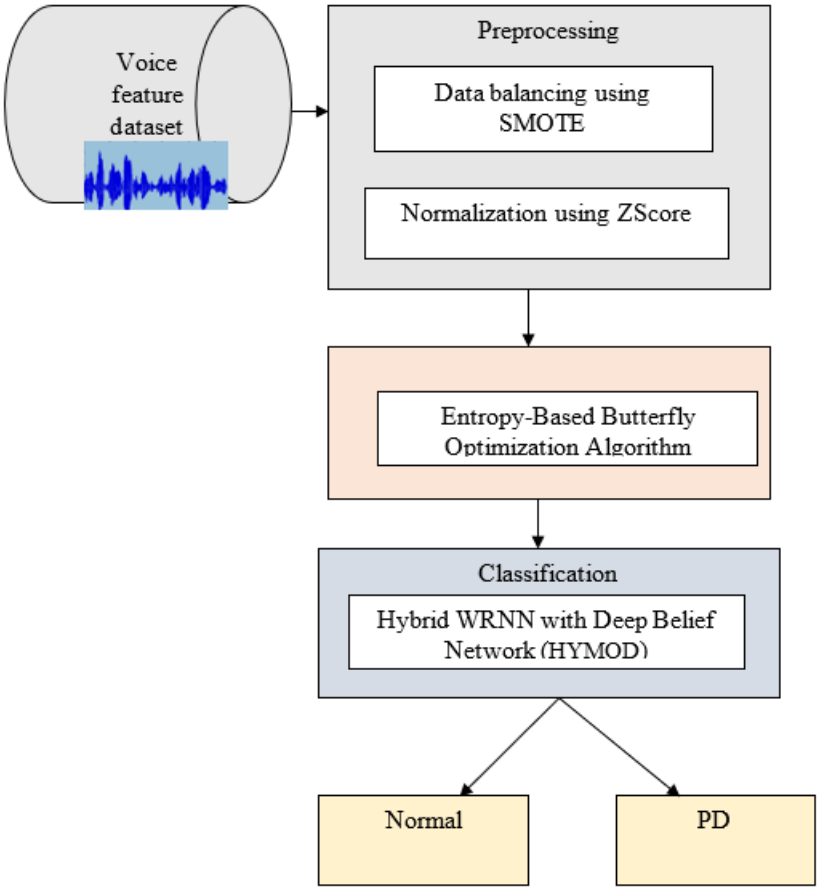

FIGURE 1. THE PROPOSED METHODOLOGY

Next, the Feature Selection step leverages the Entropy-Based Butterfly Optimization Algorithm to select enhancing classifier performance, decreasing the complexity of the data, and identifying the most relevant characteristics. Finally, the Classification phase (HYMOD) employs a Hybrid WRNN with Deep Belief Network (DBN), combining the strengths of WRNN for sequential data and DBN for complex pattern recognition. The proposed paradigm's overall structure is shown in Figure 1.

3.1. Dataset Description

Archive.ics.ucsd.edu/ml is where the voice data that was used in this study may be accessed by the general public. Supplied by Oxford University, the dataset comprises a range of acoustic speech features collected from 195 participants, including 147 diagnosed with Parkinson's disease [12,13]. It includes the total number of voice recordings per individual, where each feature corresponds to a distinct aspect of vocal measurement. The "status" column indicates health condition, where a negative value represents healthy individuals and a positive value indicates those with Parkinson's disease, allowing for differentiation between cases. Table 1 has a comprehensive description of the dataset.

TABLE 2. DETAIL OF PD DATASET

Dataset Characteristic	**Multivariate**
Counts of Instances	197
Attribute Characteristics	Real
Counts of Attributes	23
Missing Values	N/A
Made by	Max Little of the University of Oxford
Associated Tasks	Classification
Types of Classification	Binary {0 for healthy and 1 for PD patient}

3.2. Data Balancing by Synthetic Minority Over-Sampling Technique (SMOTE)

Oversampling was used during pre-processing since the PD dataset the research examined was imbalanced. The process of overs is the process of match the majority class by increasing the number of minority class samples, and it is a data analysis strategy used to modify class distribution [14]. This is achieved by randomly replicating minority instances to boost their representation. By using linear interpolation to create synthetic samples for the minority class, SMOTE is a popular method for addressing class imbalance. The algorithm operates in two primary stages.

The first step involves identifying the k nearest neighbors by calculating the Euclidean distances between minority samples and sorting them in ascending order. Then, the k-nearest neighbors (kNN) are determined based on these distances, as calculated using Equation (1), which determines the Euclidean distance across n characteristics between a minority sample x and another minority sample y:

$$d(x,y)=\sqrt{\sum_{a=1}^{n}(x_a - y_a)^2} \quad (1)$$

In the second phase, artificial data is generated using interpolating between the two minority samples. A candidate from the kNN is randomly selected, and new data points are generated by combining the chosen neighbor (y) with the original sample (x). The interpolation process is defined by Equation (2) for the a-th attribute between x and y:

$$\text{SyntheticData}_a\,(x, y) = x_a + r \cdot (x_a - y_a) \text{ for } 0 \le r \le 1 \quad (2)$$

Where,

r- random number between 0 and 1

The method is repeated until the required amount of synthetic data is produced, using this formula across all n characteristics.

3.3. Normalize data using Z-Score normalization

A popular data pre-processing method for normalizing feature values in a dataset is Z-Score normalization, often referred to as standardization.

To normalize data using Z-Score normalization **[15]**, each feature in the dataset is adjusted for the standard deviation to be one and the mean to be zero. This technique is especially beneficial when the dataset contains features with varying scales, as all characteristics have an equal impact on the model training process.

The formula for Z-Score normalization is:

$$z = \frac{X-\mu}{\sigma} \quad (3)$$

Where:

- X is the original data point.

- μ is the dataset's mean.
- σ is the dataset's standard deviation.

This transformation standardizes the data by subtracting the standard deviation (σ) divided by the mean (μ) of each data point. Following Z-Score normalization, considering a standard deviation of 1 and a mean of 0, the data is appropriate for machine learning models that require normalized inputs for faster convergence and improved performance. The improved data goes through the feature selection phase after pre-processing. At the stage, optimal **features** that are most relevant for speech signals are selected from the pre-processed data.

3.4. Feature selection Using Entropy-Based Butterfly Optimization Algorithm

In this study, the Entropy-Based Butterfly Optimization Algorithm (EBFO) is employed for feature selection to extract the most significant attributes from the medical dataset. EBFO is an innovative nature-inspired optimization technique that mimics the foraging and mating behaviours of butterflies, aiming to improve classification performance in medical diagnosis tasks [16,17]. The algorithm is inspired by butterflies' ability to use their strong sense of smell to efficiently locate nectar sources, it is comparable to selecting the attribute that is most relevant for precise categorization. Scientific research has shown that butterflies possess a highly accurate ability to locate the source of fragrance (akin to improving classification accuracy).

A butterfly's fitness varies with the intensity of the scent it produces as it moves from one position to another. This is referred to as classification accuracy. Three key terms sensory modality (c), stimulus intensity (I), and power exponent (a) for optimum feature selection form the foundation of the EBFO Algorithm's the concept of identifying and processing the modality. The fitness (accuracy) of the EBFO Algorithm for feature selection from medical datasets is connected with I [18]. Equation (4) of the EBFO Algorithm uses these ideas to develop the fragrance according to the stimulus's physical intensity,

$$f = cI^a \tag{4}$$

where f is the degree to which other butterflies perceive the fragrance, or its observed magnitude, in is the input intensity, c is the sensory modality, which is defined by the precision of classification, and in is the modality-dependent power exponential. The range [0,1] is therefore a & c. Alternatively, if a = 0, then no one else is able to detect the fragrance that a butterfly produces. The value of a determines how the algorithm behaves in this section. The EBFO algorithm's overall performance and rate of convergence are significantly influenced by another crucial parameter, c. A search algorithm's representation of these ideas uses the following idealized behavioural characteristics of butterflies:

It is assumed that each butterfly releases a certain fragrance, allowing butterflies (representing features) to be attracted to one another.

Each butterfly either travels at random or is attracted to the strongest-fragrance, most appealing butterfly.

The objective or function landscape's form or structure affects how strong a butterfly's stimulus is.

The initialization phase, iteration phase, and final phase are the three primary steps of the EBFO algorithm's operation. The initialization phase, an iterative search for optimal features, and the final phase, which occurs when the best feature selection is found, are the first steps in each EBFO execution. During the initialization phase, the EBFO algorithm evaluates classification accuracy and explores the solution space. Additionally, parameter values required for the algorithm are defined in this stage.

Along with their matching fragrance and fitness evaluations, the butterflies' beginning positions which serve as features are selected at random inside the feature selection search area. The algorithm moves on to the iteration stage when the initialization stage is completed. The exact position of every butterfly are updated throughout every phase within the solution space, and their classification accuracy is subsequently assessed [19]. Initially, based on each butterfly's unique position inside the solution space, the algorithm determines its fitness values. Following this, each butterfly emits a fragrance at its location using Equation (4). A butterfly moves going toward the optimum possible result during the global search phase found so far (g∗), representing the optimal set of features, as described by equation (5),

$$x_i^{t+1} = x_i^t + (r^2 \times g^* - x_i^t) \times f_i * ECE_W \tag{5}$$

Iteration number ith butterfly uses x_i^t to represent the solution vector x_i. f_i and r∈[0,1] represent the fragrance of the ith butterfly, and g^* the current iteration's most significant feature solution is a random integer. A representation of the local search phase is provided by equation (6),

$$x_i^{t+1} = x_i^t + \left(r^2 \times x_j^t - x_k^t\right) \times f_i * ECE_W \tag{6}$$

where x_j^t and x_k^t are j[th] and k[th] the feature selection solution space's butterflies. Equation (6) becomes a local random walk if $r \in [0,1]$ is a random number and x_j^t and x_k^t are members of the same swarm. To efficiently extract the best characteristics from the dataset, butterflies use the EBFO algorithm when searching for food and mate both locally and worldwide. Switch probability p alternates global and local searches. Until the stopping conditions are satisfied, the iteration phase continues. The method provides the best answer and its greatest fitness value at the conclusion of the iteration phase. Additionally, feature weight is incorporated into Equations (5) and (6) to help determine the ideal number of characteristics chosen from the medical gathering of data.

The EBFO algorithm selects the most relevant characteristics from the provided medical data in an effort to improve classifier accuracy. By reducing the divergence between two sample distributions, Cross Entropy (CE) analyses the optimum parameters of the probability distribution and helps solve optimization problems. The CE method is known for its strong global search ability, high adaptability, and robust performance.

$$CE = \frac{1}{N}\sum_{i=1}^{N} I_{s<r} \frac{f(x^i,v)}{g(x^i)} \quad (7)$$

with significance sampling density g(x) and a random sample from $f(x;v)$ denoted by x^i. To determine the optimum importance sampling density, the Kullback–Leibler divergence, it is presented to assess the distance between two sample distributions and is also referred to as the cross-entropy.

Algorithm 3 shows the general procedures necessary for the suggested EBFO algorithm. Algorithm 3 develops initial population based on medical dataset attributes (Step 1) and computes stimulus intensity I_i at x_i (Step 2) using sensor modality c and power exponent a from Step 3. An accurate categorization generates these factors. Following stopping criteria (Step 4), each butterfly's fragrance value is calculated (Step 6).

Then, in Step 8, identify the population's largest attribute, and in Step 10, produce a random integer, r. Move in the direction of the best butterfly using equation (5) if r<p; if not, move at random using equation (6). After updating a value (Step 17), people are assessed based on their new positions (Step 18). Finally, use the end while (Step 19) to conclude the operation. Figure 3 depicts the flowchart of the proposed Entropy Butterfly Optimization Algorithm (EBFO).

Algorithm 3: Entropy Butterfly Optimization Algorithm (EBFO)

Input: Medical datasets (Data sets on hepatitis, diabetes, heart disease, and fertility for Pima Indians)

Objective function: Classifier accuracy, $f(x)$, $x = (x_1, x_2, \ldots, x_{dim})$ $dim = no.of\ dimesnions$

Output: Selection of optimal features

- Using the number of features in the dataset, produce the initial population of n butterflies, $x_i = (i = 1,2, \ldots, n)$.
- The accuracy of classifying $f(x_i)$ provides the stimulus intensity I_i at x_i.
- Describe the power exponent (a), switch probability (p), and sensor modality (c).
- If the stopping requirements are not met, do
- For each population of butterfly fins, what
- Calculate fragrance for f using equation (5) and generate weight via entropy by equation (7)
- End for
- Find the best butterfly
- For each butterfly f in population do
- Generate random number r
- If $r < p$ then
- Move toward the path of the best butterfly (the optimal characteristics) by equation (5) and generate weight via entropy by equation (7)
- Else
- Move randomly using the equation (6)
- End if

- End for
- Update the value of a
- Consider an individual's abilities based on their new role.

Once the Entropy-Based Butterfly Optimization Algorithm (EBFO) completes its optimal feature selection, the output is a refined subset of the pre-processed data's most relevant and instructive aspects. These selected features are critical because they have the highest contribution to the classification task, making them optimal for enhancing model performance. The key benefit of this process is that it maximizes classification accuracy by removing redundant or irrelevant features, which can otherwise introduce noise or lead to overfitting.

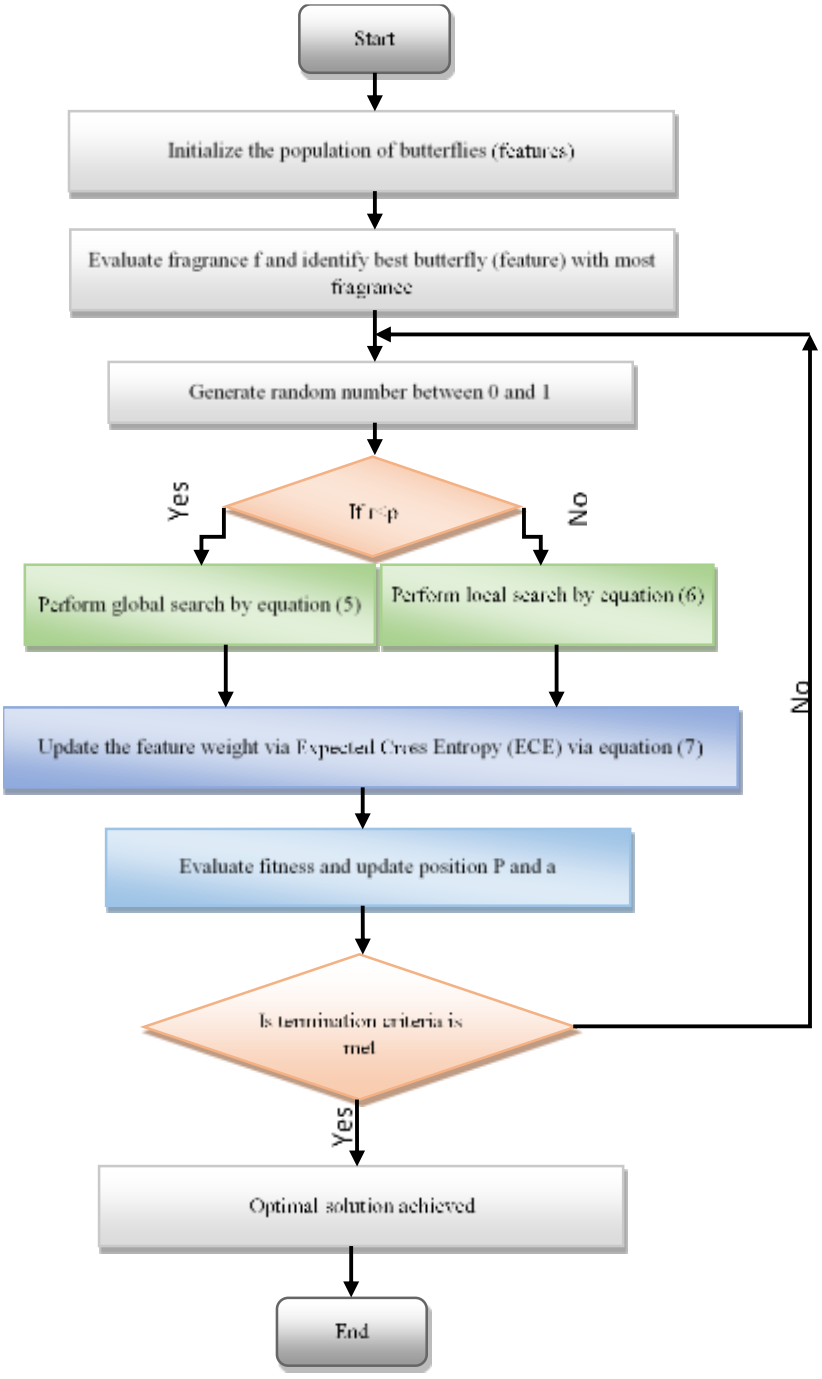

FIG 2 FLOWCHART OF ENTROPY BUTTERFLY OPTIMIZATION ALGORITHM (EBFO)

3.5. Classification using Hybrid Model (HYMOD)

In this hybrid approach, a Weighted Recurrent Neural Network (WRNN) is integrated with a Deep Belief Network (DBN) to enhance the classification accuracy of PD detection. The goal is to leverage the strengths of both models temporal processing from WRNN [20,21]and deep hierarchical feature extraction from DBN.

The WRNN first processes the input features, particularly those with temporal correlations, to extract high-quality feature representations. The recurrent nature of WRNN ensures that relevant temporal patterns in the data are captured. The output from WRNN is fed into the DBN, which further processes the extracted features through multiple hidden layers. Each RBM layer enhances the feature abstraction, allowing DBN to better model complex relationships within the data. Once the features have been learned, the top layer of DBN performs classification using the extracted feature set. The hybrid WRNN-DBN model provides a powerful mechanism for Parkinson's Disease detection, as WRNN [22] effectively handles temporal dependencies, while DBN captures high-level abstract features for precise classification.

3.5.1. Weighted Recurrent Neural Network (WRNN)

This method solves this problem by using an input layer fuzzy weight function. The output of unit I of class k is determined by the activation function $S_k(I_{ki})$, where I_{ki} is often denoted by the symbol θ. The inputs' weighted total per unit is represented by I_{ki}, which is defined as

$$\theta = I_{ki} = \sum_{j=0}^{L_{k-1}} f(w_{kij}) Z_{(k-1)j} \qquad (8)$$

The unit outputs of the kth layer are denoted by Z_{kj}. The kth layer's unit counts are implied by L_k. The fuzzy weights of the connection between neurons I and the j^{th} neuron in previous layers are denoted by $f(w_{kij})$. Figure 3 shows the RNN's structure.

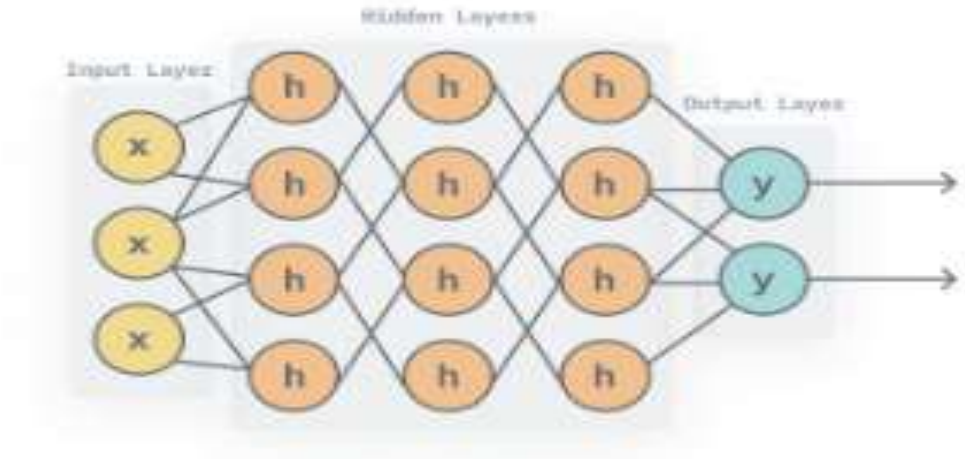

FIGURE 3.RNN ARCHITECTURE

3.5.2. Deep Belief Network (DBN)

Layers of hidden units are connected in DBN. Three types of hidden units are binary latent variables. DBN can learn to probabilistically recreate its inputs after being trained on a set of CTG samples. DBN may get further training to execute CTG classification under supervision. Each hidden layer of a DBN subnetwork acts as the visible layer for the subsequent layer, making it comparable to a RBM. RBM comprises a hidden layer, an input layer, and inter-layer connections.

Working Procedure of DBN: The initial step involves training a feature layer that can directly extract input characteristics from pixel data [23]. In the subsequent hidden layer, the model learns additional features by interpreting the outputs of the previous layer as input pixels. With each new layer of features added to the network, the lower bound on the log-likelihood of the training dataset increases, indicating improved model performance.

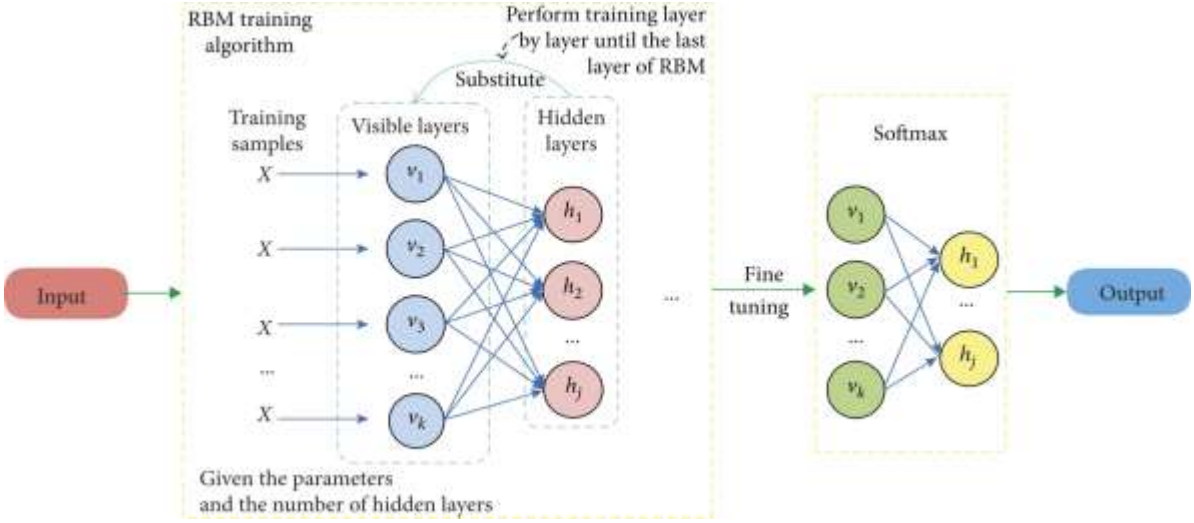

FIGURE 4. WORKING PROCEDURE OF DBN

The Deep Belief Network (DBN) is pre-trained using the Greedy Learning Algorithm (GLA), which follows a top-down generative weight approach through layer-by-layer training. The connections between the variables are defined by these generative weights in adjacent layers. In the top two hidden layers of the DBN, multiple iterations of Gibbs sampling are performed. The uppermost two hidden layers form a Restricted Boltzmann Machine (RBM) [24], where this stage focuses on effectively sampling from the RBM. Following this, ancestral sampling is used once across the remaining model layers to provide a selection from the units that are shown. A single bottom-up pass is then used to estimate the latent variable values at each layer. Greedy pretraining starts at the bottom layer by previously fine-tuning the generating weights in the opposite direction using an observed data vector.

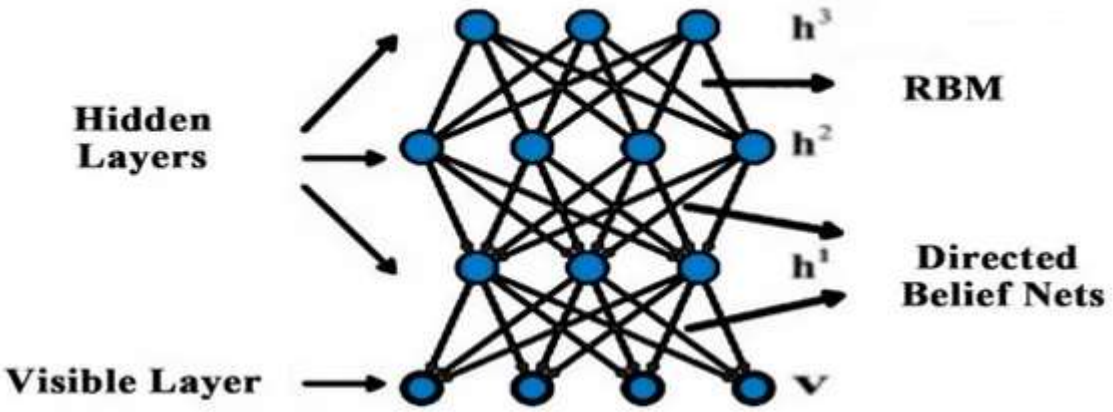

FIGURE 5. TRAINING EACH RBM LAYER

Training a Deep Belief Network (DBN) involves individually training each Restricted Boltzmann Machine (RBM) layer. This process begins by initializing the units and setting the parameters. The Contrastive Divergence algorithm, used for training, consists of two phases: positive and negative. The probabilities based on the weights and visible units are used for determining the hidden layers in the positive phase's binary states. Because it increases the probability of the training dataset, this step is referred to as the positive phase. Reducing the model's probability of producing its own samples is the goal of the negative phase, on the other hand. To train the entire Deep Belief Network (DBN) [25], the greedy learning approach is used. This method involves training each RBM individually, one at a time, until all RBMs in the network are fully trained.

Creating DBN Model: It enables a top layer to been trained for creating a class labels in input data vectors and it has been used to classify unknown data vectors. Equation (9) illustrates how the RBM's weights determined the joint probability distribution, which used an energy-based function of $\{v, h\}$,

$$En(v,h,\theta) = v^T Wh - a^T v - b^T h = \sum_{i=1}^{D_v}\sum_{j=1}^{D_k} w_{ij} v_i h_j - \sum_{i=1}^{D_v} a_i v_i - \sum_{j=1}^{D_k} b_j h_j \quad (9)$$

where θ={b_i, a_j, w_{ij}, w_{ij} weighs visible cell i to hidden cell j, and a_i and b_j are units i and j biases. Equation (10) calculates the RBM model's joint probability distribution across visible-hidden cells,

$$P(v,h;\theta) = \frac{1}{Z(\theta)}\exp\big(-E(v,h;\theta)\big) \quad (10)$$

where $Z(\theta)$is a normalizing constant value by i & j. It is computed by equation (11),

$$Z(\theta) = \sum_r \sum_h \exp(-E(v,h;\theta)) \quad (11)$$

RBM uses the energy equation to estimate input dataset probability. Cells i and j have conditional probability functions given by the joint probability distribution function. It is computed by equations (12-14),

$$P\big(h_j = 1 \big| v\big) = \delta(b_j + \sum_i v_i w_{ij}) \quad (12)$$

$$P(v_i = 1|h) = \delta(a_i + \textstyle\sum_j h_j w_{ij}) \quad (13)$$

$$\delta(x) = \frac{1}{1 + exp(-x)} \quad (14)$$

Considering the probability specified in Equation (13), each v_i is set to 1 to restore the input state. This process gradually updates the hidden units to reflect the features of the reconstructed input. The training process in the RBM is carried out through a maximization approach, which estimates as shown below, the training data's probability distribution with respect to the model parameters:

$$\text{maximize}\{b_j, a_i, w_{ij}\}\frac{1}{m}\textstyle\sum_{l=1}^{m}\log(P(v^l)) \quad (15)$$

where m is the training datasets' length. Consequently, a gradient descent method should to solve the objective function, which is a log-likelihood term. However, the existence of $Z(\theta)$ makes it difficult to apply the gradient computation of the log-likelihood component. Therefore, in a gradient computation, sampling techniques like contrasting convergence and persistent contrasting convergence may be used in its place. Once the DBN has been trained using greedy layer-wise training with the RBMs, the final top layer acts as the classifier. This layer takes the highest-level abstract features learned from the RBMs and uses them to perform the actual classification. For multi-class classification, a softmax function is used, while a sigmoid function is utilized for binary classification. These functions output the probability distribution across the possible classes, allowing the network to assign a class label to each input. After performing classification using the hybrid WRNN and DBN approach for PD detection, the next step is to conduct the experimental research section.

4. RESULTS AND DISCUSSION

This section presents the results of experiments of the suggested model. F-Measure, Precision, Recall, and Efficiency measure the HYMOD model's ability to diagnose Parkinson's disease. These indicators offer a thorough assessment of the model's effectiveness in accurately classifying speech signals for early disease detection. Table 2 displays the results of the performance comparison.

Performance Metrics

Precision

Precision may be defined as the proportion of probable positive forecasts that are really positive. It addresses the question: "Among all cases labeled as positive, how many are truly positive?" Low false positive rates are an indicator of high accuracy, which is very essential in medical diagnostics to prevent incorrectly diagnosing healthy individuals with PD.

Precisions, which are defined as the proportion of results that are relevant,

$$\text{Precision}=\frac{\text{Truepositive}}{\text{truepositive+falsepositive}} \quad (16)$$

Precision evaluates the reliability of the model's forecasts were favorable

2) Recall

Recall that the sensitivity, or true positive rate, is the percentage of actual positive instances that the model properly detected. "Out of all the true positive examples, how many were accurately recognized by the model?" is the question it addresses. A high recall is essential for ensuring that most positive cases, such as those with Parkinson's Disease, are identified, thereby reducing the number of false negatives.

$$\text{Recall}=\frac{\text{Truepositive}}{\text{truepositive+FalseNegative}} \quad (17)$$

In situations involving disease detection, recall is very important, where missing a diagnosis (false negatives) can have severe consequences.

3) Accuracy

Accuracy represents the overall percentage includes all forecasts both true positives and true negatives were among those that were accurate. It offers a broad evaluation of performance of the model in both classes. In imbalanced data sets when one class surpasses the other, accuracy may not always be accurate.

$$\text{Accuracy}=\frac{\text{Truepositive+TrueNegative}}{\text{Total}} \quad (18)$$

In cases where the dataset is balanced, accuracy serves as a good indicator of the model's general performance.

4) F measure

Precision and recall are combined into a single score termed the F1-Score, that provides an equitable assessment of a model's efficacy. Reducing both false positives and false negatives is crucial when dealing with unbalanced datasets, which is where it becomes extremely effective. As the precision and recall harmonic mean, the F1-Score offers a single value that reflects the trade-off between these two metrics. Essentially, the F-measure captures a single, comprehensive score that achieves a balance between recall and accuracy.

F-Measure=2 x [(Precision x Recall) / (Precision + Recall)] (19)

A higher *F-measure* indicates that the model has avoiding false positives while yet capturing true positives.

TABLE 2. RESULTS OF PERFORMANCE COMPARISON

Metrics		**Methods**		
	LSVM	**ANN**	**WRNN**	**HYMOD**
Accuracy (%)	90	96.7	98	99
Precision(%)	74	82	90	95
Recall(%)	88	92.42	93	95
F-measure (%)	76	87.01	89	93

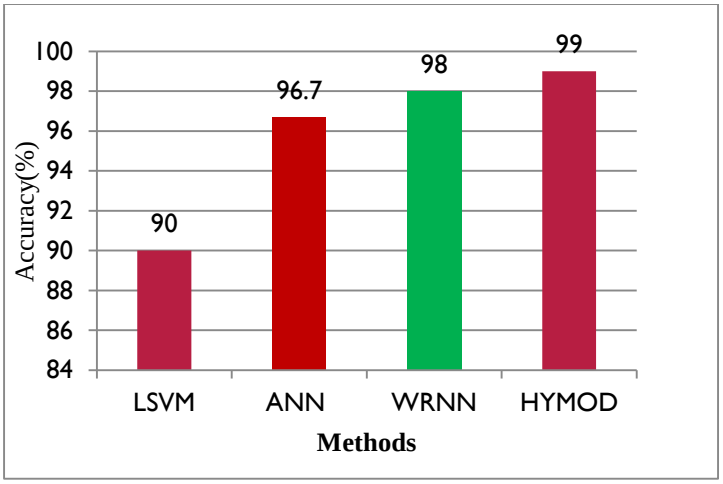

FIGURE 6. ACCURACY RESULTS COMPARISON

Figure 6 shows a comparison of the accuracy of the suggested HYMOD approach with the results of the existing LSVM, ANN, and WRNN methods for Parkinson's disease categorization. The graph's Y-axis shows the accuracy rates, while the X-axis depicts the various approaches. Present LSVM, ANN, and WRNN models, according to results show accuracies of 90%, 96.7%, and 99%, respectively, while the suggested HYMOD model achieves an incredible 99% accuracy.

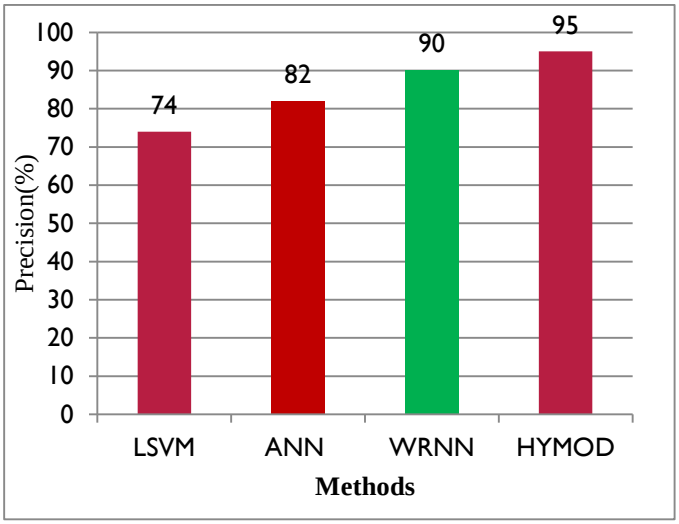

FIGURE 7. PRECISION RESULTS COMPARISON

Figure 7 presents an analysis comparing the accuracy efficiency indicators of the suggested HYMOD approach and the existing LSVM, ANN, and WRNN methods. The graph's X-axis relates to the different techniques, while the Y-axis displays the accuracy results. The proposed HYMOD model, which incorporates maximum and minimum normalization, demonstrates an improved accuracy of over 95%. In contrast, the existing LSVM, ANN, and WRNN models achieve accuracies of only 74%, 82%, and 90%, respectively, according to the results.

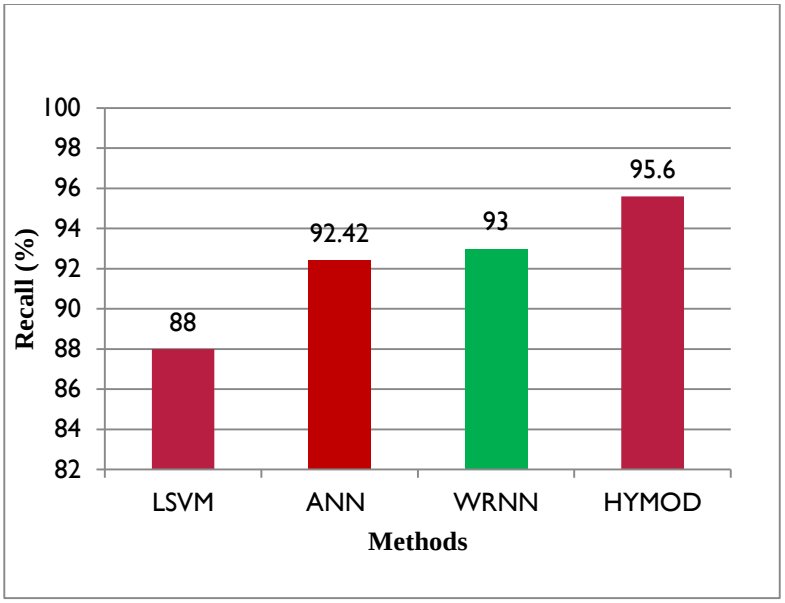

FIGURE 8. RECALL RESULTS COMPARISON

Figure 8 displays a comparison of recall metrics for the proposed HYMOD techniques against the existing LSVM, ANN, and WRNN methods. The X-axis of the figure lists the various techniques, while the Y-axis represents recall performance. According to the data, the HYMOD model excels with a recall rate of 95%, whereas the LSVM, ANN, and WRNN methods show recall rates of 88%, 92.42%, and 95.6%, respectively.

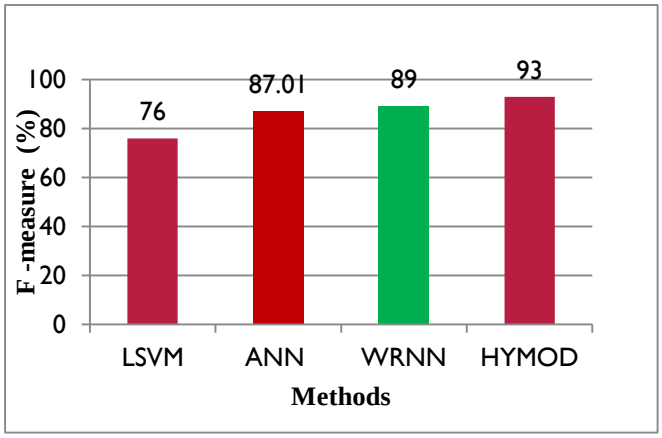

FIGURE 9. F -SCORE RESULTS COMPARISON

Figure 9 showcases a comparison of F-measure performance metrics between the proposed HYMOD approach and the existing LSVM, ANN, and WRNN techniques. The X-axis in the figure represents the different methods, while the Y-axis displays the F-measure scores. Based on the data, the HYMOD model achieves the highest F-measure score of 93%. In contrast, the current LSVM, ANN, and WRNN models yield F-measure scores of 76%, 87.01%, and 89%, respectively. This information suggests a need for feature selection and refinement to improve the F-measure outcomes further.

CONCLUSION

The proposed hybrid model (HYMOD), which combines a Weighted Recurrent Neural Network (WRNN) with a DBN, offers a significant improvement in early Parkinson's Disease (PD) detection. By using Z-Score normalization instead of min-max normalization, the model better handles outliers, leading to more stable performance. The entropy-based butterfly optimization algorithm enhances feature selection, focusing on the most relevant features and reducing noise. HYMOD achieves superior accuracy, precision, recall, and F1-scores compared to existing methods, thanks to its effective integration of advanced pre-processing, feature selection, and classification techniques. This model represents a more reliable approach for diagnosing PD early, improving disease management and enabling early intervention. To further increase the precision

of the model and generalizability, future research will investigate the use of larger data sets and more complex neural network topologies.

REFERENCES

[1] Schalling, E., Johansson, K. and Hartelius, L., 2018. Speech and communication changes reported by people with Parkinson's disease. Folia Phoniatrica et Logopaedica, 69(3), pp.131-141.

[2] Balestrino, R. and Schapira, A.H.V., 2020. Parkinson disease. European journal of neurology, 27(1), pp.27-42.

[3] Bhat, S., Acharya, U.R., Hagiwara, Y., Dadmehr, N. and Adeli, H., 2018. Parkinson's disease: Cause factors, measurable indicators, and early diagnosis. Computers in biology and medicine, 102, pp.234-241.

[4] Polychronis, S., Niccolini, F., Pagano, G., Yousaf, T. and Politis, M., 2019. Speech difficulties in early de novo patients with Parkinson's disease. Parkinsonism & related disorders, 64, pp.256-261.

[5] Braga, D., Madureira, A.M., Coelho, L. and Ajith, R., 2019. Automatic detection of Parkinson's disease based on acoustic analysis of speech. Engineering Applications of Artificial Intelligence, 77, pp.148-158.

[6] Almasoud, A.S., Eisa, T.A.E., Al-Wesabi, F.N., Elsafi, A., Al Duhayyim, M., Yaseen, I., Hamza, M.A. and Motwakel, A., 2022. Parkinson's detection using RNN-graph-LSTM with optimization based on speech signals. Comput. Mater. Contin, 72, pp.872-886.

[7] Abd El Aal, H.A., Taie, S.A. and El-Bendary, N., 2021. An optimized RNN-LSTM approach for parkinson's disease early detection using speech features. Bulletin of Electrical Engineering and Informatics, 10(5), pp.2503-2512.

[8] Al-Fatlawi, A.H., Jabardi, M.H. and Ling, S.H., 2016, July. Efficient diagnosis system for Parkinson's disease using deep belief network. In 2016 IEEE Congress on evolutionary computation (CEC) (pp. 1324-1330). IEEE.

[9] Qasim, H.M., Ata, O., Ansari, M.A., Alomary, M.N., Alghamdi, S. and Almehmadi, M., 2021. Hybrid feature selection framework for the Parkinson imbalanced dataset prediction problem. Medicina, 57(11), p.1217.

[10] Pramanik, A. and Sarker, A., 2021. Parkinson's disease detection from voice and speech data using machine learning. In Proceedings of International Joint Conference on Advances in Computational Intelligence: IJCACI 2020 (pp. 445-456). Springer Singapore.

[11] Khaskhoussy, R. and Ayed, Y.B., 2022. Speech processing for early Parkinson's disease diagnosis: machine learning and deep learning-based approach. Social Network Analysis and Mining, 12(1), p.73.

[12] Karaman, O., Çakın, H., Alhudhaif, A. and Polat, K., 2021. Robust automated Parkinson disease detection based on voice signals with transfer learning. Expert Systems with Applications, 178, p.115013.

[13] Varghese, B.K., Amali, D. and Devi, K.S., 2019. Prediction of parkinson's disease using machine learning techniques on speech dataset. Research Journal of Pharmacy and Technology, 12(2), pp.644-648.

[14] Camacho, L., Douzas, G. and Bacao, F., 2022. Geometric SMOTE for regression. Expert Systems with Applications, 193, p.116387.

[15] Lillig, R., Ophey, A., Schulz, J.B., Reetz, K., Wojtala, J., Storch, A., Liepelt-Scarfone, I., Becker, S., Berg, D., Balzer-Geldsetzer, M. and Kassubek, J., 2021. A new CERAD total score with equally weighted z-scores and additional executive and non-amnestic „CERAD-Plus "tests enhances cognitive diagnosis in patients with Parkinson's disease: Evidence from the LANDSCAPE study. Parkinsonism & related disorders, 90, pp.90-97.

[16] Sabeena, B. and Sivakumari, S., 2021. Parkinson's Disease Classification Using Fuzzy-Based Optimization Approach And Deep Learning Classifier. Turkish Online Journal of Qualitative Inquiry, 12(5).

[17] Nalluri, M.R., Kannan, K., Gao, X.Z. and Roy, D.S., 2020. Multiobjective hybrid monarch butterfly optimization for imbalanced disease classification problem. International Journal of Machine Learning and Cybernetics, 11, pp.1423-1451.

[18] Wang, G.G., Deb, S. and Cui, Z., 2019. Monarch butterfly optimization. Neural computing and applications, 31, pp.1995-2014.

[19] Tubishat, M., Alswaitti, M., Mirjalili, S., Al-Garadi, M.A. and Rana, T.A., 2020. Dynamic butterfly optimization algorithm for feature selection. IEEE Access, 8, pp.194303-194314.

[20] Fujita, T., Luo, Z., Quan, C., Mori, K. and Cao, S., 2021. Performance evaluation of RNN with hyperbolic secant in gate structure through application of Parkinson's disease detection. Applied Sciences, 11(10), p.4361.

[21] Nagasubramanian, G. and Sankayya, M., 2021. Multi-variate vocal data analysis for detection of Parkinson disease using deep learning. Neural Computing and Applications, 33(10), pp.4849-4864.

[22] Rizvi, D.R., Nissar, I., Masood, S., Ahmed, M. and Ahmad, F., 2020. An LSTM based Deep learning model for voice-based detection of Parkinson's disease. Int. J. Adv. Sci. Technol, 29(8).

[23] Shen, T., Jiang, J., Lin, W., Ge, J., Wu, P., Zhou, Y., Zuo, C., Wang, J., Yan, Z. and Shi, K., 2019. Use of overlapping group LASSO sparse deep belief network to discriminate Parkinson's disease and normal control. Frontiers in neuroscience, 13, p.396.

[24] Shen, L., Shi, J., Dong, Y., Ying, S., Peng, Y., Chen, L., Zhang, Q., An, H. and Zhang, Y., 2020. An improved deep polynomial network algorithm for transcranial sonography–based diagnosis of parkinson's disease. Cognitive Computation, 12, pp.553-562.

[25] Alshammri, R., Alharbi, G., Alharbi, E. and Almubark, I., 2023. Machine learning approaches to identify Parkinson's disease using voice signal features. Frontiers in artificial intelligence, 6, p.1084001...