

PARKINSON'S DISEASE AND AUTISM SPECTRUM DISORDER DIAGNOSIS USING SIGNIFICANT FEATURES AND ENSEMBLE DEEP LEARNING

DR. A. SYED HAROON ¹, DR JAISINGH W ², Dr. KAARTHIEKHEYAN V,

Dr. LEELAMBIKA KV⁴

^{1&2} School of Information Science, Presidency University, Bangalore, Karnataka.

³Department of Management Sciences, Hindusthan College of Engineering and Technology, Coimbatore, Tamilnadu -641032.

⁴School of Computer Science and Engineering, Presidency University, Bangalore, Karnataka.

E-mail: ¹ syedharoonmca@gmail.com, ² jaisinghw@gmail.com

ABSTRACT

More than 6 million people globally suffer from progressive Neurological Disorders (NDD) such as Parkinson's disease (PD). Furthermore, a variety of developmental impairment disorders that have an impact on a patient's capacity for social interaction and communication are collectively referred to as Autism Spectrum Disorder (ASD). Conventional methods for detecting PD and ASD, however, are frequently manual and need expertise. To solve this issue, a prior study presented an Ensemble Feature Selection (EFS) method based on the Fuzzy-based Beetle Swarm Optimization Algorithm (FBSOA) and Score-based Artificial Fish Swarm Algorithm (SAFSA). This technique chooses the most significant features from the database to increase the PD detection rate. First, Min-Max Normalization (MMN) was used to normalize the dataset's scale. Dimensionality reduction (DR) is achieved by the application of Binomial Cumulative Distribution Function-based Principal Component Analysis (BCDPCA). The optimal features for PD classification are then found via Ensemble-based FS (EFS). To classify PD, Ensemble Learning (EL) classification approaches such as Fuzzy K-Nearest Neighbor (FKNN) hybrid classifier, Kernel Support Vector Machines (KSVM), Fuzzy Convolutional (NN) Neural Network (FCNN), and RF are employed. However, dataset which are used for this work has missing values which will leads to the misclassification error. And also, MMN does not handle outliers very well. Resolving those challenges, this study proposed an improved model for PD and ASD detection. Then first missing values are imputed using Most Common value (MCI). And data normalization will be done by ZSN. DR will be performed based on BCDPCA. Significant features are selected from the dimension reduced data using ensemble of Enhanced Chicken Swarm Optimization (ECSO) and Improved Whale Optimization (IWO). Finally, PD and ASD are detected using ensemble of FCNN and LSTM network. The suggested schema obtained better performances for obtained values of precision, recall, and f measure in experimental outcomes.

Keywords: *Parkinson's Disease (PD), Autism Spectrum Disorders (ASD), Detection, Z Score Normalization (ZSN), missing values, Long Short-Term Memory (LSTM) network, Chicken Swarm Optimization (CSO), Whale Optimization (WO).*

1. INTRODUCTION

The motor system in brain cells are seriously impacted by PD, degenerative ailments in central nervous system of humans, one of the most prevalent and progressive NDD, globally impacting 7 to 10 million people. Its effects, which include voice and speech difficulty, dementia, depression, slow thinking, rigidity, tremor, bradykinesia, and

other cognitive problems, can be classified as either motor or non-motor. Shortages in dopamine, neurotransmitters in human brain result in PD. In addition to other symptoms, speech impairments, such as slurs, mumbled or sluggish speeches affects between 60% and 90% of PD patients [1,2].

Stereotypical behaviors and impairments in social communication and reciprocal interactions are characteristics of the broad group of NDD

known as ASD. Over the previous 20 years, the prevalence of ASD has increased, reaching 1% of the global population. In order to maximize the results of therapy, particularly for children, convergent evidence suggests that an early diagnosis and intensive intervention are essential [3,4,5, 6].

ANFIS and NNs were employed for PD detection; NNs have certain limitations, such as the need for a lengthy training period and uncertainties regarding the Activation Function (AF) to be utilized in the Hidden Layer (HL), the cells count in the HL, and the amount of HL. To shorten the screening process and find ASD features more quickly, Alternating Decision Tree (ADTREE) was employed nowadays. High level of accuracy was attained by the utilization of Autism Diagnostic Interview, Revised (ADI-R) method. Through the type and parameters of (MF) Membership Function and output linear parameters, the performance of ANFIS is determined [7,8,9].

An EFS method based on the FBSOA and SAFSA was presented in previous work to overcome this problem. This technique chooses the most significant features from the database to increase the PD DR. First, MMN was used to normalize the dataset's scale. BCDPCA is used for DR. The optimal features for PD classification are then found via EFS. Using EL classification techniques, such as RF, FCNN, KSVM, and hybrid classifier of FKNN, PD classification is performed. However, dataset which are used for this work has missing values which will leads to the misclassification error. And also, MMN does not handle outliers very well.

To overcome those issues in this study proposed an effective model for detecting PD and ASD. In which first missing values are imputed using Most Common value (MCI). And data normalization will be done by ZSN. DR will be performed based on BCDPCA. Significant features are selected from the dimension reduced data using ensemble of ECSO and IWO. Finally, PD and ASD are detected using ensemble of FCNN and LSTM network.

2. RELATED WORKS

PD was analyzed and classified by Karan et al. [2018][10]. Speech features change during phonation in individuals with PD due to significant impairments in Vocal Folds (VF) and Vocal Tracts (VT). This work pertinent Speech Signals (SS) were extracted using Variational Mode Decompositions (VMD). SS were divided into

modes or sub-signals using VMD. PD was detected using multiple statistical variables, including energies and energy entropies, means, variances, skewnesses, and kurtosis. The experiment demonstrates that the VMD-based feature outperforms the Mel Frequency cepstral coefficient (MFCC). 96.29% classification accuracy is demonstrated by the suggested feature.

For creating synthetic data instances, Piecewise Cubic Hermite Interpolating Polynomial (Pchip) and Spline interpolation techniques was suggested by Abayomi-Alli et al. [2020][11]. While taking into account the computational difficulties of implementation on actual mobile phones, researchers further explore the reduction of feature dimensionality for efficient and real-time classification. Using Bidirectional LSTM (BiLSTM) deep learning (DL) networks, researchers compare the classification results with those of those classic Machine Learning (ML) approaches including SVM, DT, Logistic Regression (LR), KNN, and Ensemble Bagged Tree (EBT). They have added 571 synthetic data samples to the 195 data samples in the Oxford PD dataset for experimental validation.

In order to create new, highly effective features without Feature Transformation (FT), a novel hybrid feature learning technique was suggested by Li et al. [2017][12]. This technique combines the features of each subject's speech segments collectively. First, features and segments were combined to create hybrid features. Second, a number of criteria were used to choose the most effective hybrid features. Thirdly, PD was classified using the chosen hybrid features. Additionally, this manuscript adds a number of evaluation criteria to the FS process. The suggested approach can produce new features (hybrid features) with good classification accuracy, according to experimental data. Highly stable and significant features have been selected.

A SVM, k-NN, and Artificial NN (ANN) were employed by Ranjan and Swetapadma[2018][13] to detect PD. The handwriting of a variety of normal and PD patients is used to gather input features. ML-based techniques are provided the inputs and their corresponding targets. An investigation has been conducted to compare SVM, k-NN, and ANN. For all of the tested data, the suggested method's accuracy is found to be 100%. Thus, real-time scenarios can benefit from the execution of ML-based techniques.

The early detection of ASD in toddler, child, adolescent, and adult datasets was

accomplished by Akter et al. using several FT approaches. There are a few FT techniques, such as sine, Z-score, and log [2019] [14]. After testing a range of classification techniques on these updated ASD datasets, the outcomes were assessed. Results determined that Adaboost performed best for children's datasets, GImboost performed best for teenage datasets, and Adaboost performed best for adult datasets. SVM demonstrated the greatest performance for the toddler dataset. The most accurate classifications are generated by FT via the following: for toddler datasets, sine function is used. For children datasets and adolescent datasets, Z-score can be used. This can be used to apply a number of FS algorithms to identify the significant ASD risk factors for toddlers, children, adolescents, and adults. The resultant outcomes of analysis demonstrated that appropriately tuned ML techniques provided reliable diagnoses of ASD statuses implying model's utility in early identifications of ASD.

Kalman filter based learning framework was presented by Kushki et al. [2014] [15] to detect physiological arousals from heart activities. To evaluate the effectiveness of the system, data from a sample of children with ASD were utilized. The results showed that the system could identify anxiety-related activities of children, obtaining 99% in sensitivity and 92% in specificity on 146 genes'

and 237 SNPs used to develop genetic diagnostic classifiers. It accurately predicted the diagnosis of ASD in 85.6% of CEU cases and it was suggested by Skafidas et al [2014] [16]]. In an ethnically close Tuscan cohort, this classifier also identified 84.3% of cases; however, in a genetically distinct Han Chinese cohort (HAN), prediction accuracy was lower at 56.4%. The greatest impact on the classifier was caused by eight SNPs spread across 3 genes (KCNMB4, GNAO1, and GRM5), some of which functioned as vulnerability SNPs and others as protective SNPs. With a decrease in the SNP count that is examined in the framework, prediction accuracy dropped. When CEU individuals from the SFARI (ASD) and WTBC (controls) validation data sets were utilized, the diagnostic classifier predicted the diagnosis of ASD with 71.7% accuracy.

3. PROPOSED METHODOLOGY

This section discusses the suggested PD and ASD detection model. In this missing value are imputed using Most Common value (MCI) model. Data Normalization (DN) is performed using ZSN. DR is computed based on BCDPCA. FS is done by ensemble of ECSO and IWO. PD and ASD are detected using ensemble of FCNN and LSTM network. Figure 1 presents the overall structure of the suggested framework.

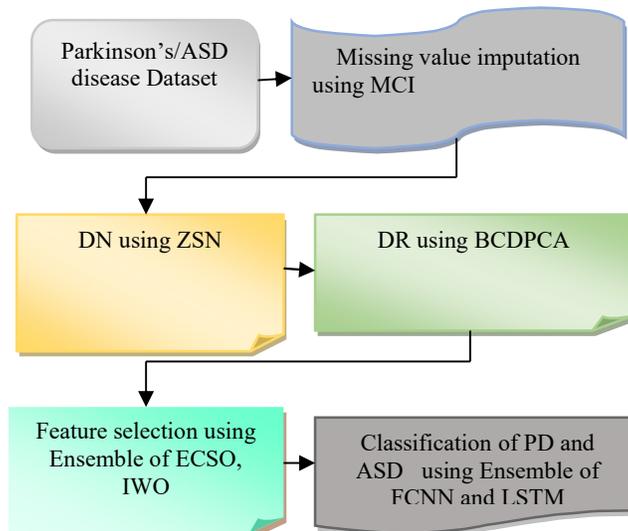


Figure: 1. Overall structure of the suggested framework

3.1. Dataset description

With the support of suggested methodologies, the psychological condition is investigated in this study using 2 datasets: PD and ASD. This work uses the

PD Classification database, which is available at UCI ML Source. The first database included 64 normal individuals (23 men and 41 women) aged between 41 and 82 years (61.1 years old 8.9 years), and 188 PD affected persons (107 men and 81 women) aged between 33 and 87 years (65.1 years old 10.9 years). There are 21 attributes and 292 instances in the second dataset ASD. Age, gender, ethnicity, country of residency, screening techniques, and so on are among the characteristics. Section 4 provides a detailed study of two datasets.

3.2. Missing value (MV) imputation using MCI

A comparatively easy procedure is used for imputing using the MCI: The most frequent value of this attribute is used in place of the MV for nominal attributes. When dealing with numerical attributes, the average value of the attribute is used in place of the MV.

3.3. DN with ZSN

One of the most common Data Mining (DM) techniques for improving the precision of ML methods is data pre-processing. Normalization is applied to all the data before to training and testing. This is done in order to prevent data from becoming excessively redundant. Data from several scales are converted to equal scales using normalizations with mean and Standard Deviation (SD) values of features A based on the formula in [17]:

$$v' = \frac{v - \underline{A}}{\sigma_A} \tag{1}$$

Here, SD of A can be denoted as σ_A . Mean of A can be denoted as \underline{A} . Each data entry's new and old values are denoted by v' and v .

3.4. DR with BCDPCA

After data normalization it needs to find a dimensionality reduced features to increase the classification accuracy. BCD-PCA is used in this study for DR. For spectrally compressing high-dimensional data, PCA is frequently employed. A novel set of variables that is generated by this approach called PC. By linear combination of initial variables, every PC are constructed. PCA provides effectiveness in capturing variations in chosen datasets by projecting data onto orthogonal subspaces with reduced dimensions. PCA is needed to normalize the original data with significant

observed scale variability in order to maintain this crucial information.

Algorithm 1 explains the manner in which the suggested approach built a BCDPCA for effective DR to address the loss of these essential data. The problem of representing low-dimensional features is as follows: let's assume that $X = (x_1, x_2, \dots, x_n)$ represents a $n \times N$ data matrix. Feature Vector (FV) with dimension $n \dots$. As a result, every x_i represents a Feature Vector (FV) with n dimensions.

Computing Scatter Matrix (SM)

Equation (2) below is employed to calculate the SM:

$$S = \sum_{k=1}^n (x_k - m)(x_k - m)^T \tag{2}$$

Here, the mean vector can be denoted as m .

Equation (3) illustrates the manner in which the suggested approach created a BCD for MV computation in order to raise the possibility of DR.

$$m = np \tag{3}$$

Here, the total number of observations can be denoted as n

And the probability of success is denoted as p

Calculating EigenVectors (EV) and corresponding EV

The eigenvalues, as derived from the Covariance Matrix (CM), were unquestionably scaled by the factor. Make sure equation (4) \sum has been satisfied that the EV-eigenvalue computation is accurate.

$$\sum v = \lambda v \tag{4}$$

Here the EV can be denoted as v . The Eigen value can be denoted as λ . The CM can be denoted as \sum .

To minimize the amount of information regarding the data distribution, avoid the v with the smallest λ . The most widely used technique is to choose the top k v by ranking the v according to their related λ . The two eigenvectors with the highest λ are combined to create the $d \times k$ -dimensional eigenvector matrix W^T .

In the final phase, use the system-generated 2X3-dimensional matrix W to transfer the data into the novel subspace $y = W^T \times x$.

Algorithm 1: BDPKA

Phase 1: d -dimensional sampled data.

Phase 2: Determine D -Dimensional Mean Vectors, or means of dimensions in databases.

Phase 3: Compute CM Of Entire database

Phase 4: Compute EV (E_1, E_2, \dots, E_D) And Related Eigenvalues ($\lambda_1, \lambda_2, \dots, \lambda_D$)

Phase 5: Rank EV By Reducing λ And choose K EV With Highest λ for obtaining $D \times K$ Dimensional Matrices W (columns specify EV)

Phase 6: Transfer samples onto the $y = W^T \times x$ using $d \times k$ EV matrix, expressed mathematically as $y = W^T \times x$ where y represents transformed $k \times 1$ -dimensional samples in $y = W^T \times x$, and x represents $d \times 1$ -dimensional vectors represent singular samples

3.5. Feature selection using Ensemble of Enhanced Chicken Swarm Optimization (ECSO) and Improved Whale Optimization (IWO)

This work utilizes ECSO and IWO for FS after DR, for selecting most relevant features from datasets and minimizes FS time consumptions.

CSO

An approach for bio-inspired Meta-Heuristic (MH) optimization is called CSO. The program emulates the individual chickens' behaviors as well as the hierarchical structure of a swarm of chickens. A Chicken Swarm (CS) is hierarchically ordered into multiple groups, with one RN and numerous HN and CN in every group. The laws of motion that apply to different types of chickens vary. Chickens' social lives are significantly influenced by a hierarchical order. The stronger hens will dominate the weaker ones in a flock. The group includes both the submissive HN and RN positioned at the edges, as well as the dominant HN that remain near the head RN. Traditional CSO will easily falls into the trap of local optimal features.

This approach utilizes the mutation operator in CSO to avoid this problem in this study. Flip bit mutation was employed in this study. Using the selected genome, this mutation operator flips the different bits. It is converted from 1 to 0 (i.e., if the genomic bit is 1) and vice versa [18].

Enhanced CSO (ECSO)

The following recommendations, which describe the behaviors of the chickens, are the basis of the ECSO framework:

1) The CS is divided into different groups. There is a dominant rooster, some HN and CN in every group following it.

2) The hierarchy of swarms are determined by Fitness Values (FV) of chickens; roosters, who possess highest FV and are group leaders, while chicks, who possess lowest FV are classified as individual chickens. Hens are categorized as others.

3) The following will remain unchanged in a group: the mother-child bond, dominance dynamic, and swarm hierarchy. These statuses are only updated every several (G) time steps.

4) The N virtual chickens that make up the swarm are separated into the following groups: RN-roosters, HN- hens, CN- chicks, and MN –mother hens. Locations in a D-dimensional space are used to represent each individual.

$$x_{i,j} (i \in [1, \dots, N], j \in [1, \dots, D]), \quad (5)$$

Rooster Movement: Equations (6) and (7) describe the manner in which RN with higher FV can search for food in a greater variety of locations than those with lower FV.

$$x_{i,j}^{t+1} = x_{i,j}^t * (1 + \text{Randn}(0, \sigma^2)) \quad (6)$$

$$\sigma^2 = \begin{cases} 1, & \text{if } f_i \leq f_k \\ \exp\left(\frac{f_k - f_i}{|f_i| + \epsilon}\right) & \text{otherwise } k \in [1, N], k \neq i \end{cases} \quad (7)$$

Here the chosen rooster with index i is denoted by $x_{i,j}$. To prevent zero-division errors, the computer uses ϵ , the smallest constant. k implies random selections of RN indices from groups of roosters. The related rooster x_i 's fitness values are represented by f and mathematically $\text{Randn}(0, \sigma^2)$ represents gaussian distributions with 0 for means where σ^2 implies standard deviations.

Hen movement: HN follow the RN in their flock in search of food. In addition, while being suppressed by the other chickens, they would also haphazardly steal the tasty food that they found. The more dominant HN would have an advantage over the timid ones in a competition for food. Equations (8) and (9) provide a mathematical formulation for these events.

$$x_{i,j}^{t+1} = x_{i,j}^t + S1 * r \text{Rand} * (x_{r_1,j}^t - x_{i,j}^t) + S2 * \text{Rand} * (x_{r_2,j}^t - x_{i,j}^t) \quad (8)$$

$$S1 = \exp\left(\frac{(f_i - f_{r_1})}{\text{abs}(f_i) + \epsilon}\right) \quad (9)$$

$$S2 = \exp\left(\frac{(f_{r_2} - f_i)}{\epsilon}\right) \quad (10)$$

Indices of chickens (RN or HN) from swarms are randomly selected using $r2 \in [1, \dots, N]$. The indices of RN, or i th HN's group-mates are $r1 \in [1, \dots, N]$ while Rand implies uniform random numbers between [0, 1].

Chick movement: In an attempt to get food, the chicks roam about their mother. Equation (11) is used to formulate this.

$$x_{i,j}^{t+1} = x_{i,j}^t + FL * (x_{m,j}^t - x_{i,j}^t) \quad (11)$$

The mother of the *i*th chick is located at $x_{m,j}$ such that $m \in [1;N]$. To account for the variations amongst each chick, the FL parameter indicates the rate at which a chick would follow its mother. Within the range [0, 2], FL is selected at random. The specified Fitness Function (FF) can be maximized by Search Space (SS) to identify optimal points. Due to high sizes of Feature Spaces (FS), one must use intelligent searching approaches, where features are represented by distinct dimensions with spans of 0 to 1. Given the training data, limit the number of features selected while optimizing classification efficiency throughout the validation set, and demonstrated the FF for CSO in eqn 12.

$$f_{\theta} = \omega * E + (1 - \omega) \frac{\sum_i \theta_i}{N} \quad (12)$$

Constant ω regulates classification weights in relation to selected feature counts and N stands for total feature counts, FF for a vector θ with 0/1 elements representing unselected / chosen features is f_{θ} whereas E represents classifier error rates.

The amount of features in the provided dataset matches the amount of variables used. Every variable has a restriction range of [0, 1], and when a variable value gets close to 1. Then the associated feature is a potential candidate for selection in the classification process. When calculating individual fitness, the variable that determines which particular features to consider is the threshold, as shown in equation (13).

$$f_{i,j} = \{1 \text{ if } X_{i,j} > 0.5 \text{ 0 otherwise,} \quad (13)$$

Here, search agent *i*th dimension value at dimension *j* is represented by X_{ij} . To ensure variable limits while updating the firefly location solution, simple truncation rule is used. At certain dimensions [0,1], new value may break the limiting constraints.

Algorithm2: ECSO

Input: PD, ASD dataset

Output: Optimized features

1. Set RN, HN, CN, MN, G;
2. Randomize swarm chickens
3. $X_i (i = 1,2,\dots,N);$

4. Set max iterations Tmax;
5. while $T < Tmax$ do for every iteration
6. if $T \% G = 0$ then
7. Generate hierarchical ranks chickens FV in the swarm;
8. Divide swarms into diverse groups and obtain relationships between CN and MN;
9. end
10. for every chicken X_i in the swarm do
11. if X_i is a roster then
12. Update X_i 's position with eqn 6;
13. end
14. if X_i is a hen then
15. Update X_i 's position with eqn 8;
16. end
17. if X_i is a chick then
18. Update X_i 's position with eqn 11;
19. end
20. Assess the novel solution with eqn 13;
21. If the novel solution is better than its previous one, update it;
22. end
23. end
24. Use FBM to the updated solution
25. Calculate the novel solution with eqn 13
26. end

WOA

WOA are nature-inspired population-based novel Stochastic Optimizations (SO) which employ groups of search agents for determining optimal solutions to solve optimization problems. Bubble-Net Hunting (BNH) of WOA which mimic behaviors of humpback whales hunting preys was employed where encircling, searches, and Bubble Net Attacking (BNA) are 3 general processes of WOA [19].

To select the best method of action, the whales encircle their prey, which includes fish, and then try to change their locations. The primary mathematical component of the WOA is illustrated by equations (14) and (15).

$$X(t+1) = X^*(t) - A * |C * X^*(t) - X(t)| \text{ if } p < 0.5 \quad (14)$$

$$X(t+1) = |C * X^*(t) - X(t)| * e^{bl} * \cos(2\pi t) + X^*(t) \text{ if } p \geq 0.5 \quad (15)$$

where the vector X represents the coordinates of every whale. T is the index of time or iterations. X^* has been identified as obtained best solutions.

Over the course of iterations, the coefficient vector $A=2a * (r-a)$; $C=2 * r$; a fall linearly from 2 to 0. r has values ranging from 0 to 1 and here r denotes the random vector. Depending on the specific path, the shape of the logarithmic spiral is determined by the

constant variable b and, its value is fixed to 1 in this work. The random number l ranges from -1 to 1.

The random number p , which ranges from 0 to 1, is used to update the locations of the whales in Equations (14) and (15). The probabilities are 50% and 50%, respectively in (14) and (15). Throughout optimizations whales follow defined or random paths equally. Vectors A have random values between $[-1, 1]$, although it can be greater or less than 1 in the searches during the bubble-net phase. Equation (16) represents the Search Mechanism (SM).

$$X(t + 1) = X_{rand} - A \cdot |C \cdot X_{rand} - X(t)| \quad (16)$$

By implementing WOA procedures for Global Searches (GS), random SM with values of $|A|$ larger than one highlights search processes. WOA searches start with creations of random solutions which are then applied iteratively for solutions. Searches continue until they reach a predetermined iteration counts.

Improved WOA (IWO)

Because they are two crucial components of an optimization algorithm, exploration and exploitation provide a good trade-off that allows the algorithm to escape local optima and arrive at a precise solution. A search agent's step size in WOA reduces linearly with the number of iterations. This step size is controlled by a parameter called A . Nevertheless, it has been observed that poor divergence limits WOA's ability to trap into local optimum at later iterations. This paper employs an IWO approach to address those problems. This includes the Levy Flight (LF) function to change the value A . It improves WOA's capacity for simultaneous exploration and exploitation.

Calculating the jump size in LF involves using the Levy PDF, a Power-Law (PL) function. The Levy distribution's mathematical formula is given below.

$$L(s, \gamma, \mu) = \begin{cases} \sqrt{\frac{\gamma}{2\pi}} \exp \exp \left[-\frac{\gamma}{2(s-\mu)} \right] \frac{1}{(s-\mu)^{3/2}} & \text{if } 0 < \mu < \\ \infty & \text{if } s \leq 0 \end{cases} \quad (17)$$

In this distribution, the position parameter can be denoted as μ , scale parameter can be denoted as γ , and sample collection parameters are denoted by s .

Algorithm :3. IWO

Input: PD, ASD dataset

Output: Optimal features

- START
- import data
 - set the positions of the whale population X

- calculate the fitness of every whale
 - set a and r , compute A and C
 - set X^* as the best hunter whale's position
 - set $t = 1$
 - while $t \leq \text{max iterations do}$
 - for every hunting whale do
 - if $p < 0.5$
 - if $|A| < 1$
 - update the present hunting whale's position by (14)
 - else if $|A| \geq 1$
 - randomly choose another search agent
 - update the present hunting whale's position by (16)
 - end if
 - else if $p \geq 0.5$
 - update the present hunting whale's position by (15)
 - end if
 - end for
 - update X^* if there is a better solution
 - $t = t + 1$
 - end while
 - output X^*
- END

The Majority Voting (MV) combination rule is applied during the voting process. Through the combination of ECSO and IWO for selection, a meta-algorithm engaged in Decision Making (DM). For each test scenario, more than half of the votes are obtained by the final output prediction. Each Feature Selection (FS) creates a prediction.

3.6. Detection and Classification of PD and ASD using Ensemble of FCNN and LSTM network

Selected features are the input for the classification and detection of PD and ASD process. In this work Ensemble of FCNN and LSTM network are used for this process.

Fuzzy CNN

Four layers: Input Layer (IL), Convolution Layer (CL), Pooling (POOL) Layer, and softmax layer are used by FCNN to ensure prediction accuracy while minimizing computations. It uses the IL, CL, softmax, and Output Layers (OL) of CNN.

(i) **Input Layer:** This layer is trained using $N \times k$ neurons, here N is the datalength and k is the variate value of the input data.

(ii) **CL:** The data supplied from previous layer was convoluted using convolution filters.

(iii) **Pooling Layer:** These layers minimize dimensionalities, samples, and parameters counts. Computations as parts of masking processes on

input matrices are done using Sliding Windows (SW) that move with convolution kernel sizes. Additionally, Fuzzy MF (FMF) are employed in weight computations.

(iv) Fully Connected (FC) or Softmax Layer: Activation Functions (AF) produce nonlinear outputs through the merging of linear networks.

Softmax functions (SF) are squashing functions where layers produce multi-class probabilities.

(v) OL: OL is FC to the feature layer. It outputs neurons or nodes based on n feature classes. In classifications, inputs are used by the highest output neuron to generate class labels.

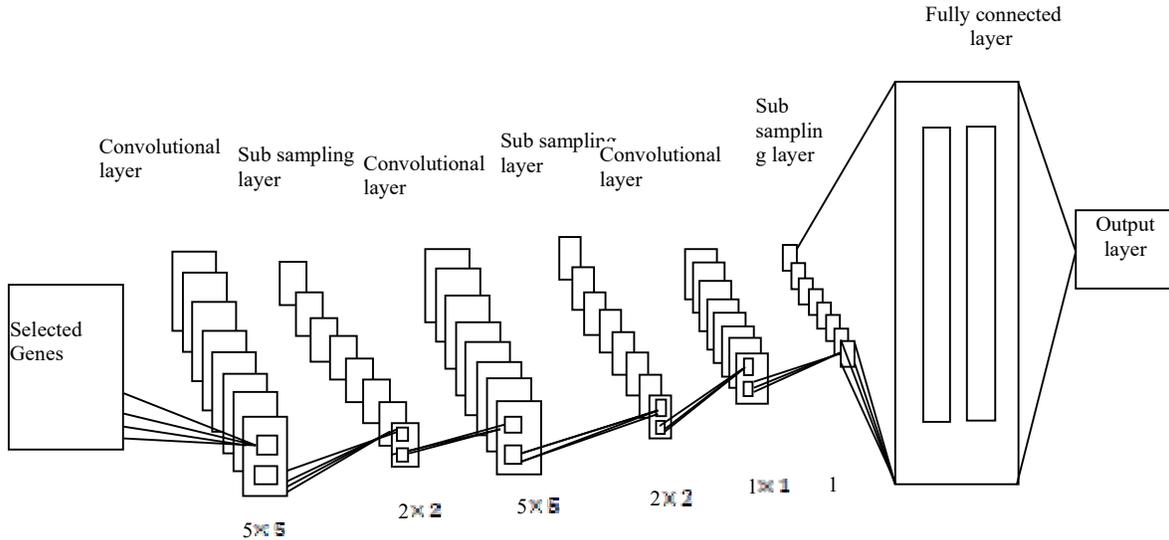


Figure.2: CNN framework

Modified LSTM

Compared to conventional RNNs, it is necessary to more accurately represent temporal sequences and their long-range dependencies. A unique type of RNN architecture known as the LSTM was created. As seen in Figure 2.2, an Input Gate (IG), Forget Gate (FG), and Output Gate (OG) as well as a cell activation component comprise a typical LSTM cell. The integrated multipliers in these components regulate the cell activation, which receive activation signals from various sources [20].

In LSTM, IG is expressed as

$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i) \quad (18)$$

The FG is formulated as

$$f_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}c_{t-1} + b_f) \quad (19)$$

The following eqn 20 represents the cell gate

$$c_t = f_t c_{t-1} + i_t \tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c) \quad (20)$$

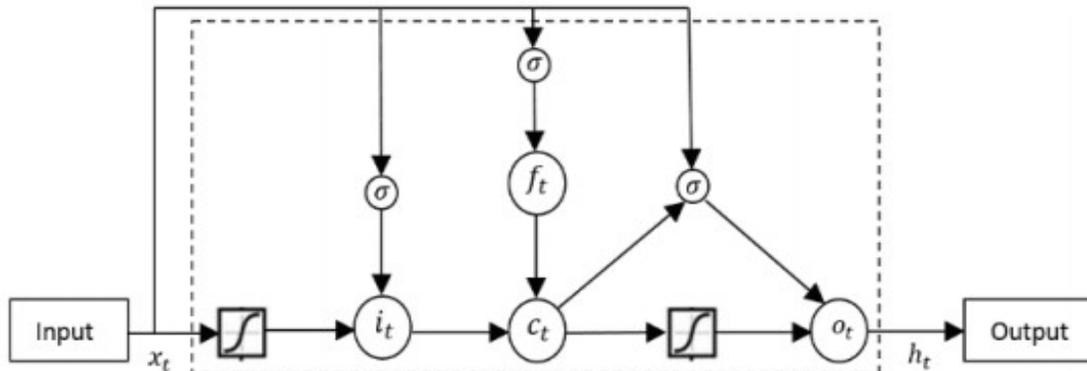


Figure.3: LSTM Cell

The OG is expressed below

$$o_t = \sigma(W_{x0}x_t + W_{h0}h_{t-1} + W_{c0}c_t + b_0) \quad (21)$$

At last, the concealed condition is calculated

$$h_t = o_t \tanh(c_t) \quad (22)$$

Here, hyperbolic tangent activation function can be represented as tanh.

input at time t, represented by x_t , W and b denote the weights and biases of the network.

Modified LSTM

For several time steps, the remainder of the network from changing the data in the memory cells can be prevented by the LSTM gates. LSTM networks maintain signals and transmit mistakes for far longer as compared to standard RNN.

By introducing a weighted approach based on weighted regression, which weights each feature according to its importance to the estimation point, the suggested work leverages Modified LSTM to circumvent these problems. A 3-D distance metric among the observation point and the estimation point can be used to establish the weight assigned to every feature.

The weight function determined has the following form.

$$\omega = \left\{ \left(1 - \left(\frac{d}{h}\right)^3\right)^3 \rightarrow \text{if } |d| \leq h \quad 0 \rightarrow \text{if } |d| > h \right\} \quad (23)$$

Here,

The weight was indicated by ω ,

The distance among the observation point and the estimation point is denoted as d

The half window width can be denoted as h.

Thus, the input gate can be denoted as i, forget gate can be denoted as f, output gate can be denoted as o, and cell state can be denoted as c. The logistic (SF) Sigmoid Function is represented by the symbol σ . For peephole connections, the weight matrices W_{ci} , W_{cf} , and W_{co} are indicated. 3 gates (i, f, and o) regulate the data flow in an LSTM. The input ratio is determined by the IG. This ratio affects equation (20) in the cell state calculation. The forget gate determines which information must

be stored in memory and which must be forgotten, passing the previous memory h_{t-1} or not. Calculated in equation (19), the ratio of the prior memory is utilized in equation (20). Passing or failing the memory cell's output is determined by the output gate.

This mechanism can be seen in equation (22). Because of the 3 gates, LSTM can be used for addressing vanishing and exploding gradient problems. The LSTM cell takes the place of the recurrent (HL) Hidden Layer in the LSTM-RNN structure. The classification of lung cancer is provided by the OG.

EL using MV

By adding more capabilities to classifiers run n times in a separate and independent manner, the MV approach gets DM from them. Let C be a set of Q classes and χ be a set of N samples. Using the M classifiers utilized for voting, lets construct an algorithm set $S = \{A1, A2, AM\}$. One of the Q classes is allocated to each case $x \in \chi$. Every instance will have predictions for every time classifier. For every instance, the MV class from classifiers is the class that is finally allocated to it. In MV, the prediction accuracy value (also known as Acc) of the classifier is used to weight each vote.

Next, define the total number of votes for a class c_k as follows:

$$T_k = \sum_{l=1}^M Acc(A_l) \times F_k(c_l) \quad (24)$$

$$F_k(c_l) = \begin{cases} 1 & c_l = c_k \\ 0 & c_l \neq c_k \end{cases} \quad (25)$$

Here, the C classes are represented by c_l and c_k . The class with the highest cumulative weight is selected. In general, to achieve the highest classification rate for classifying the data as positive or negative, each classifier is trained on a distinct set of independent training sets and weights are assigned.

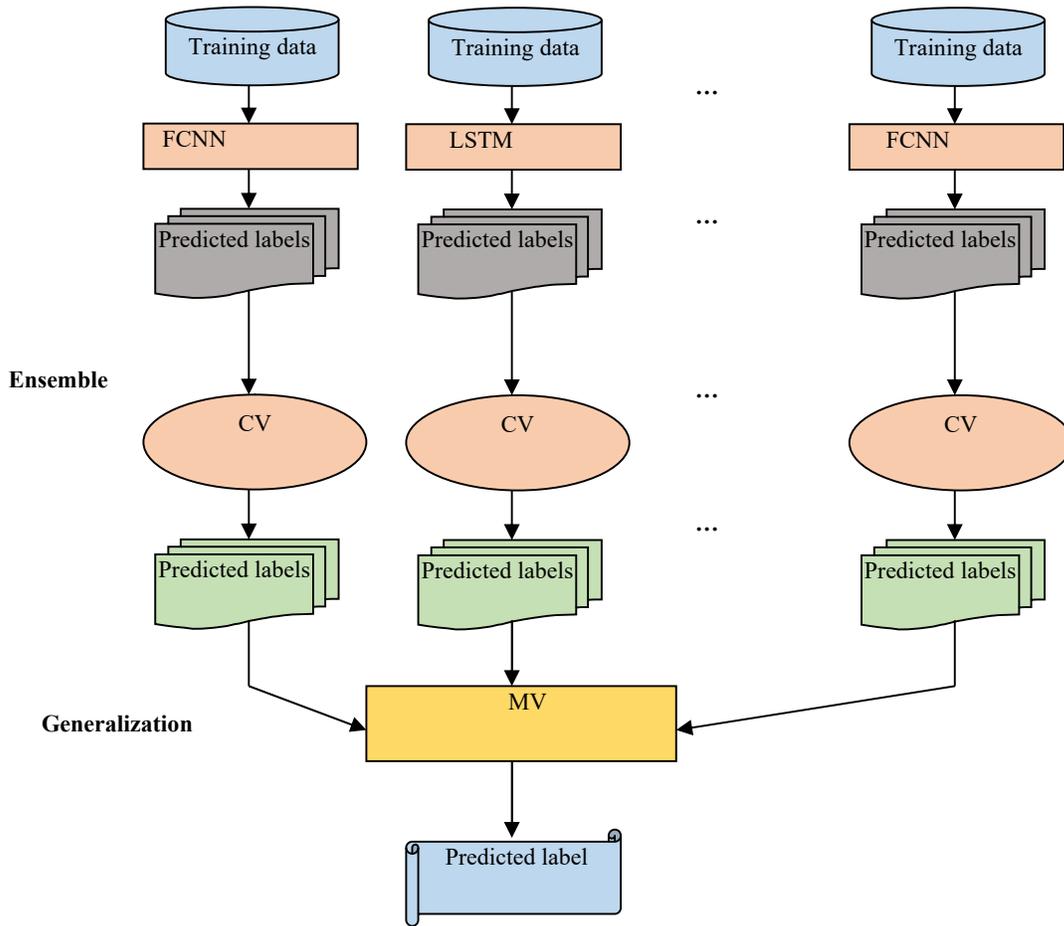


Figure: 4. EL method

4. EXPERIMENTAL RESULTS

The recommended EFCLS approach is contrasted to the already-variable wrappers + MLP, SAFSA+FCNN, wrappers + SVM -RBF, Wrapper +KNN, SAFSA+RF, SAFSA +FKNN, SAFSA+KSVM, wrappers+RF, EnsembleCL, and EFCLS techniques. The experiments conducted on the suggested model are analyzed in this section. The application of MATLAB facilitates the execution of this concept. In terms of Precision (P), recall (R), accuracy, F-measure, and error rate (ER), the above comparison can be made.

Training sets and Testing sets: Two sets of feature vectors are maintained: C₁ for disease-affected individuals and C₂ for healthy individuals. These sets contain feature vector from both sick and healthy individuals. Ten components comprise C₁ and C₂,

$$C_1 = \{c_{1,1}, c_{1,2}, \dots, c_{1,10}\} \text{ and}$$

$$C_2 = \{c_{2,1}, c_{2,2}, \dots, c_{2,10}\}. \text{ Then, the fragments } c_{1,fi} \text{ (from } C_1) \text{ and } c_{2,fi} \text{ (from } C_2) \text{ combine}$$

to form c_i . After these random mixings, ten pieces, or folds, $\{c_1, c_2, \dots, c_{10}\}$ were produced, with cases from sick and healthy patients included in each fold. The set is divided into ten folds as an outcome where nine folds are used in training and the balance for testing.

Then repeat the 10-fold process (10-fold (CV) Cross-Validation) by choosing one-fold for testing and others for training. Ten performance measure estimates,

performance1; ...; performance10 are generated by the procedure. The overall estimate is obtained by averaging these values.

Important features are extracted from speech signal spectrograms using a variety of signal processing techniques from 754 features provided to Wrappers selection processes in development phases. Various feature groups connect with different signal processing approaches (vocal fold, WT, baseline, time-frequency, MFCCs, and TQWT characteristics). The features that belong to each

feature group are included in Table 1 along with a description of each feature group.

Table 1 – Features derived from SS spectrograms

List of features	Description	Over-all features
Baseline	There are variations in shimmer and jitter. The vocal fold vibration frequencies were computed using the mean, median, SD, min, and max values. examination of detrended fluctuations, Characteristics of harmonicity Density entropy for the pitch period and the repetition period	32
Time-frequency	The first four formant frequency bandwidths are the speech signal strength in dB at its maximum, minimum, and means.	11
MFCCs	The mean, SD, and first and second derivatives of the initial 13 MFCCs. The logenergy of the signal.	84
Wavelet transform	DWT are extracted at level 10 using computations of Shannon and log energy entropies. The Teager-Kaiser energies and coefficients of approximations also computed.	182
VF	The patterns of movement of Glottis are regular. noise-induced activation of the glottis. The sound pressure produced by the pathologically vibrating VF. Utilizing Empirical Methods of Decomposition	22
TQWT	The TQWT approach makes use of the Q (Q-factor from 1 to 10), r (redundancy selected as 3, 4, 5), and J (number of levels between 5 and 50) factors. Following TQWT decompositions, energy/entropy scores for each level are obtained.	423

Table.2: Features with descriptions of ASD

Attributes	Types	Descriptions
Ages	Number	In years
Genders	String	Males or Females
Ethnicities	String	Records of ethnicities as text
Jaundice from Birth	Boolean (yes or no)	Cases with jaundice at birth
Family members with PDD	Boolean (yes or no)	Family member history of PDD
Test completed by	String	Parents, self, care takers, medical staff, clinicians, etc.
Residing Country	String	Countries listed as text
Prior usage of screening app	Boolean (yes or no)	Usage of screening apps by user
Screening Method Types	Integer (0,1,2,3)	Determined Screening techniques based on ages (0 for toddlers, 1 for children, 2 for adolescents, and 3 for adults).
Screening Scores	Integer	The scoring system used in the screening process was used to determine the final score. This was calculated automatically.

Table 3 – Performance metrics for PD-classification over four distinct classification frameworks utilizing a feature set chosen using Ensemble Classification across several classification frameworks

Metrics	Methods									
	Wrappers using KNN	SAFS A using FKN N	Wrappers using MLP	SAFS A using FCN N	Wrappers using SVM-RBF	SAFS A using KSV M	Wrappers using RF	SAFS A using RF	Ensemble CL	EFCLS
A	0.8558	0.8612	0.8664	0.8712	0.947	0.9621	0.922	0.930	0.97	0.99
Sensitivity	0.9379	0.9399	0.9362	0.9400	0.9645	0.9699	0.984	0.990	0.984	0.99
Specificity	0.6146	0.6150	0.6615	0.6655	0.9268	0.9281	0.7396	0.7421	0.941	0.96
P	0.8773	0.8801	0.8904	0.8999	0.9722	0.9790	0.9174	0.9200	0.981	0.99
FAR	0.1227	0.1210	0.1096	0.1011	0.0277	0.0222	0.0826	0.0700	0.021	0.01
MCC	0.5986	0.6010	0.6326	0.6330	0.8686	0.8690	0.7878	0.7910	0.881	0.89
F₁ score	0.9066	0.9099	0.9127	0.9187	0.9633	0.9721	0.9495	0.9501	0.97	0.98

Table 4 – ASD-classification performance metrics for 4 varied classification frameworks via a feature set selected via Ensemble Classification using numerous classification frameworks

Metrics	Approaches									
	Wrappers using KNN	SAFS A using FKN N	Wrappers using MLP	SAFS A using FCNN	Wrappers using SVM-RBF	SAFS A using KSV M	Wrappers using RF	SAFS A using RF	Ensemble CL	EFCLS
A	0.879	0.909	0.9162	0.92	0.935	0.9499	0.948	0.954	0.955	0.97
Sensitivity	0.909	0.919	0.9262	0.93	0.9445	0.9599	0.964	0.978	0.98	0.99
Specificity	0.646	0.65	0.6615	0.6755	0.9068	0.911	0.7596	0.7821	0.931	0.95
P	0.8573	0.860	0.87	0.88	0.9522	0.96	0.90	0.91	0.97	0.98
FAR	0.1327	0.121	0.1085	0.1071	0.0287	0.0232	0.0886	0.07	0.022	0.01
MCC	0.5786	0.581	0.6126	0.623	0.846	0.85	0.7278	0.751	0.901	0.95
F₁ score	0.8966	0.901	0.927	0.0387	0.9433	0.951	0.935	0.941	0.96	0.98

Tables 3 and 4 present a comparison between the proposed EFCLS approach and the existing strategies for the identification of PD and ASD, including SAFSA, wrapper KNN, wrapper MLP, wrapper SVM, RF, EnsembleCL, FKNN, FCNN, KSVM, and RF. A list of the outcomes is provided in the table below. The performance of the proposed EFCLS approaches is found to be superior than that of the existing classifiers. The current methods accuracy values are 0.8612 for FKNN, 0.8712 for FCNN, 0.9621 for KSVM, 0.930 for the RF and 0.8558 for KNN, 0.8664 for MLP, 0.947 for SVM, and 0.922 for RF strategy, and 0.95 for ensemble CL. In contrast, the accuracy values of the suggested PD EFCLS Classify are 0.99. Comparatively, the accuracy of the current approaches is lower than that of the suggested ASD EFCLS Classify, which has an accuracy value of 0.97.

When compared to the current wrappers-based classifiers, figure 6 shows that the suggested EFCLS classification approach performs better in terms of sensitivity.

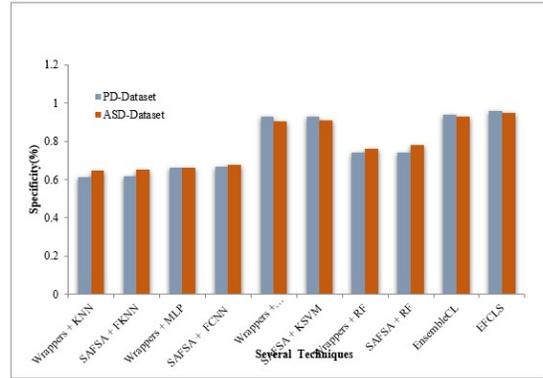


Figure 7: Specificity Comparison

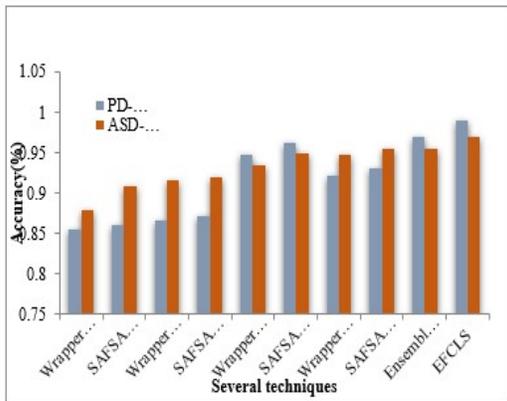


Figure .5: Comparison of accuracy

When compared to the current wrapper-based classifiers, figure 5 demonstrates the manner in which the suggested EFCLS technique performs better in terms of accuracy.

When compared to other classifiers used as a comparison, figure 7 indicate that the suggested EFCLS method executes well in terms of specificity.

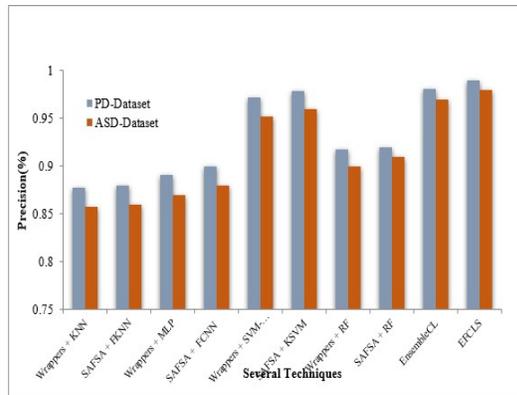


Figure .8: Precision Comparison

The suggested EFCLS method is compared with the current classifiers, such as SAFSA, wrapper KNN, wrapper MLP, wrapper SVM, RF, EnsembleCL, FKNN, FCNN, KSVM, and RF. Figure 8 shows that the suggested EFCLS techniques perform well in terms of precision.

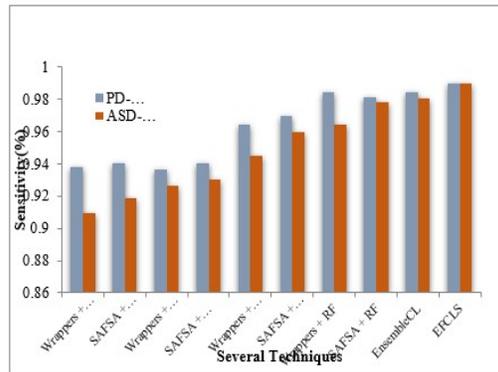


Figure .6: Comparison of Sensitivity

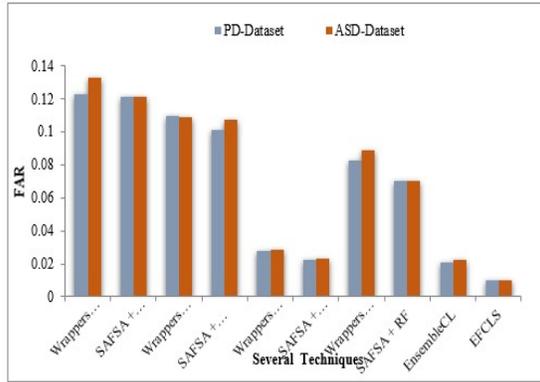


Figure .9: FAR Comparison

The suggested EFCLS method is compared with the current classifiers, such as SAFSA, wrapper KNN, wrapper MLP, wrapper SVM, RF, EnsembleCL, FKNN, FCNN, KSVM, and RF. Figure 9 illustrates that the suggested EFCLS techniques perform better in terms of FAR.

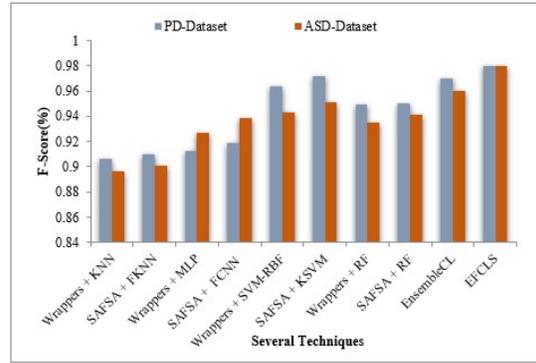


Figure .11: F1 Score Comparison

The suggested EFCLS method is compared with the current classifiers, such as SAFSA, wrapper KNN, wrapper MLP, wrapper SVM, RF, EnsembleCL, FKNN, FCNN, KSVM, and RF. Figure.11 shows that the suggested EFCLS approaches perform very well in terms of F1 score.

5. CONCLUSION AND FUTURE WORK

Reducing the rate of progression and giving patients access to disease-modifying treatments is certainly made possible by early and accurate detection of PD and ASD.

In this work suggested an improved model for PD and ASD diagnosis. In which first missing values are imputed using MCI. And data normalization will be done by z score normalization. DR will be performed based on BCDPC A. Significant features are selected from the dimension reduced data using ensemble of ECSO and IWO. Finally, PD and ASD are detected using ensemble of FCNN and LSTM network. Simulation outcomes shows that this suggested model provides better accuracy results than other existing models. It could be worked in the future. But, considering this proposed methodology is not currently employed for diagnosing other diseases.

REFERENCES:

[1] Solana-Lavalle, G., Galán-Hernández, J.C. and Rosas-Romero, R., “Automatic Parkinson disease detection at early stages as a pre-diagnosis tool by using classifiers and a small set of vocal features”, *Biocybernetics and Biomedical Engineering*, Vol.40, No.1, 2020, pp.505-516.

[2] Bazgir, O., Frounchi, J., Habibi, S.A.H., Palma, L. and Pierleoni, P., “A neural network system

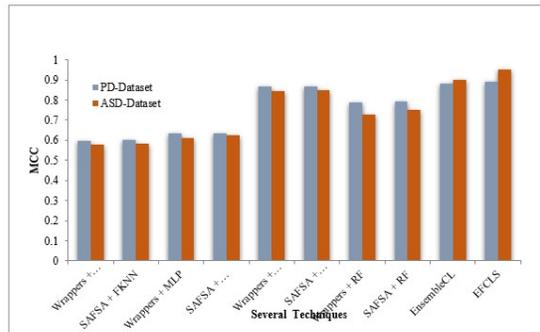


Figure .10: MCC Comparison

The suggested EFCLS method is compared with the current classifiers, such as SAFSA, wrapper KNN, wrapper MLP, wrapper SVM, RF, EnsembleCL, FKNN, FCNN, KSVM, and RF. Figure 10 shows that the suggested EFCLS methodology performs better in terms of MCC.

- for diagnosis and assessment of tremor in Parkinson disease patients”, In 2015 *22nd Iranian Conference on Biomedical Engineering (ICBME)*, 2015, pp. 1-5.
- [3] Upadhyya, S.S., Cheeran, A.N. and Nirmal, J.H., “Thomson Multitaper MFCC and PLP voice features for early detection of Parkinson disease”, *Biomedical Signal Processing and Control*, Vol.46, 2018 pp.293-301.
- [4] Sun, K., Li, L., Li, L., He, N. and Zhu, J., 2020, May. Spatial attentional bilinear 3d convolutional network for video-based autism spectrum disorder detection. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (pp. 3387-3391). IEEE.
- [5] Thapaliya, S., Jayarathna, S. and Jaime, M., “Evaluating the EEG and eye movements for autism spectrum disorder”, In 2018 *IEEE international conference on big data (Big Data)*, 2018, pp. 2328-2336.
- [6] Sharma, P., Sundaram, S., Sharma, M., Sharma, A. and Gupta, D., “Diagnosis of Parkinson’s disease using modified grey wolf optimization”, *Cognitive Systems Research*, Vol.54, 2019, pp.100-115.
- [7] Tomar, D., Prasad, B.R. and Agarwal, S., “An efficient Parkinson disease diagnosis system based on least squares twin support vector machine and particle swarm optimization”, In 2014 *9th International Conference on Industrial and Information Systems (ICIIS)*, 2014, pp. 1-6.
- [8] Zheng, W., Eilamstock, T., Wu, T., Spagna, A., Chen, C., Hu, B. and Fan, J., “Multi-feature-based network revealing the structural abnormalities in autism spectrum disorder”, *IEEE Transactions on Affective Computing*, 2019.
- [9] Adjorlu, A., Høeg, E.R., Mangano, L. and Serafin, S., “Daily living skills training in virtual reality to help children with autism spectrum disorder in a real shopping scenario”, In 2017 *IEEE International Symposium on Mixed and Augmented Reality (ISMAR-Adjunct)*, 2017, pp. 294-30.
- [10] Karan, B., Mahto, K. and Sahu, S.S., “Detection of Parkinson disease using variational mode decomposition of speech signal”, In 2018 *International Conference on Communication and Signal Processing (ICCSP)*, 2018, pp. 0508-0512.
- [11] Abayomi-Alli, O.O., Damaševičius, R., Maskeliūnas, R. and Abayomi-Alli, A., “BiLSTM with Data Augmentation using Interpolation Methods to Improve Early Detection of Parkinson Disease”, In 2020 *15th Conference on Computer Science and Information Systems (FedCSIS)*, 2020, pp. 371-380.
- [12] Li, Y., Zhang, C., Jia, Y., Wang, P., Zhang, X. and Xie, T., “Simultaneous learning of speech feature and segment for classification of Parkinson disease”, In 2017 *IEEE 19th International Conference on e-Health Networking, Applications and Services (Healthcom)*, 2017, pp. 1-6.
- [13] Ranjan, A. and Swetapadma, A., “An Intelligent Computing Based Approach for Parkinson Disease Detection”, In 2018 *Second International Conference on Advances in Electronics, Computers and Communications (ICAEECC)*, 2018, pp. 1-3.
- [14] Akter, T., Satu, M.S., Khan, M.I., Ali, M.H., Uddin, S., Lio, P., Quinn, J.M. and Moni, M.A., “Machine learning-based models for early stage detection of autism spectrum disorders”, *IEEE Access*, Vol.7, 2019, pp.166509-166527.
- [15] Kushki, A., Khan, A., Brian, J. and Anagnostou, E., “A Kalman filtering framework for physiological detection of anxiety-related arousal in children with autism spectrum disorder”, *IEEE Transactions on Biomedical Engineering*, Vol.62, No.3, 2014, pp.990-1000.
- [16] Skafidas, E., Testa, R., Zantomio, D., Chana, G., Everall, I.P. and Pantelis, C., “Predicting the diagnosis of autism spectrum disorder using gene pathway analysis”, *Molecular psychiatry*, Vol.19, No.4, 2014, pp.504-510.
- [17] Ingre, B. and Yadav, A., “Performance analysis of NSL-KDD dataset using ANN”, In 2015 *international conference on signal processing and communication engineering systems*, 2015, pp. 92-96.
- [18] Wu, D., Xu, S. and Kong, F., “Convergence analysis and improvement of the chicken swarm optimization algorithm”, *IEEE Access*, Vol. 4, 2016, pp.9400-9412.
- [19] Kumawat, I.R., Nanda, S.J. and Maddila, R.K., “Multi-objective whale optimization”, In *Tencon 2017-2017 ieee region 10 conference*, 2017, pp. 2747-2752.
- [20] Dai, S., Li, L. and Li, Z., “Modeling vehicle interactions via modified LSTM models for trajectory prediction”, *IEEE Access*, Vol.7, 2019, pp.38287-38296.