Int. J. Appl. Math. Comput. Sci., 2024, Vol. 34, No. 4, 647–660
DOI: 10.61822/amcs-2024-0043

A NOVEL CHAOTIC BINARY BUTTERFLY OPTIMIZATION ALGORITHM BASED FEATURE SELECTION MODEL FOR CLASSIFICATION OF AUTISM SPECTRUM DISORDER

ANANDKUMAR RAMAKRISHNAN [a], RAJAKUMAR RAMALINGAM [b], PADMANABAN RAMALINGAM [c], VINAYAKUMAR RAVI [d,*], TAHANI JASER ALAHMADI [e], SITI SARAH MAIDIN [f]

[a]Department of Information Technology
Sri Manakula Vinayagar Engineering College, 605107 Puducherry, India

[b]Centre for e-Automation Technologies
Vellore Institute of Technology, 600127 Chennai, Tamil Nadu, India

[c]School of Computer Science and Engineering
Vellore Institute of Technology, 600127 Chennai, Tamil Nadu, India

[d]Center for Artificial Intelligence
Prince Mohammad bin Fahd University, 34754 Khobar, Saudi Arabia
e-mail: vravi@pmu.edu.sa

[e]Department of Information Systems, College of Computer and Information Sciences
Princess Nourah bint Abdulrahman University, P.O. Box 84428, Riyadh, Saudi Arabia

[f]Faculty of Data Sciences and Information Technology
INTI International University, 71800 Nilai, Malaysia

Autism spectrum disorder (ASD) issues formidable challenges in early diagnosis and intervention, requiring efficient methods for identification and treatment. By utilizing machine learning, the risk of ASD can be accurately and promptly evaluated, thereby optimizing the analysis and expediting treatment access. However, accessing high dimensional data degrades the classifier performance. In this regard, feature selection is considered an important process that enhances the classifier results. In this paper, a chaotic binary butterfly optimization algorithm based feature selection and data classification (CBBOAFS-DC) technique is proposed. It involves, preprocessing and feature selection along with data classification. Besides, a binary variant of the chaotic BOA (CBOA) is presented to choose an optimal set of a features. In addition, the CBBOAFS-DC technique employs bacterial colony optimization with a stacked sparse auto-encoder (BCO-SSAE) model for data classification. This model makes use of the BCO algorithm to optimally adjust the 'weight' and 'bias' parameters of the SSAE model to improve classification accuracy. Experiments show that the proposed scheme offers better results than benchmarked methods.

Keywords: data classification, feature selection, metaheuristics, machine learning, autism spectrum disorder.

1. Introduction

Autism spectrum disorder (ASD) is a complex neurological disease that involves difficulties in social communication, repetitive behaviours, and a limited range of interests or activities (Duan *et al.*, 2022). Research indicates that the origins of this phenomenon include a complex interaction between genetic predispositions and environmental effects, such as maternal immunological stimulation during pregnancy. Although ASD is a disorder that lasts a lifetime, prompt intervention can

*Corresponding author

greatly improve the developmental quotients, cognitive abilities, and linguistic skills of individuals with ASD (Hirota and King, 2023; Gaspar *et al*., 2022). Recent research has emphasized the crucial need of identifying and addressing ASD at an early stage. These studies have shown that younger children with ASD have a higher potential for brain flexibility and the ability to adjust their behaviour. The progress in machine learning (ML) methods has created opportunities for automating the diagnosis of ASD by extracting behavioral or brain characteristics using various classifiers and feature extractors. This methodology not only simplifies the diagnostic procedure, but also provides more impartial and precise evaluations, thereby facilitating earlier detection and intervention for persons with ASD (Li *et al*., 2022a).

The prevalence and diverse nature of ASD have prompted many to favor ML over traditional statistical methodologies for data analysis. While standardized diagnostic tools are commonly used by clinicians, they often require considerable time for evaluation, presenting a drawback in the diagnostic process. ML offers an intelligent solution to this issue, with its primary aim being the reduction of diagnostic time while maintaining or improving accuracy levels. By expediting the diagnostic process, ML enables prompt intervention for individuals with ASD. Additionally, ML techniques aim to identify key ASD features by reducing the dimensionality of input datasets, further enhancing diagnostic efficiency and accuracy (Parlett-Pelleriti *et al*., 2023).

Feature selection (FS) plays a vital role in enhancing the accuracy and interpretability of predictive models for ASD (Mohammed *et al*., 2021). Given the complexity and heterogeneity of ASD, selecting the most relevant features from high-dimensional datasets is essential for constructing robust diagnostic or predictive models. Feature selection methods aim to identify a subset of features that are informative and discriminatory for distinguishing between individuals with ASD and neurotypical individuals. These methods can range from filter methods, which depend on statistical measures such as correlation or mutual evidence, to wrapper methods that utilize the predictive performance of a specific learning algorithm. By minimizing the data dimensionality while preserving the discriminatory power, feature selection not only improves the efficiency of machine learning models, but also aids in identifying biomarkers and underlying mechanisms associated with ASD, thus facilitating early diagnosis and personalized intervention strategies (Raj and Masood, 2020).

The FS method is generally modeled as an optimization process (Majidpour *et al*., 2024), with certain interest in those named wrapper methods. These techniques utilize classification efficiency as the fitness function (FF) and direct the procedure to select the subset of instances that maximize a few measures that consider the classification output. In previous years, some nature inspired meta-heuristic methods were assumed to tackle individual objective functions (Rahman *et al*., 2020). Maintaining a subgroup of related features and removing unrelated features could be useful in enhancing computation effectiveness and increasing classification accuracy (Alhafedh and Qasim, 2019). The wrapper module could be utilized for distinguishing related features and deleting unrelated features by accepting methods that choose many subsets of features and test them by utilising the recommended FF. With the improvement of computation intelligence, many attempts have been made to develop emerging algorithms for FS like the genetic algorithm (GA), gray wolf optimization (GWO), grasshopper algorithm, bat algorithm (BA) and particle swarm optimization (PSO).

The major objectives of this paper are as follows:

- Develop a chaotic binary butterfly optimization algorithm-based feature selection and data classification (CBBOAFS-DC) technique.
- Design a binary variant of the chaotic butterfly optimization algorithm (CBOA) for selecting an optimal set of features to enhance classification accuracy while minimising the number of chosen features.
- Implement the CBBOAFS-DC technique, incorporating the bacterial colony optimization with a stacked sparse autoencoder (BCO-SSAE) model for data classification.
- Utilize the BCO-SSAE algorithm to optimally adjust the ‘weight’ and ‘bias’ parameters of the SSAE model to enrich classification accuracy.
- Perform a number of simulations on common ASD datasets to see how well the proposed CBBOAFS-DC scheme works at classifying data.

The rest of the paper is structured as follows. Section 2 discusses various approaches involved in feature selection and their challenges. Section 3 outlines the proposed methodology and various architectures. Section 4 describes the experimentation and an analysis of the proposed model compared with other existing models. Finally, Section 5 concludes the paper with findings and future improvements.

2. Related work

This section contains a review of the recent standard methods associated with FS and classification processes. Omuya *et al*. (2021) designed a hybrid filter module for

FS depending on principal component analysis (PCA) along with information gain. The hybrid module is later employed for supporting classifiers by utilizing ML methods like the NB method. Thirumoorthy and Muneeswaran (2021) presented a new FS technique depending upon the hybrid binary poor and rich optimization (HBPRO) method for obtaining a suitable subset of optimum features. An optimum feature subset is chosen by NB classification with two standard text corpus datasets. Jiménez-Cordero *et al*. (2021) proposed an embedded FS technique depending on min and max optimization problems when a trade-off between the complexity module and the classification accuracy is required. With the leverage duality concept, they consistently recreate the min-max problems and resolve them without further ado by standard software for nonlinear optimization. Dong *et al*. (2020) presented several objective optimization-based multi-label FS (MMFS) methods. To improve the convergence and diversity of NSGA III, they proposed an enhanced NSGA III technique with two archives.

Qasim *et al*. (2020) chose a common discriminative feature using the novel chaotic binary black hole algorithm (CBBHA) while chaotic mappings include the motion of stars in BBHA. Research on 3 chemical datasets displays the presented method, CBBHA, which has the benefits of conventional BBHA based on related FS with higher classification efficiency. Li *et al*. (2022b) propounded meta-learning based industrial intelligence of FS architecture for classifying challenges that are widespread in implementation. The presented architecture contains three parts: algorithms, datasets, and classification recommendation modules. Paul *et al*. (2021) utilized multi-label learning and online appearance of features. This technique automatically defines the optimal subcategory of features that are appropriate for multi-label classifiers. A three-stage filtering procedure is employed for selecting suitable features. The initial stage is an evolution-based PSO method, which is employed for the set of received features in a multi-objective architecture. The next stage verifies the redundant FS in the present set with respect to the previously selected feature, and lastly, it discovers the features in the previously selected feature list that are turned into insignificant on choosing recently attained features and discards them.

Chen *et al*. (2020) employed three common datasets with a large number of parameters (Human Activity Recognition by Smartphones Car Evaluation Database and Bank Marketing). Moreover, they calculate and relate the performance and accuracy of the classification modules, like KNN, LDA, SVM, and RF. Spencer *et al*. (2020) investigated the efficiency of the ML method utilising related features selected via several FS techniques. Four commonly utilized heart disease datasets were processed by relief, PCA, chi-squared testing and symmetric uncertainty for creating separate feature sets. Therefore, different classification techniques were utilized for building modules, which are later related to seeking combinations of optimal features and obtaining precise predictions of heart condition. Liu *et al*. (2020) have proposed an FS technique for classifying text depending on independent feature space search. Initially, a relative document term frequency difference (RDTFD) technique is presented for dividing features in every text document into 2 independent sets based on feature capability for discriminating negative and positive instances that contain 2 significant functions: for improving higher-class correlation of features, decreasing correlation among features along with the range of search of feature space, and preserving suitable redundant features.

Loganathan *et al*. (2023) proposed chaotic Henry gas solubility optimization to select the features from an EEG microstates dataset. Better predictions are also made with the help of the bidirectional gated recurrent unit (Bi-GRU) and a weighted average ensemble approach. Uddin *et al*. (2023) introduced a machine learning approach to handle the ASD classification. In addition, the SMOTE method is used to handle the imbalanced dataset. They compared the model with numerous deep learning approaches. However, the results are not satisfactory and fail to determine the significant features.

3. Proposed model

The working process of the presented CBBOAFS-DC scheme is shown in Fig. 1. The figure portrays that the input is primarily preprocessed to eradicate the noise that occurs in it. In the CBBOAFS technique, chaos theory is incorporated into the BOA, and then a binary variant of the CBOA is developed for selecting an optimal collection of features. At last, the BCO algorithm with the SSAE based data classification model is executed to allocate class labels to input, where parameter tuning of the SSAE model is done using the BCO algorithm such that the classification performance is increased.

3.1. Data preprocessing. Initially, input data undergoe preprocessing in three different ways involving format transformation, replacing missing values and class labeling. Firstly, the original data in .arff format are transformed to a compatible .csv format. Secondly, the median technique is applied for the replacement of the missing values. Thirdly, the class labeling procedure takes place, which allocates the class labels to the original data instances.

3.2. Feature selection. Once input is pre-processed, in the ensuing stage, optimal features are chosen from preprocessed data using the CBBOAFS technique. Chaos theory and a binary version for FS are both included

Table 1. Analysis of various approaches for detecting ASD.

Reference	Methodology	PP	FS	PT	Classifier
Umamaheswari and Parthiban (2020)	IWOA-based feature selection Hybrid of PSO and WOA approach	✓	✓	×	FC
Nogay and Adeli (2023)	GSO for PT	✓	✓	✓	DCNN
Zhang *et al.* (2022)	Simplified VAE and MLP	✓	✓	×	MLP
Sahu and Verma (2022)	PSO for FS	✓	✓	×	MLP
Al-Muhanna *et al.* (2024)	AHHO for FS	✓	✓	×	ARTM
Almars *et al.* (2023)	GTO with TL for FS and PT	✓	✓	✓	TL
Mengash *et al.* (2023)	Owl search algorithm for FS BSAS with ID 3	✓	✓	✓	ID3

PP: preprocessing, FS: feature selection, PT: parameter tuning, ARTM: attention-based residual term memory, MLP: multi-layer perceptron, DCNN: deep convolution neural network, TL: transfer learning, ID3: iterative dichotomiser 3, FC: fuzzy classifier.

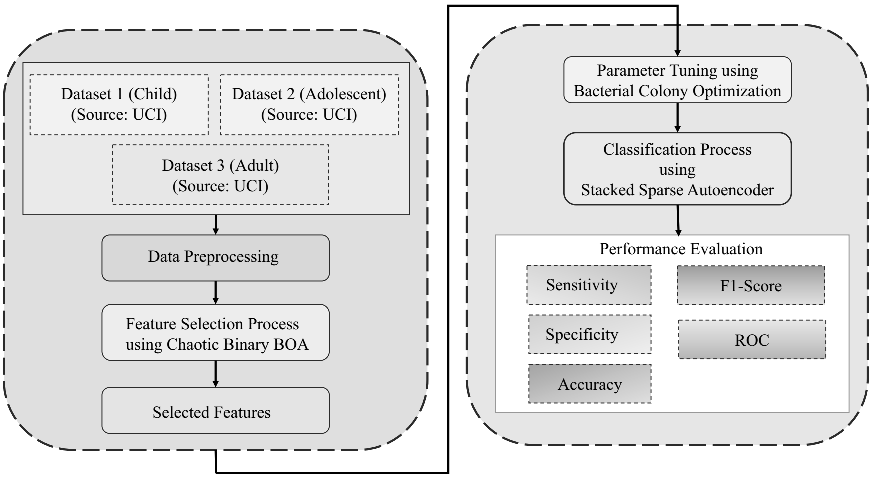

Fig. 1. Overall process of the CBBOAFS-DC model.

in the design of the CBBOAFS technique. Stimulated by speckled woods' mate finding approach, Qi *et al.* (2017) presented a novel metaheuristic technique named the BOA. Butterfly population is divided into two sets based on fitness. Butterflies with optimum fitness values form sunspot butterflies and others are named canopy butterflies. Besides, a distinct flight approach is employed for every group. The three-flight mode forms BOA methods like canopy, free, and sunspot flight modes. Several guidelines are created for idealizing mate finding approaches of butterfly in the BOA method, as given below:

- to raise the probability of facing female butterflies, every male flies closer to an optimum location called sunspot,
- to conquer an optimal sunspot, each sunspot butterfly frequently flies to a nearby sunspot, and
- each canopy butterfly frequently flies to other sunspot butterflies to contend the sunspot.

Let $P = \{p_1, p_2, \dots, p_m\}$ represent the butterfly population and $p_i \in \mathbb{R}^n$. The next approach is used for sunspot and canopy flight modes: every butterfly flies to arbitrarily chosen butterflies in accordance with the formula

$$p_{i,j}^{t+1} = p_{i,j}^t + (p_{i,j}^t - p_{k,j}^t)\beta. \tag{1}$$

Here i represents the i-th butterfly, j indicates a randomly chosen dimension from among $\{1, 2, \dots n\}$, t indicates present iteration, β represents an arbitrarily created amount from the interval $[1, -1]$ and k denotes a randomly chosen butterfly $(k \neq i)$.

Additionally, each butterfly uses the equation below to fly closer to a randomly chosen sunspot butterfly in the sunspot/canopy flight mode,

$$p_{i,j}^{t+1} = p_{i,j}^t + \frac{p_{k,j}^t - p_{i,j}^t}{x_{k,j}^t - p_{i,j}^t}(U - L)s\beta, \tag{2}$$

where U and L denote respectively the lower and upper bounds of the flying range of the i-th butterfly. The variable s linearly decreases from 1 to s_e using

$$s = 1 - (1 - s_e)\frac{t}{T} \tag{3}$$

where T represents the maximum number of iterations. In the case of the free-flight mode, each butterfly flies to a randomly novel location for enhancing the exploration stage in the BOA using

$$p_{i,j}^{t+1} = p_{i,j}^t + 2\alpha\beta - \alpha D, \tag{4}$$

where α gets linearly reduced from 2 to 0 in the sequence of iterations, and

$$D = |2\beta(p_{k,j}^t - p_{i,j}^t)|. \tag{5}$$

3.2.1. Design of the CBOA. To improve global optimization capabilities of the classical BOA, the CBOA is derived by including chaos theory. It is included in numerous optimization schemes for evading getting trapped in a local optimum and improving the solution quality. Every meta-heuristic scheme (including the BOA) depends on two classes: exploration and exploitation. During exploitation, the search is performed for obtaining an optimal solution, when exploration permits an effective search. Chaos is included in the BOA for striking a balance amid exploitation as well as exploration, therefore attaining an optimum solution effectively (Sayed *et al.*, 2019). In the BOA, variable θ is assumed as a major aspect influencing convergence actions. The efficiency of the BOA is based on its variables, and they noted that Eqn. (6) offers an essential momentum for stars by a probable search space and they noted the motion of stars to an optimum location, implying that the search space cannot be effectively examined.

Chaos is included for getting improved features during exploitation and exploration in each search space, therefore enhancing the efficacy of the optional technique in finding on optimal universal solution. A chaotic map (C_{map}) is used for finding the position of x_i^k, whereas θ is replaced with the attained value, as defined below

$$x_i^{k+1} = x_i^k + C_{\text{map}} \times (x_{\text{BH}} - x_i^k), \quad i = 1, 2, \ldots, N_\Delta, \tag{6}$$

where x_i^k and x_i^{k+1} represent the locations of the i-th star in iterations k and $k+1$, respectively, x_{BH} denotes the position of BH in space, C_{map} indicates the chaotic map, and N_s represents the number of stars. Around 10 calls to C_{map} are used for manipulating values of random variables in the BOA, and a primary value for each mapping is fixed as 0.7.

3.2.2. Binary version of the CBOA. The novel location of the butterfly produced by the local/global search would have a continuous solution, which needs to be converted to an equivalent binary value. This transformation is executed by employing a squash of continuous solutions at every dimension utilizing sigmoidal (S shaped) transfer function (Mirjalili and Hashim, 2012) that would force a butterfly to move into a binary search space,

$$S(F_i^k(t)) = \frac{1}{1 + e^{-F_i^k(t)}}. \tag{7}$$

Here F_i^k represents the continuous fragrance value of the i-th butterfly at k-th dimension in iteration t. The output from this function remains continuous and hence there must be a threshold for reaching binary values. The S shape function maps efficiently the infinite input to a finite output. It is beneficial to mention the likelihood of altering the values of the location vector in relation to the slope of the transfer function. A general stochastic threshold is employed for reaching a binary solution in the case of the sigmoidal function:

$$x_i^k(t+1)) = \begin{cases} 0 & \text{if rand} < S(F_i^k(t)), \\ 1 & \text{if rand} \geq S(F_i^k(t)). \end{cases} \tag{8}$$

3.2.3. Application of the CBBOA to feature selection. The FS is a binary optimization problem where the search agent is restricted to binary values of zero and one. In this study, all solutions are described as vectors the length of which is based on the number of attributes or features in the dataset. The vector entries may have two values, like zero/one, where a value of one represents the fact that the equivalent attribute or feature is selected and a value of zero indicates that the attribute or feature is not chosen. The FS problem is assumed by a multi-objective optimization problem where two differing objectives should be determined choosing the least number of features and highest classification accuracy. To resolve this problem, two binary optimization methods are presented (Arora and Anand, 2019). In the FS problem, the solution is assumed as an optimum that has fewer features along with maximum classification accuracy. All solutions are evaluated by the presented FF that is based on SSAE classification for calculating classification accuracy and the number of chosen features. Considering the aim to detect balance between the number of attributes and classification accuracy, the following fitness function (FF) is utilized for every optimization technique to calculate the solution:

$$\text{Fitness} = \alpha\gamma_R(D) + \beta\frac{|R|}{|N|}, \tag{9}$$

where $\gamma_R(D)$ denotes the error rate of SSAE classification. Additionally, $|R|$ denotes the cardinality of the selected feature subset and $|N|$ denotes the complete set of features from the actual dataset, α and β indicate two variables equivalent to the significance of classification quality along with the subset length, $\alpha \in [0, 1]$ and $\beta = 1 - \alpha$.

3.3. Data classification. During the data classification process, the BCO-SSAE model is employed for finding proper class labels. The SSAE model requires an optimal tuning of 'weight' and 'bias' parameters, which can be optimally accomplished using BCO algorithm.

3.3.1. SSAE architecture. Deep canonical correlation analyses (DCCA) could extract nonlinear data. But it neglects the significance of nonlinear dimension reduction. An autoencoder (AE) is recurrently utilized for nonlinear dimension reduction, particularly, an AE (Li *et al.*, 2020) encoding as well as decoding layers that are feedforward neural networks (FFNNs). Besides, an AE is mostly utilized for reducing data dimensionality. In the encoder layer, the AE gets $x \in \mathbb{R}^p$ as input, and the encoder x in hidden layer h deals with reducing input dimensions. In the decoder layer a reduced data length is interpreted as an outcome. The mathematical expression of the encoded input vector is

$$h = \sigma(Wx + b), \tag{10}$$

where σ denotes the activation function, e.g., a sigmoid or a tanh, $W \in \mathbb{R}^{n\times p}$ stands for the weight matrix, $b \in \mathbb{R}^n$ signifies the bias vector. The hidden expression is responsible for getting data closer to input x with the decoding expression

$$\widehat{x} = \sigma(W'h' + b'), \tag{11}$$

where $W' \in \mathbb{R}^{(p\times n)}$ represents the weight matrix and $b' \in \mathbb{R}^p$ means the bias vector. The difference between the input and output is called a reconstruction error. For optimizing the variables W, W', b and b', the reconstruction error is utilized as the cost function. For an individual training sample, the cost function is

$$J_{\text{AE}} = \frac{1}{2}\|\widehat{x} - x\|^2. \tag{12}$$

For several training samples (the number of training samples is N), the complete cost function is

$$J_{\text{AE}} = \frac{1}{N}\sum_{i=1}^{N}\|\widehat{x}(i) - X(i)\|^2 \tag{13}$$

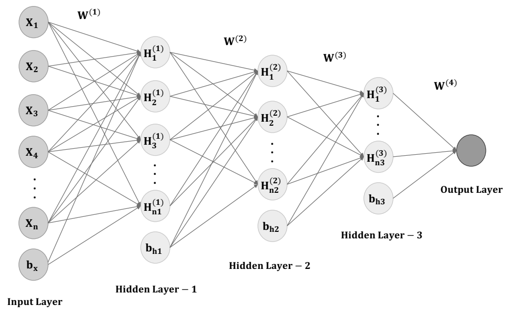

Fig. 2. Structure of the SSAE model.

Overfitting is a problem in training the autoencoder network. Therefore, a weight penalty is employed for the cost function that can effectively resolve too much overfit (Czmil *et al.*, 2024). The penalized cost function is

$$\begin{aligned} J_{\text{AE}} = {} & \frac{1}{N}\sum_{i=1}^{N}\frac{1}{2}\|\widehat{x}(i) - X(i)\|^2 \\ & + \frac{\gamma}{2}(\|W\|^2 + \|W'\|^2). \end{aligned} \tag{14}$$

If the input dimension and the number of hidden units are high, sparsity is enforced on the hidden unit in the training process for discovering an accurate input form. A neuron is active when its output value is close to 1, whereas inactive output is close to 0. The average activation is

$$\widehat{\rho}_j = \frac{1}{N}\sum_{i=1}^{N} h_j(X(i)). \tag{15}$$

For enforcing sparsity, it is imposed that $\widehat{\rho}_j = \rho$, while ρ represents a sparsity target (generally a small positive number is near zero). Therefore, the KL deviation between $\widehat{\rho}_j$ and ρ is minimized,

$$J_{\text{KL}}(\rho\|\widehat{\rho}_j) = \sum_{j=1}^{S}\rho\log\frac{\rho}{\widehat{\rho}_j} + (1-\rho)\log\frac{1-\rho}{1-\widehat{\rho}_j}, \tag{16}$$

where S denotes the number hidden layer nodes. Thus, the total cost function of a sparse AE is

$$\begin{aligned} J_{\text{SAE}} = {} & \frac{1}{N}\sum_{i=1}^{N}\frac{1}{2}\|\widehat{x}(i) - X(i)\|^2 \\ & + \frac{\gamma}{2}(\|W\|^2 + \|W'\|^2) + \beta J_{KL}(\rho\|\widehat{\rho}_j), \end{aligned} \tag{17}$$

where β represents the sparsity penalty. Minimizing the cost function, we could attain optimum values of variables W' and bias b'. The AEs are stacked altogether to learn useful features. The stacked sparse auto encoder (SSAE) is represented using many AEs. Figure 2 illustrates the structure of the SSAE model.

3.3.2. Parameter tuning using the BCO algorithm. BCO (Wang *et al.*, 2014) is a simple optimization procedure with several exchange topologies, better convergence, characterized by design simplicity. The chemotaxis procedure is a critical optimization phase in the BCO algorithm. It comprises two major approaches: tumbling and running strategies. Predominantly, the goal of running approach is to create a novel location ($\theta_i(T)$) with respect to the prior location ($\theta_i(T-1)$), a dynamic/random oriented study ($Pbest_i$), and a group-oriented study ($Gbest$). To improve convergence and diversity, the tumbling approach is introduced while the running approach becomes invalid at a particular level. During the tumbling procedure, additional arbitrariness term (tur_i) is utilized for seeking an optimal location for nutrients. The running formula has the form

$$\begin{aligned}\theta_i(T) &= \theta_i(T-1) + R_i \times (Gbest - \theta_i(T-1)) \\ &\quad + (1-R_i) \times (Pbest_i - \theta_i(T-1)).\end{aligned} \tag{18}$$

The tumbling formula is

$$\begin{aligned}\theta_i(T) &= \theta_i(T-1) + R_i \times (Gbest - \theta_i(T-1)) \\ &\quad + (1-R_i) \times (Pbest_i - \theta_i(T-1)) \\ &\quad + C(i) \times tur_i.\end{aligned} \tag{19}$$

Here, $Gbest$ and $Pbest$ denote the group oriented and dynamic oriented studies, respectively, C represents the step size of chemotaxis. A random variable $tur_i = \Delta(i)/\sqrt{(\Delta^T(i)\Delta(i))}$ is used, in which $\Delta(i)$ indicates the angle representing the direction of the i-th bacterium lying in $[-l, 1]$. The variables like R_1 and R_2 represent two randomly formed constants and their values range from 0 to 1. Figure 3 displays the workflow of the BCO technique.

At the final stage, the parameters involved in the SSAE model are optimally adjusted using the BCO algorithm, which enhances the classification performance to a maximum extent. The weight and bias parameters of the SSAE model are tuned using the BCO algorithm (Wang *et al.*, 2019; Helen Josephine *et al.*, 2023). The 10-fold cross-validation (CV) approach is used for assessing the FF. The training dataset is randomly separated as a set of 10 mutually exclusive subsets of approximately same sizes, where 9 of them are employed to trained the model and the last subset is employed in testing. This process iterates 10 times so that every subset is utilized for the tested model. The FF is $1 - \mathrm{CA}_{\text{validation}}$ of 10-fold CV from trained datasets,

$$\text{Fitness} = 1 - \mathrm{CA}_{\text{validation}}, \tag{20}$$

$$\mathrm{CA}_{\text{validation}} = 1 - \frac{1}{10}\sum_{i=1}^{10}\left|\frac{y_c}{y_c + y_f}\right| \times 100, \tag{21}$$

where y_c and y_f indicate the true and false classification counts, respectively. A solution with maximal $CA_{\text{validation}}$ yields a minimum fitness value.

Table 2. ASD dataset from UCI.

No.	Dataset	NoA	NoI
1	Children	21	292
2	Adolescent D	21	104
3	Adult	21	704

NoA: number of attributes,
NoI: number of instances.

Table 3. Attributes in the applied dataset.

No.	Description
1	Age of patient
2	Sex of patient
3	Society
4	Neonatal jaundice?
5	Does any relative have PDD?
6	Who fulfils the test?
7	Country of residence
8	Already used screening app before or not?
9	Type of screening test
10–19	Answer 10 qns. based on screening method
20	Screening score
21	Target class [Yes/No]

4. Experiments and result analysis

4.1. Experimental setup. The CBBOAFS-DC technique was applied to the ASD dataset to select significant features and, thereby, to improve the classification accuracy. A Windows PC equipped with an AMD Ryzen 7 2700x processor, 16 GB of RAM, a 500 GB SSD, and an NVIDIA GeForce GTX 1050 Ti graphics card was used for the experimentation. Furthermore, the Python language was employed for both code execution and outcome analysis.

4.2. Dataset description. The presented method is implemented using Python 3.6.5 and the outcomes are observed for three ASD datasets. The first ASD-Children dataset includes 292 instances, the ASD-Adolescent dataset has 104 instances, and the ASD-Adult dataset comprises 704 instances. The dataset details are given in Table 2. Besides, attribute details of the datasets are illustrated in Table 3.

4.3. Result analysis. Table 4 and Fig. 4 examine FS performance of the CBBOAFS-DC model with other present schemes. From the obtained results, it is apparent that the PSO-FS (Shami *et al.*, 2023) and GWO-FS (Banaie-Dezfouli *et al.*, 2023) algorithms yielded inferior FS performance with the least costs of 0.7891 and 0.6523, respectively. At the same time, the QODF-FS model (Zhao *et al.*, 2023) slightly improved the FS results with a moderate better cost of 0.3127.

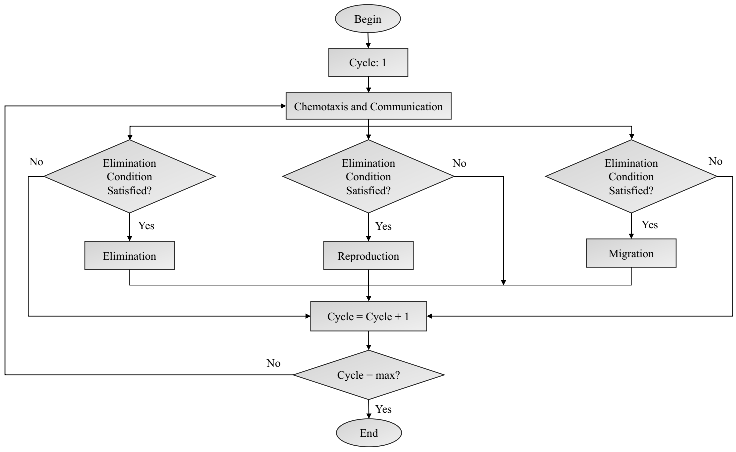

Fig. 3. Flowchart of the BCO algorithm.

Table 4. Attributes chosen by the proposed CBBOAFS-DC technique.

Methods	Best cost	Chosen features
CBBOAFS-DC	0.2983	1, 2, 4 − 7, 10, 12, 13, 16
QODF-FS	0.3127	1 − 4, 7, 9 − 11, 14, 15, 20
GWO-FS	0.6523	1, 4 − 9,11 − 17, 19
PSO-FS	0.7891	3 − 8,10 − 18

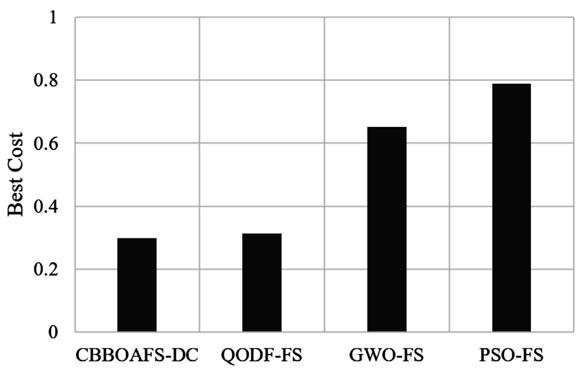

Fig. 4. Best cost investigation of the CBBOAFS-DC method.

Table 5. Result of the proposed CBBOAFS-DC model on applied datasets.

Measures	SE (%)	SP (%)	Acc (%)	F (%)
	ASD-Children dataset			
Run 1	98.58	98.68	98.63	98.58
Run 2	97.16	99.34	98.29	98.21
Run 3	97.87	98.01	97.95	97.87
Run 4	98.58	99.34	98.97	98.93
Run 5	97.87	98.68	98.29	98.22
	ASD-Adolescent dataset			
Run 1	100	97.56	99.04	99.21
Run 2	98.41	92.68	96.15	96.88
Run 3	100	95.12	98.08	98.44
Run 4	96.83	95.12	96.15	96.83
Run 5	98.41	97.56	98.08	98.41
	ASD-Adult dataset			
Run 1	97.88	99.03	98.72	97.63
Run 2	97.35	99.22	98.72	97.61
Run 3	98.94	99.03	99.01	98.16
Run 4	99.47	99.61	99.57	99.21
Run 5	97.88	99.81	99.29	98.67

SE: sensitivity, SP: specificity, Acc: accuracy, F: F-score.

However, the presented CBBOAFS-DC technique has offered an improved outcome with the worst cost of 0.2983. Therefore, the CBBOA technique has appeared as a better FS technique than other optimization algorithms.

The confusion matrix generated by the presented technique after distinct five runs for three different ASD datasets is illustrated in Fig. 5.

Table 5 showcases the classification outcomes obtained by the CBBOAFS-DC model on the applied dataset. Classification results of the proposed CBBOAFS-DC model on the ASD-Children dataset are demonstrated in Fig. 6. It is clear that the CBBOAFS-DC model produces improved results under distinct runs. For example, for Run 1, the CBBOAFS-DC model produces 98.58% sensitivity, 98.68% specificity, 98.63% accuracy and 98.58% F-score. In addition, for Run 3, the CBBOAFS-DC approach yields 97.87% sensitivity, 98.01% specificity, 97.95% accuracy and 97.87% F-score. Also, for Run 5, the CBBOAFS-DC model ends up with 97.87% sensitivity, 98.68% specificity, 98.29% accuracy and 98.22% F-score.

A classification results analysis of the proposed

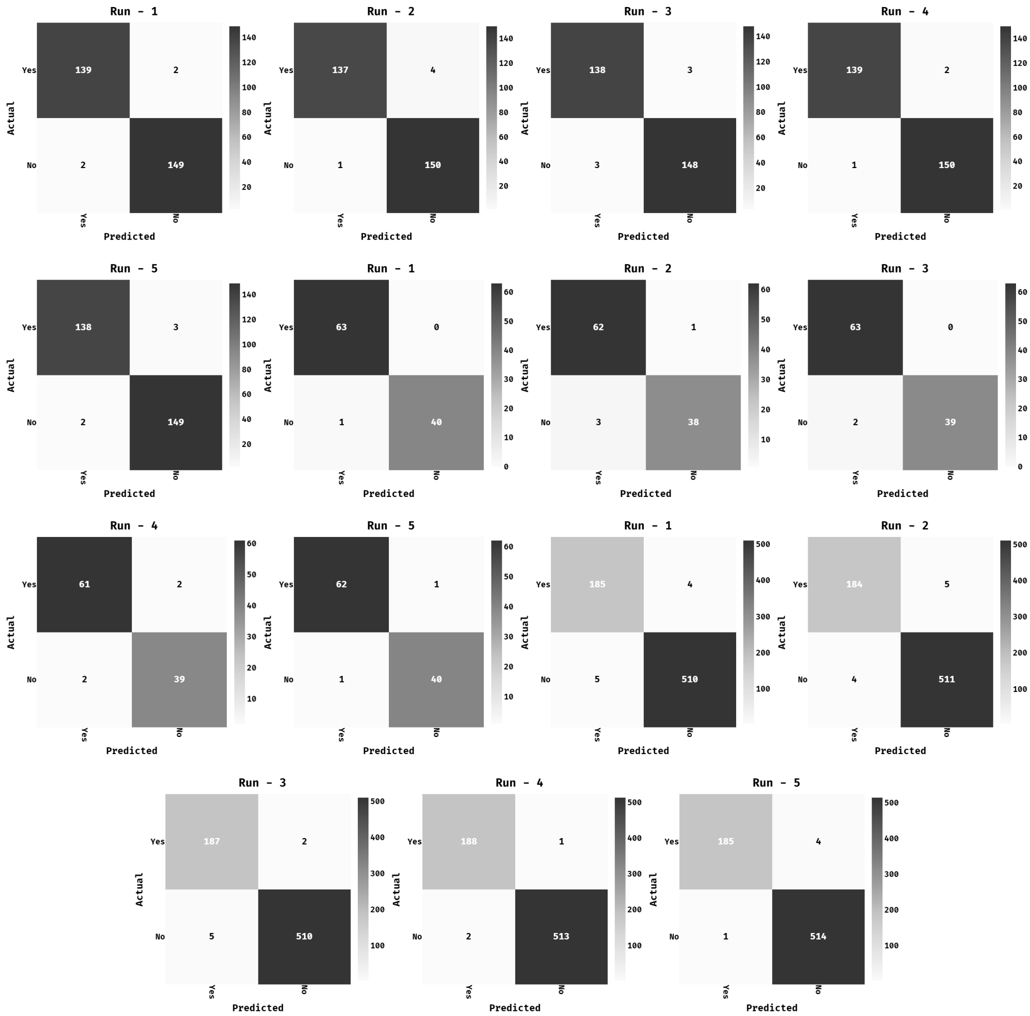

Fig. 5. Confusion matrix of different runs: ASD-Children dataset (Runs 1–5) (a), ASD-Adolescent dataset (Runs 1–5) (b), ASD-Adult dataset (Runs 1–5) (c).

CBBOAFS-DC technique on the applied ASD-Adolescent dataset is demonstrated in Fig. 7. It is obvious that the CBBOAFS-DC model exhibits improved results for distinct runs. For Run 1, the CBBOAFS-DC model yields 100% sensitivity, 97.56% specificity, 99.04% accuracy and 99.21% F-score. Besides, for Run 3, the CBBOAFS-DC technique produces 100% sensitivity, 95.12% specificity, 98.08% accuracy and 98.44% F-score. Additionally, for Run 5, the CBBOAFS-DC approach produces 98.41% sensitivity, 97.56% specificity, 98.08% accuracy and 98.41% F-score.

An analysis of the classification results for the proposed CBBOAFS-DC method on the ASD-Adult dataset is demonstrated in Fig. 8. It is seen that the CBBOAFS-DC model exhibits better results for varying runs. For Run 1, the CBBOAFS-DC model yields 97.88% sensitivity, 99.03% specificity, 98.72% accuracy and 97.63% F-score. Moreover, for Run 3, the CBBOAFS-DC method produces 98.94% sensitivity, 99.03% specificity, 99.01% accuracy and 98.16% F-score. Furthermore, for Run 5, the CBBOAFS-DC scheme offers 97.88% sensitivity, 99.81% specificity, 99.29% accuracy and 98.67% F-score.

Table 6 and Fig. 9 show the analysis of the average classification results of the CBBOAFS-DC model on the applied dataset. The CBBOAFS-DC method classified the ASD-Children dataset with average 98.01% sensitivity, 98.81% specificity, 98.43% accuracy and 98.36% F-score. Besides, the CBBOAFS-DC model classified the ASD-Adolescent dataset with 98.73% sensitivity, 95.61% of specificity, 97.50% accuracy and 97.95% F-score. At last, the CBBOAFS-DC methodology classified the ASD-Adult dataset with 98.30% sensitivity, 99.34% specificity, 99.06% accuracy and 98.26% F-score. The analysis of the ROC curve for different runs on the ASD-Children dataset, ASD-Adolescent dataset and

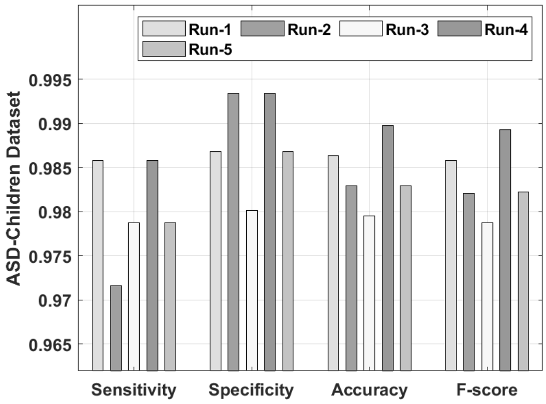

Fig. 6. Result analysis of the CBBOAFS-DC method on the ASD-Children dataset.

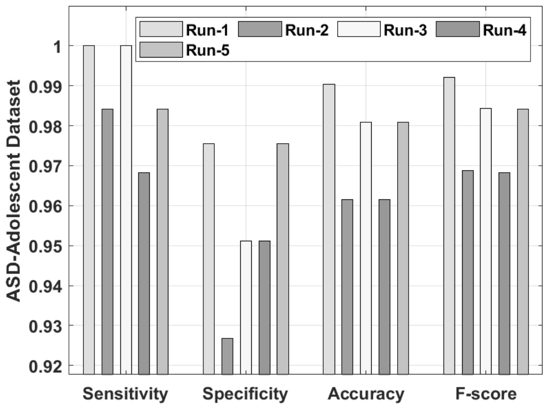

Fig. 7. Results for the CBBOAFS-DC method on the ASD-Adolescent dataset.

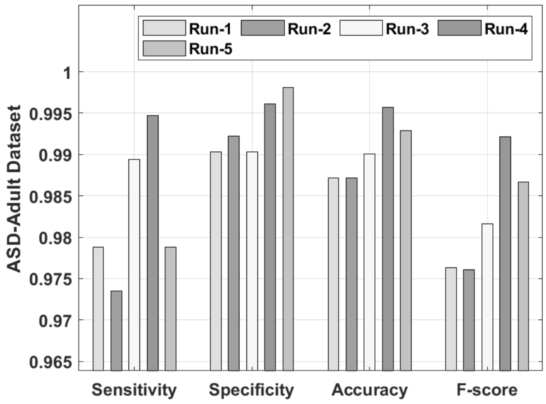

Fig. 8. Results for the CBBOAFS-DC method on the ASD-Adult dataset.

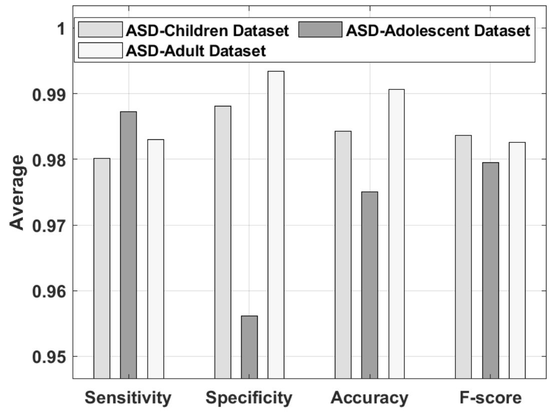

Fig. 9. Average performance of the CBBOAFS-DC model.

Table 6. Average performance of the proposed CBBOAFS-DC method on the dataset applied.

Measures	SE (%)	SP (%)	Acc (%)	F (%)
	ASD-Children dataset			
Average	98.01	98.81	98.43	98.36
	ASD-Adolescent dataset			
Average	98.73	95.61	97.5	97.95
	ASD-Adult dataset			
Average	98.3	99.34	99.06	98.26

ASD-Adult dataset is presented in Fig. 10.

A detailed comparative study of the classification results produced by the CBBOAFS-DC model with other existing techniques is illustrated in Table 7 and Figs. 11 and 12. On examining the results with respect to sensitivity and specificity, the KNN approach yielded poorest classification outcomes with 46.6% sensitivity and 72.1% specificity. Apart from that, the DT technique displayed slightly improved results with 53.3% sensitivity and 54.9% specificity. Then, the NN model put forward slightly increased performance with 53.3% sensitivity and 71.2% specificity. Besides, the LR model resulted in a moderate performance with 55% sensitivity and 62.6% specificity. Moreover, the QODF-DSAN model showcased reasonable results with 95.31% sensitivity and 98.83% specificity. Furthermore, the SSAE model generated nearly acceptable outcomes with 96.41% sensitivity and 97.92% specificity. Though the BCO-SSAE model brought about competitive outcomes with 97.43% sensitivity and 98.61% specificity, the presented CBBOAFS-DC technique outperformed the earlier methods with 98.3% sensitivity and 99.34% specificity.

Finally, on investigating the results based on accuracy and F-score, the DT model provided worst classification outcomes with 54.7% accuracy and 52.81% F-score. In the meantime, the LR model exhibited somewhat improved results offering 59.1% accuracy and 60.82% F-score. At the same time, the KNN technique lead to an even increased performance offering 61.8% accuracy and 68.7% F-score. The NN model offers a moderate performance with 62% accuracy and

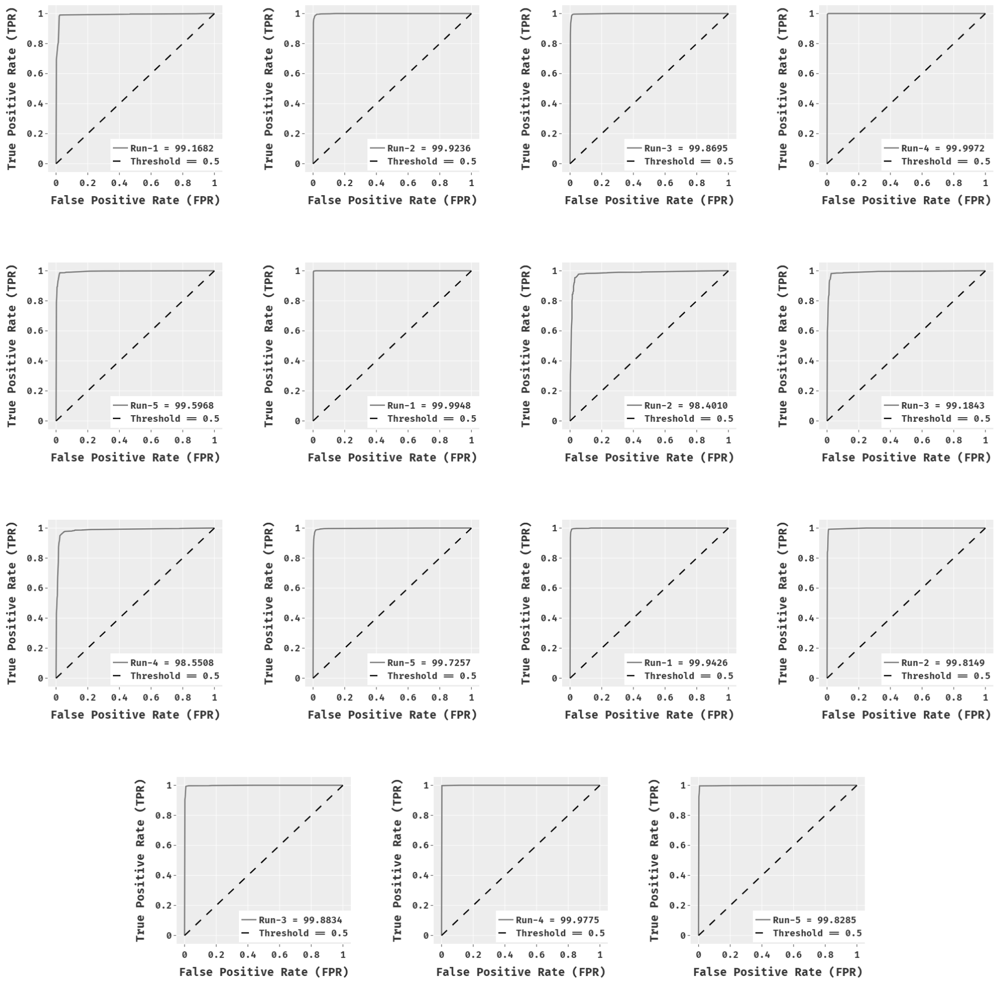

Fig. 10. ROC analysis of different runs: ASD-Children dataset (Runs 1–5) (a), ASD-Adolescent dataset (Runs 1–5) (b), ASD-Adult dataset (Runs 1–5) (c).

Table 7. Comparative analysis of the proposed CBBOAFS-DC model with existing methods.

Methods	SE (%)	SP (%)	Acc (%)	F (%)
CBBOAFS-DC	98.3	99.34	99.06	98.26
BCO-SSAE	97.43	98.61	98.61	97.95
SSAE	96.41	97.92	98.32	96.49
QODF-DSAN	95.31	98.83	97.87	96.06
Decision tree	53.3	54.9	54.7	52.81
LR	55.5	62.6	59.1	60.82
Neural network	53.3	71.2	62	69.88
k-NN	46.6	72.1	61.8	68.7

69.88% F-score. Concurrently, the QODF-DSAN model showcased somewhat reasonable outcomes with 97.87% accuracy and 96.06% F-score.

Along with that, the SSAE model resulted in nearly acceptable results with 98.32% accuracy and 96.49% F-score. Eventually, the BCO-SSAE model yielded a competitive outcome with 98.61% accuracy and 97.95% F-score, and the presented CBBOAFS-DC method beat up the earlier models with 99.06% accuracy and 98.26% F-score. After examining the above-mentioned tables and figures, it is seen that the CBBOAFS-DC method is considered an effectual data classification tool. Besides, it is also ensured that the inclusion of the CBBOA based FS technique helps to considerably boost the overall classification results.

4.4. Discussion. The CBBOAFS-DC method demonstrates superior classification performance on the ASD datasets across various age groups, achieving remarkable metrics with an average sensitivity, specificity, accuracy, and F-score of 98.01%, 98.81%, 98.43%, and 98.36%, respectively, for the ASD-Children dataset. For the ASD-Adolescent dataset, it attained 98.73%

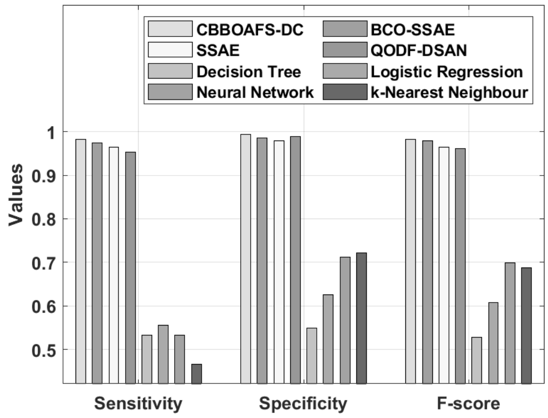

Fig. 11. Comparative analysis of the CBBOAFS-DC model with existing techniques.

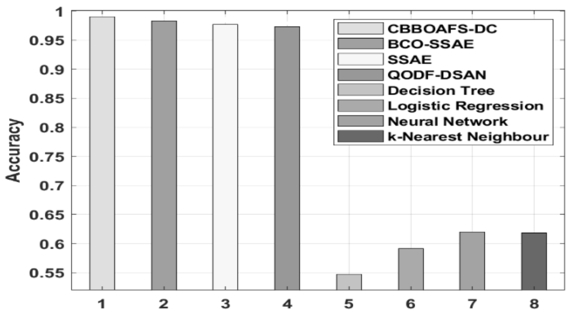

Fig. 12. Accuracy analysis of the CBBOAFS-DC model with existing techniques.

sensitivity, 95.61% specificity, 97.50% accuracy, and a 97.95% F-score. Furthermore, the method excelled with the ASD-Adult dataset, achieving 98.30% sensitivity, 99.34% specificity, 99.06% accuracy, and a 98.26% F-score. These results are comprehensively compared with existing techniques in Table 6 and Figs. 11 and 12. Notably, the KNN approach yielded the lowest classification outcomes, with sensitivity and specificity of 46.6% and 72.1%, respectively. The DT technique showed a slight improvement with 53.3% sensitivity and 54.9% specificity, while the NN model performed better with 53.3% sensitivity and 71.2% specificity. The LR model provided moderate results, achieving 55% sensitivity and 62.6% specificity.

The QODF-DSAN model produced reasonable results with 95.31% sensitivity and 98.83% specificity, and the SSAE model displayed nearly acceptable outcomes with 96.41% sensitivity and 97.92% specificity. Although the BCO-SSAE model exhibited competitive outcomes with 97.43% sensitivity and 98.61% specificity, the CBBOAFS-DC technique surpassed all with 98.3% sensitivity and 99.34% specificity. Additionally, in terms of accuracy and F-score, the DT model had the lowest performance with 54.7% accuracy and a 52.81% F-score. The LR model improved slightly with 59.1% accuracy and 60.82% F-score, while the KNN technique further enhanced performance with 61.8% accuracy and 68.7% F-score. The NN model provided moderate results with 62% accuracy and 69.88% F-score. The QODF-DSAN model again showed reasonable outcomes with 97.87% accuracy and a 96.06% F-score, underscoring the superior performance of the CBBOAFS-DC method in classifying ASD datasets.

5. Conclusion

This paper offers a novel CBBOAFS-DC scheme for an effective data classification process. The presented CBBOAFS-DC technique involves preprocessing, CBBOAFS based selection of features, and BCO-SSAE based data classification. The chaos concept is incorporated into the classical BOA to enhance global optimization abilities. Then, a binary variant of the CBOA technique is used for selecting an optimum subset of features from the preprocessed data. Next, the SSAE model is applied to the data classification process where weight and bias values of the SSAE model are optimally tuned using the BCO algorithm. The application of the BCO algorithm assists in enhancing the classification accuracy of the SSAE model.

For examining the superior data classification performance of the proposed CBBOAFS-DC technique, a sequence of simulations was performed on standard ASD datasets. The obtained investigational values show supremacy of the CBBOAFS-DC scheme over recent standard methods. The CBBOAFS-DC technique demonstrated superior performance compared with previous models, with an accuracy of 99.06% and an F-score of 98.26%.

The CBBOAFS-DC approach is well recognized as an efficient data categorization tool. Furthermore, the use of the CBBOA based FS approach significantly enhances the overall classification outcomes. As a future research direction, the classification performance can be boosted using data clustering techniques.

Acknowledgment

The authors extend their appreciation to Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia, for the Researchers Supporting Project no. PNURSP2024R513.

References

Al-Muhanna, M.K., Alghamdi, A.A., Alrfaei, B.M., Afzal, M., Al-Subaiee, R. and Haddadi, R. (2024). An attention-based hybrid optimized residual memory network (AHRML) method for autism spectrum disorder (ASD) detection, *Journal of Disability Research* **3**(3): 20240030.

Alhafedh, M.A.A. and Qasim, O.S. (2019). Two-stage gene selection in microarray dataset using fuzzy mutual information and binary particle swarm optimization, *Indian Journal of Forensic Medicine & Toxicology* **13**(4): 1162–1171.

Almars, A.M., Badawy, M. and Elhosseini, M.A. (2023). ASD^2-TL* GTO: Autism spectrum disorders detection via transfer learning with gorilla troops optimizer framework, *Heliyon* **9**(11): e21530.

Arora, S. and Anand, P. (2019). Binary butterfly optimization approaches for feature selection, *Expert Systems with Applications* **116**: 147–160.

Banaie-Dezfouli, M., Nadimi-Shahraki, M.H. and Beheshti, Z. (2023). BE-GWO: Binary extremum-based grey wolf optimizer for discrete optimization problems, *Applied Soft Computing* **146**: 110583.

Chen, R.-C., Dewi, C., Huang, S.-W. and Caraka, R.E. (2020). Selecting critical features for data classification based on machine learning methods, *Journal of Big Data* **7**(1): 52–78.

Czmil, S., Kluska, J. and Czmil, A. (2024). An empirical study of a simple incremental classifier based on vector quantization and adaptive resonance theory, *International Journal of Applied Mathematics and Computer Science* **34**(1): 149–165, DOI: 10.61822/amcs-2024-0011.

Dong, H., Sun, J., Sun, X. and Ding, R. (2020). A many-objective feature selection for multi-label classification, *Knowledge-Based Systems* **208**: 106456.

Duan, L., Liu, J., Yin, H., Wang, W., Liu, L., Shen, J. and Wang, Z. (2022). Dynamic changes in spatiotemporal transcriptome reveal maternal immune dysregulation of autism spectrum disorder, *Computers in Biology and Medicine* **151**: 106334.

Gaspar, A., Oliva, D., Hinojosa, S., Aranguren, I. and Zaldivar, D. (2022). An optimized kernel extreme learning machine for the classification of the autism spectrum disorder by using gaze tracking images, *Applied Soft Computing* **120**: 108654.

Helen Josephine, V., Vedaiyan, R., Xavier, V., Winston, J., Jegatheesan, A., Lakshmi, D. and Raj, J.S. (2023). Hybrid bacterial foraging optimization with sparse autoencoder for energy systems, *Computer Systems Science and Engineering* **45**(1): 701–714.

Hirota, T. and King, B.H. (2023). Autism spectrum disorder: A review, *Jama* **329**(2): 157–168.

Jiménez-Cordero, A., Morales, J.M. and Pineda, S. (2021). A novel embedded min-max approach for feature selection in nonlinear support vector machine classification, *European Journal of Operational Research* **293**(1): 24–35.

Li, G., Han, D., Wang, C., Hu, W., Calhoun, V.D. and Wang, Y.-P. (2020). Application of deep canonically correlated sparse autoencoder for the classification of schizophrenia, *Computer Methods and Programs in Biomedicine* **183**: 105073.

Li, J., Chen, Z., Li, G., Ouyang, G. and Li, X. (2022a). Automatic classification of ASD children using appearance-based features from videos, *Neurocomputing* **470**: 40–50.

Li, L., Wang, Y., Xu, Y. and Lin, K.-Y. (2022b). Meta-learning based industrial intelligence of feature nearest algorithm selection framework for classification problems, *Journal of Manufacturing Systems* **62**: 767–776.

Liu, Y., Ju, S., Wang, J. and Su, C. (2020). A new feature selection method for text classification based on independent feature space search, *Mathematical Problems in Engineering* **2020**(1): 6076272.

Loganathan, S., Geetha, C., Nazaren, A.R. and Fernandez, M.H.F. (2023). Autism spectrum disorder detection and classification using chaotic optimization based Bi-GRU network: An weighted average ensemble model, *Expert Systems with Applications* **230**: 120613.

Majidpour, J., Rashid, T.A., Thinakaran, R., Batumalay, M., Dewi, D.A., Hassan, B.A., Dadgar, H. and Arabi, H. (2024). NSGA-II-DL: Metaheuristic optimal feature selection with deep learning framework for HER2 classification in breast cancer, *IEEE Access* **2**(12): 38885–38898.

Mengash, H.A., Alqahtani, H., Maray, M., Nour, M.K., Marzouk, R., Al-Hagery, M.A., Mohsen, H. and Al Duhayyim, M. (2023). Automated autism spectral disorder classification using optimal machine learning model, *Computers, Materials and Continua* **74**(3): 5251–5265.

Mirjalili, S. and Hashim, S.Z.M. (2012). BMOA: Binary magnetic optimization algorithm, *International Journal of Machine Learning and Computing* **2**(3): 204–208.

Mohammed, M.B., Salsabil, L., Shahriar, M., Tanaaz, S.S. and Fahmin, A. (2021). Identification of autism spectrum disorder through feature selection-based machine learning, *2021 24th International Conference on Computer and Information Technology (ICCIT), Dhaka, Bangladesh*, pp. 1–6.

Nogay, H.S. and Adeli, H. (2023). Diagnostic of autism spectrum disorder based on structural brain MRI images using, grid search optimization, and convolutional neural networks, *Biomedical Signal Processing and Control* **79**: 104234.

Omuya, E.O., Okeyo, G.O. and Kimwele, M.W. (2021). Feature selection for classification using principal component analysis and information gain, *Expert Systems with Applications* **174**: 114765.

Parlett-Pelleriti, C.M., Stevens, E., Dixon, D. and Linstead, E.J. (2023). Applications of unsupervised machine learning in autism spectrum disorder research: A review, *Review Journal of Autism and Developmental Disorders* **10**(3): 406–421.

Paul, D., Jain, A., Saha, S. and Mathew, J. (2021). Multi-objective PSO based online feature selection for multi-label classification, *Knowledge-Based Systems* **222**: 106966.

Qasim, O.S., Al-Thanoon, N.A. and Algamal, Z.Y. (2020). Feature selection based on chaotic binary black hole algorithm for data classification, *Chemometrics and Intelligent Laboratory Systems* **204**: 104104.

Qi, X., Zhu, Y. and Zhang, H. (2017). A new meta-heuristic butterfly-inspired algorithm, *Journal of Computational Science* **23**: 226–239.

Rahman, M.M., Usman, O.L., Muniyandi, R.C., Sahran, S., Mohamed, S. and Razak, R.A. (2020). A review of machine learning methods of feature selection and classification for autism spectrum disorder, *Brain Sciences* **10**(12): 949.

Raj, S. and Masood, S. (2020). Analysis and detection of autism spectrum disorder using machine learning techniques, *Procedia Computer Science* **167**: 994–1004.

Sahu, S.K. and Verma, P. (2022). Classification of autistic spectrum disorder using deep neural network with particle swarm optimization, *International Journal of Computer Vision and Image Processing* **12**(1): 1–11.

Sayed, G.I., Hassanien, A.E. and Azar, A.T. (2019). Feature selection via a novel chaotic crow search algorithm, *Neural Computing and Applications* **31**: 171–188.

Shami, T.M., Mirjalili, S., Al-Eryani, Y., Daoudi, K., Izadi, S. and Abualigah, L. (2023). Velocity pausing particle swarm optimization: A novel variant for global optimization, *Neural Computing and Applications* **35**(12): 9193–9223.

Spencer, R., Thabtah, F., Abdelhamid, N. and Thompson, M. (2020). Exploring feature selection and classification methods for predicting heart disease, *Digital Health* **6**, Article no. 205507620914777, DOI: 10.1177/2055207620914777.

Thirumoorthy, K. and Muneeswaran, K. (2021). Feature selection using hybrid poor and rich optimization algorithm for text classification, *Pattern Recognition Letters* **147**: 63–70.

Uddin, M.J., Ahamad, M.M., Sarker, P.K., Aktar, S., Alotaibi, N., Alyami, S.A., Kabir, M.A. and Moni, M.A. (2023). An integrated statistical and clinically applicable machine learning framework for the detection of autism spectrum disorder, *Computers* **12**(5): 92.

Umamaheswari, K. and Parthiban, L. (2020). Improved whale optimization algorithm based feature selection with fuzzy rule base classifier for autism spectrum disorder diagnosis, *International Journal of Advanced Research in Engineering and Technology* **11**(11): 41–55.

Wang, H., Jing, X. and Niu, B. (2014). A weighted bacterial colony optimization for feature selection, *Intelligent Computing in Bioinformatics: 10th International Conference, ICIC 2014, Taiyuan, China*, pp. 379–389.

Wang, H., Tan, L. and Niu, B. (2019). Feature selection for classification of microarray gene expression cancers using bacterial colony optimization with multi-dimensional population, *Swarm and Evolutionary Computation* **48**: 172–181.

Zhang, F., Wei, Y., Liu, J., Wang, Y., Xi, W. and Pan, Y. (2022). Identification of autism spectrum disorder based on a novel feature selection method and variational autoencoder, *Computers in Biology and Medicine* **148**: 105854.

Zhao, J., Zhang, D., He, Q. and Li, L. (2023). A hybrid-strategy-improved dragonfly algorithm for the parameter identification of an SDM, *Sustainability* **15**(15): 11791.

Anandkumar Ramakrishnan is currently an associate professor at Sri Manakula Vinayagar Engineering College. His research interests include artificial intelligence and networks.

Rajakumar Ramalingam is currently an associate professor at the Vellore Institute of Technology. His research interests include optimization algorithms and wireless sensor networks.

Padmanaban Ramalingam is currently an assistant professor at the Vellore Institute of Technology. His research interests include deep learning and cyber security.

Vinayakumar Ravi is currently an assistant research professor at Prince Mohammad bin Fahd University. His research interests include deep learning and cyber security.

Tahani Jaser Alahmadi is currently an assistant professor at Princess Nourah bint Abdulrahman University. Her research interests include data analysis and mining.

Siti Sarah Maidin is currently an associate professor at INTI International University. Her research interests include deep learning and cyber security.

Received: 2 April 2024
Revised: 31 May 2024
Accepted: 23 August 2024