International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
IJISAE
ISSN:2147-6799 www.ijisae.org Original Research Paper

Identification of Autism Spectrum Disorder Using Modified Convolutional Neural Network (MCNN) and Feature Selection Techniques

S. Saravana Kumar[1], K. Selvakumar[2] and V. Senthil Murugan[3]

Submitted: 20/07/2023 **Revised**: 10/09/2023 **Accepted**: 22/09/2023

Abstract: ASDs (Autism Spectrum Disorders) are neurological disorders that impact people's lifetime capacity to connect or interact with others. Autisms are behavioural diseases where symptoms often erupt in the first two years. The problems may bein at very early ages in patients of ASDs and extend into adolescence. The previous research developed a method called AGWO (Adaptive Grey Wolf Optimisation) to find the most important characteristics and effective classification methods in ASD datasets. However, because of the high training complexity of SVM (Support Vector Machines), this model employed a single classification-based prediction method, which is not appropriate for classifying big data sets. In order to effectively identify ASD, this study effort developed M-AECNN (Modified Auto Encoder Convolution Neural Networks) based classification and dimensionality reduction algorithms. In this work introduced a AGWO to identify the most significant attributes and efficient classification techniques in ASD datasets. Initially the SMOTE based pre-processing approach is applied for removing the irrelevant data in ASD dataset. Subsequently, The AGWO algorithm keeps going through this process until it finds the characteristic that has the lowest classification recall and accuracy. Finally, M-AECNN-based classification is used to determine if a dataset instance is ASD. The highest performing classifier for these binary datasets was identified by experimental examination of ASD datasets from toddlers, children, adolescents, and adults, taking into account recall, precision, F-measures, and classification errors. The dragonfly optimisation is introduced in this paper for optimising the overfitting in AECNN to increase performance.

Keywords: *ASD dataset, Adaptive Grey Wolf Optimization (AGWO), Dragonfly Optimization, Modified Auto Encoder Convolutional Neural Network (M-AECNN) and SMOTE.*

1. Introduction

Statistics provided by the WHO show that 0.63% of children have an ASD diagnosis [1]. ASD starts in childhood and spreads to adolescents and adults as it gets worse. Typically, symptoms start to show up in initial years (5) and treatment costs of treating neuro developmental diseases are substantial. The hallmarks of ASD include a persistent lack of social contact and stereotyped conduct, which is frequently followed by a general decline in communication abilities [2]. Neurological and genetic factors are linked to ASD. Social interaction, the capacity for thought and imagination, repetitive habits, and interpersonal communication issues are all ways that ASD-related characteristics of ASDs are displayed. Along with behaviour and learning, it has an impact on how people express themselves and interact with others. A kid experiences the symptoms and warnings while they are very young. It is a chronic illness that has no known treatment [3]. Due to the rise in ASD cases globally as well as the difficulty and expense of identifying a patient, ASD has a huge financial effect**.**

Additionally, it takes a long time to receive an official diagnosis; for example, the average wait period in the United Kingdom is more than 3 years [4]. As a result, researchers in the domains of psychology, behavioural science, and psychiatric health have created self- and parent-administered screening techniques that, in their early stages, enable people to identify potential autistic features [5]. Here are some examples of screening techniques: Screening Tools for Autism in Toddlers and Young Children (STAT), Childhood Autism Rating Scale (CARS-2), and Autism Spectrum Quotient (AQ).7–10. The availability and usage of screening ASDs are crucial as they shorten formal clinical evaluations and assist in understanding resources and services required in terms of support like special education, speech therapies and supporting work environments [6]. Available screening tools are diagnostic in nature and need intelligence in identifying substantial number of items. Hence, these techniques have been criticised for requiring too much time.

[1]Research Scholar, Department of Information Technology, Faculty of Engineering and Technology, Annamalai University, Chidambaram, India
[2]Professor, Department of Information Technology, Faculty of Engineering and Technology, Annamalai University, Chidambaram, India
[3]Department of Networking and communications, Faculty of Engineering and Technology, SRM Institute of science and Technology, Kattankulathur, India
**Corresponding Author Email: saravana.coolya@gmail.com*

Early diagnosis and treatment are possible for ASD. Clinical diagnosis of ASD depends critically on early identification [7].Personalised treatment plans can then be developed based for enhancing quality of life in young patients with early diagnostics. Unfortunately, determining an ASD diagnosis may be time-consuming and expensive. Researchers have recently been motivated to develop more efficient screening techniques by the rise of ASD diagnoses worldwide [8]. The development of contemporary technology has allowed us to store a vast amount of data. To create judgements based on the gathered data, data mining is a crucial duty. This relates to artificial intelligence and ML which have been performing well and have become significant in practical sciences like biomedicine [9] and healthcare. In clinical decision-making and diagnostics, ML methods are used or recommended to support data interpretation.

ML (machine learning) techniques are employed or advised to help data interpretation in clinical decision-making and diagnostics. CNNs (Convolution Neural Networks) have lately been revelling in representative learning and categorizations with a large range of free parameters that work well in a variety of applications. Additionally, CNNs models can handle a large number of free parameters and have improved feature extraction accuracy [10]. Convolution, fully connected, normalized, and pooling layers are parts of CNNs. Using fMRI (functional magnetic resonance imaging), CNNs can analyse brain biomarkers in patients affected with ASD. Identifying biomarkers is crucial for both diagnosis and therapy of ASDs and Multi-channel CNNs based on a patch-level data-expanding approach have been suggested for these tasks [11]. Multivariate voluminous dataset was condensed and AEs examined functional connectivity pattern linked to ASDs. In order to find the most important properties and effective classification methods in ASD datasets, the previous study developed an AGWO. However, because of the high training complexity of SVM, this model employed a single classification-based prediction method, which is not appropriate for classifying big data sets. Hence, this research work introduces M-AECNN based classifications and reduction of dimensionalities for efficient identifications of ASD.

Section 2 of the research project discusses several current ML strategies for the identification of ASD. The rest of this research work is organized as follows. Section 3 describes the methodology's suggested strategy. Section 4 presents the findings and comments. Section 5 discusses the findings and more study.

2. Literature Review

In this section review the some of the recent techniques for the detection of ASDs using some advanced ML and feature selection techniques for efficient prediction.

ASD was automatically detected using CNNs and a brain imaging dataset by Sherkatghanad et al. [12]. fMRI data of ABIDE (Autism Brain Imaging Exchange) dataset was used in the study to identify individuals with ASDs. Functional connectivity patterns were examined to distinguish ASDs in patients. The proposed technique's experimentations on ABIDE I and CC400 functional parcellation brain atlas datasets showed that the suggested model identified ASDs with 70.22% accuracy. The proposed CNNs used fewer parameters with reduced processing. Oosterling et al. [13] examined integrated early detections of ASDs in screening where training addressed Autistic Traits Questions and autism. Creating a multidisciplinary diagnostic team, following a defined referral methodology, and a controlled research with children from two locations (N=2793, age range 0–11 years) assessed the programme. Differences in the mean ages of ASDs in diagnosis and the percentage of children detected before the age of 36 months was the most important outcome factor. Khowaja et al. [14] identified autism early using a Screening Tool for Autism in Toddlers and Young Children in Level 1 in their improved two-tiered screening systems that incorporated Level 1 (Modified Checklist for Autism in Toddlers, Revised with Follow-Up) and Measures. 109 kids showed positive in Level 1 screening results and diagnostic assessments were Level 2 screening. The study's findings showed that their two-tiered screening reduced false positives when compared to Level 1 screening.

Guo et al. [15] proposed high-dimensional whole-brain resting-state FC pattern classifications of ASDs vs. controls with normal developments, recommended feature choices utilising DNNs (DNN-FS). The study employed several trained sparse AEs and representations of brain FC patterns with high quality, feature choices for assisting DNNs. DNN-woFS (DNN classifications without feature selections) were used to provide comparative findings utilising a variety of architectural configurations (i.e., with varied hidden layers/nodes counts). Their outcomes demonstrated that DNN-FS with three hidden layers and 150 hidden nodes (3/150) yielded classification accuracies with maximum value of 86.36%. Abdolzadegan et al. [16] suggested detected ASDs early using the EEG data. 34 ASD-affected children aged between 3 to 12 and made up the study's population. Their suggested method defined EEG signals by combining linear/non-linear features like Power Spectrums, Wavelet Transforms, FFTs (Fast Fourier Transforms), Fractal Dimensions, Correlation Dimensions, Lyapunov Exponents, Entropies, De-trended Fluctuation

Analyses, and Likelihood Synchronisations. Additionally robust removal of artefacts and density-based clustering and while applying features multiple other factors were considered including MI (Mutual Information), IGs (Information Gains), mRmR (Minimum-Redundancy Maximum-Relevancy), and GAs (Genetic Algorithms). Finally, SVMs and KNNs (K-Nearest Neighbours) in investigations showed that KNNs- had classification rates of 72.77% compared to SVMs 90.57%. Additionally, sensitivities of proposed technique was 91.96% for KNNs and 99.91% for SVMs where BFS (backward feature selections) selected of 4540 people’s features from modules 2 and 3, Kosmicki et al. [17] constructed ML algorithms obtained 98.27% and 97.66% accuracy, respectively for 9 from recorded 28 behaviours of module 2 items and 12 from 28 behaviours gathered by module 3 in establishing ASD risks. The number of behaviors reduced by more than 55% across both modules, with only a slight loss in accuracy, raises the possibility that computational and statistical approaches might be used to speed up the identification of ASD risk factors. The development of mobile, parent-directed clinical triage and/or preliminary risk assessments can be applied at large facilitating diagnostics.

Alzubi et al. [18] suggested reliable hybrid feature selections for obtaining valuable SNPs and their optimal subsets. Their recommended solution is built on the CMIM (Conditional Mutual Information Maximisation) methodology and SVM-RFEs (SVM Recursive Feature Eliminations). They used SVM, NB (Naive Bayes), LDA (Linear Discriminant Analysis), and k-NN and compared it with mRMR, CMIM, and ReliefF feature selections. The experimental findings show that the selected feature selection strategy is effective, surpassing all other feature selection methods that were examined and reaching up to 89% classification accuracy for the dataset employed. For enhanced diagnosis purposes, Hossain et al. [19] created a diagnosis method employing the categorization approaches that are now accessible. The study examined ASD datasets for toddlers, kids, teens, and adults to obtain best classifications and feature sets. Their test findings demonstrated that MLPs (multilayer perceptrons) based classifications beat existing classification methods and achieved 100% accuracy with fewest features. Additionally, "relief F" selected better features.

A number of feature extraction techniques were provided by Liu et al. [20] in an effort to thoroughly assess their predictive effectiveness. Additionally, you put forth a framework for predicting ASDs that learns its prediction model from the features that have been labelled. A test participant is also requested to see the facial photos with recorded eye gaze positions throughout the testing phase. In order to establish whether the test individual may have ASD or not, a threshold is defined based on the learnt model's predictions for the image-level labels. Despite the fact that ASD prediction is inherently challenging, experimental results reveal statistical significance of the projected results, indicating a potential future for this paradigm. Küpper et al. [21] improved identifications of ASD behavioural features from ADOS Modules using ML algorithm (SVM) in routine clinical samples (N = 673), adolescents (n = 385) and individuals with suspected ASDs but other best estimates or non-psychiatric diagnostics (n = 288). For all samples and age subgroups (adolescents vs. adults), the condensed subsets of 5 behavioural features showed good specificity and sensitivity and achieved performances comparable to existing ADOS algorithms, with no discernible differences in overall performances. By encouraging future initiatives to offer cutting-edge diagnostic techniques for the identification of ASD based on the given information and their findings, we may be able to enhance the challenging ASD diagnostic procedure by helping doctors with the challenging task of differential diagnosis. Employing independent samples of 2616 young people Duda et al. [22] evaluated using ADOS from five data sources, encompassing spectrum (n = 2333) and non-spectrum (n = 283) individuals, suggested OBCs (observation-based classifiers) were put forth. They compared the results of OBCs to the first and second rounds of the ADOS algorithms for their best estimations in the severity comparison score metrics used by ADOS-2. The OBC demonstrated much higher sensitivity and specificity (r=0.814 and r=0.779, respectively) when compared to ADOS-G, ADOS-2. With sensitivity of 97.1% and specificity of 83.3%, and accuracy=96.8%. OBCs likely serves as both a categorizations and gauge of severity in features due to substantial correlations between OBCs and comparison scores (r=0.628).

3. Proposed Methodology

This research work introduces M-AECNN based classifications and dimensionality reductions for detection efficiencies of ASDs where AGWO identify significant attributes that can assist efficient classifications in ASD datasets. Least features with maximum classification recalls and accuracies were obtained using pre-processing methods based on SMOTE and repeated using AGWO algorithm. To determine if an instance in the dataset contains an ASD or not, classification is lastly carried out using an M-AECNN-based algorithm. In order to improve overfitting in AECNN and boost speed, the dragonfly optimisation is introduced in this article. Figure 1 shows how the suggested technique works in action.

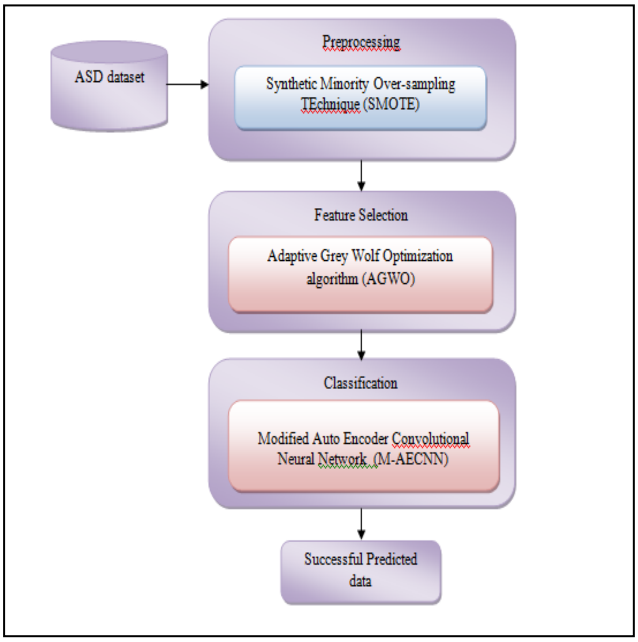

Fig 1. The process of the proposed methodology

3.1. Preprocessing

Before training and analysing the data using ML algorithms, pre-processing the data is a typical initial step. ML algorithms are only as good as the data they are fed. In order to have enough consistency to get the best outcomes, it is essential that data be structured correctly and have important features [23]. As can be seen in, there are several processes involved in pre-processing the data for ML algorithms for computer vision, including normalising the supplied data. These seek to eliminate some of the insignificant characteristics that serve as identifiers between various sets of data. Most classification algorithms try to get clean examples to learn from and make the boundary of each class as clear-cut as possible in order to perform better prediction. The majority of classifiers find it easier to learn to categorise synthetic instances that are distant from the boundary than those that are near to it. Based on these findings, we introduce a novel, cutting-edge method called SMOTE (Synthetic Minority Oversampling Technique) for preprocessing unbalanced training sets. SMOTE generalisation attempts to precisely identify the borderline and provide pure synthetic samples. The following are the two steps of our suggested process:

First stage, Synthetic instances based on the formula below applied SMOTE algorithm [24]:

$$N = 2 * (r - z) + z \qquad (1)$$

where N stands for initial synthetic instance numbers (newly generated), r, represents majority class sample counts, and z, refers to minority class sample counts.

Second stage, Here, artificial samples produced by SMOTE that are more closely related to the majority class than the minority and artificial cases that are more closely related to the borderline are eliminated.

After the provided data has been cleaned, the feature selection procedure, which is described in the next section, is carried out.

3.2. Feature Selection using grey wolf optimization

In order to prevent the impact of noise and irrelevant factors on prediction findings, the feature selection procedure comprises the effective selection of a subset of variables [25]. The entire dataset may be processed using filtering, wrapper, and embedding approaches to provide a selection of useful characteristics. The diagnostic system performs better when the proper feature set is chosen..

- **GWO (Grey Wolf Optimization)**

Based on the newly announced GWO, this paper suggests a brand-new optimisation technique, AGWO where grey wolves' are mimicked. During developments of GWO, fittest solutions are utilised where alpha wolves statistically define social structures of wolves. The second- and third-best solutions are referred to as beta and delta wolves [26] and also omega wolves are considered feasible answers. The hunting (optimisations) of GWO algorithm are directed by wolves which pursue these three wolves. Social leaderships can be mimicked using mathematical equations.

Grey wolves have unique social behaviours, including collective hunts and social hierarchies. Grey wolf hunting phases are detailed below:

• Following, surrounding, and pestering the prey until it stops moving; tracking, pursuing, and approaching the target.

• Slashing at the prey

These steps are shown in Fig. 2.

Fig. 2. Grey wolves' hunting techniques include the following: (A) pursuing, approaching, and tracking prey; (B-D) pursuing, bothering, and surrounding; and (E) attacking stationary targets.

3.2.1. Algorithm and mathematical model

This section presents mathematical models of wolves' social hierarchies, tracking, encircles, and attacks on preys and subsequently GWO algorithm is detailed.

a) Social hierarchies:

In GWO, fittest solutions are called alpha α in order to quantitatively replicate social structures of wolves. Hence, Beta and Delta are second and third best answers while balance answers are omega controlling hunting (optimisations) of GWO algorithm.

b) Encircling preys:

Encircling of the prey by gray wolves can be represented as:

$$\vec{D} = \left|\vec{C}.\overrightarrow{X_p}(t) - \vec{X}(t)\right| \tag{2}$$

$$\vec{X}(t+1) = \overrightarrow{X_p}(t) - \vec{A}.\vec{D} \tag{3}$$

where t, coefficient vectors imply current iterations, and, $\overrightarrow{X_p}$ indicate prey positions of preys, and indicate positions of grey wolves. The vectors and are calculated as follows:

$$\vec{A} = 2\vec{a}.\overrightarrow{r_1} - \vec{a} \tag{4}$$

$$\vec{C} = 2.\overrightarrow{r_2} \tag{5}$$

Where the range of the random vectors is [0, 1] and the components of the random vectors are linearly decreased from 2 to 0 across the repetitions. Fig. 3 provides a two-dimensional position vector and a few potential neighbours to illustrate how equations (2) and (3) work. As seen in this figure, a grey wolf in the location of (X,Y) may adjust its location to match that of the prey at (X*,Y*). By altering the values of vectors, several locations surrounding best agents can be achieved with respect to their present positions. For example, the result (X*-X,Y*). Because of random vectors, wolves move to any positions as indicated in Fig. 3. and grey wolves update their positions using equations (2) and (3).

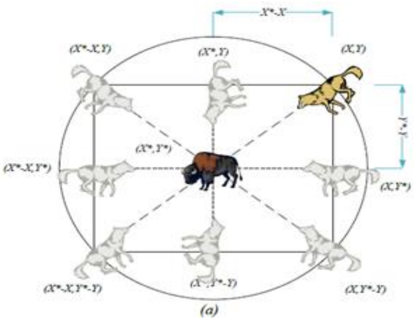

Fig. 3. 2D position vectors and their possible next locations

Similar concept may be used in search spaces with n dimensions where grey wolves hypercube or hyper sphere obtained top results.

c) **Hunting:**

During hunting, grey wolves encircle their preys with alpha wolves in charge. Beta and delta wolves also occasionally take lead. However, as ideal preys are not known they are placed abstractly in search spaces. In mathematically simulations alpha wolves are assumed to be the best candidate solutions and beta, delta wolves also have superior knowledge of likely prey locations. Hence, only three top successful solutions are maintained and other search agents including omegas which update their positions in accordance with positions of first three search agents. This is depicted mathematically:

$$\overrightarrow{D_\alpha} = \left|\overrightarrow{C_1}.\overrightarrow{X_\alpha} - \vec{X}\right|, \overrightarrow{D_\beta} = \left|\overrightarrow{C_2}.\overrightarrow{X_\beta} - \vec{X}\right|, \overrightarrow{D_\delta} = \left|\overrightarrow{C_3}.\overrightarrow{X_\delta} - \vec{X}\right| \tag{6}$$

$$\vec{X}(t+1) = \frac{\overrightarrow{X_1}+\overrightarrow{X_2}+\overrightarrow{X_3}}{3} \tag{7}$$

c) Attacking preys (exploitations):

After preys stop moving, grey wolves strike to terminate the hunt. The values are lowered mathematically to simulate approaching the prey. Alternatively, random variables are in the range [-a,a] and decrease from 2 to 0 in repetitions. Search agents' future positions can be anywhere between their present locations and prey locations when random values are between [-1,1]. The sresults of |A|1 causes wolves to charge their prey as shown in Figure 4(a).

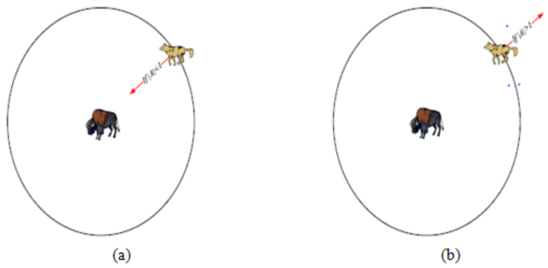

Fig. 4. Attacking prey versus searching for prey

GWO algorithm allows search agents to update their locations based on alpha, beta, and delta positions and uses operators to strike in prey directions. While using these operators, GWO may become entangled in local solutions. The proposed circumferential approaches do, to some extent, demonstrate explorations, however GWO requires more operators to do so.

d) Searches for preys (explorations):

The alpha, beta, and delta directions are where grey wolves mostly look for prey. They split apart to search for prey before coming together to attack it. We utilise random numbers greater than 1 or less than -1 to make the search agent diverge from the prey in order to mathematically approximate divergence. This encourages research and makes it possible to find the GWO algorithm globally. Furthermore, Fig. 4(b) reveals that grey wolves deviate away from their victim in quest of a fitter prey when $|A|>1$. GWO also has a feature called that encourages exploration. According to Equation (5), the vector has random values between [0, 2]. Grey wolves primarily hunt in the alpha, beta, and delta directions. They split apart to look for prey before joining forces to attack it. To approximate divergence, we use random integers higher than 1 or less than -1 to cause the search agent to diverge from the prey. This promotes research and allows the GWO algorithm to be found internationally. Additionally, Fig. 4(b) demonstrates that grey wolves divert from their victim in search of a more fit prey when $|A|>1$. Additionally, GWO features a function called that promotes exploration. According to Equation (5), the vector has random values in the range [0, 2].

Stochastically ($C>1$) is emphasise or ($C<1$) deemphasize importance of prey in impacting distances in Equation (2) by offering random weights for prey. Due to increased explorations and avoidance of local optima, GWO is able to be more unpredictable throughout the optimisation process as a result. Notably, the decline in C is non-linear when compared to the decline in A. We deliberately require C to deliver random values on a regular basis in order to showcase exploration throughout both the initial rounds and the final iterations. This element is particularly helpful when local optima stall during the latter stages.

The C vector may also be viewed as the outcome of natural obstructions placed in the route of a predator. In general, wolves have inherent problems in their hunting methods, which seriously limits their capacity to approach prey quickly and readily. This specific function is carried out by the vector C.

A wolf's location may, at random, give its prey weight, making the wolf's approach more difficult and far, or vice versa. Last but not least, the GWO generates random populations of grey wolves (possible solutions) in its inital search phases. The alpha, beta, and delta wolves scout out potential prey locations throughout a number of rounds. The prey's distance varies depending on each potential reaction. The value of an is changed from 2 to 0 to promote exploration and exploitation. Solutions diverge when $| a |>1$ and converge when $| a |<1$. GWO stops executions on finding end criteria.

3.2.2. AGWO

As soon as |A|1, the GWO algorithm may be used. When random values of A are in the range [-1,1], the search agents are aided in convergent towards an estimated position of prey provided by alpha, beta, and delta solutions.

The location of a search agent in the future might be anywhere between its present location and that of the prey.The beginning population for the GWO algorithm's optimisation process is a set of random solutions. The top three results are kept and are known as alpha, beta, and delta solutions throughout the optimisation process. Every wolf omega (with the exception of, and) has the position active. The iteration's length is accompanied by a linear decrease in the parameters a and A. As a result, when $|A|>1$ and when $|A|<1$, search agents often diverge from prey and concentrate on prey, respectively. When end conditions are met, the locations and scores of alpha solutions are returned as best optimisations. GWO is enhanced with two additional pieces to do multi-objective optimisations. The components are quite similar. The first are storehouses for previously recognised non-dominated Pareto optimal solutions. While searching through archives, second components are systems for selecting alpha, beta, and delta solutions as leaders. The archives are simple storage systems that can save or retrieve previously discovered non-dominated Pareto optimal solutions. The major modules are archive controllers, which manage archives when solutions request admittance or when archives are full. Remember that the archive has a maximum number of members.

The current non-dominated solutions and the archive inhabitants are compared during the iteration phase. The likelihood that one will be eliminated increases with the number of solutions in the hypercube (section). The most crowded segments are chosen first if the archive is already full, and a solution is at random removed from one of them to make room for the new solution. A special example involves inserting a solution outside of the hypercubes. All of the sections in this instance have been expanded to include the new methodologies.

Hence, alternative solution elements are likewise subject to modification. The second set of elements are the methods for choosing leaders. GWO uses three stages: alpha, beta, and delta. These leaders guide other search agents to active search spaces where they can uncover solutions that are closer to the overall ideal. However, due to the Pareto optimality rules outlined in the preceding paragraph, comparing solutions in a multi-objective search space is challenging. This issue is addressed in the approach for selecting leaders. As previously stated, the best non-dominated solutions to date have been retained. The leader selection component, which picks the least crowded parts of the search space, presents one of the non-dominated solutions as an alpha, beta, or delta wolf. The decision is determined using a roulette wheel, with each hypercube having the following probability:

$$P_i = \frac{c}{N_i} \quad (8)$$

where *c* is a constant number greater than one and *N* is the number of obtained Pareto optimal solutions in the *i-th s*egment.

Eq. (8) implies that less congested hypercubes are more likely to suggest new leaders and with fewer recognised solutions in hypercubes, it becomes easier to select hypercubes for selecting leaders. Given that three leaders need to be selected, certain special conditions nedd to be considered. Sections with fewest solutions have the three solutions assigned to alpha, beta, and delta solutions. Second least crowded hypercubes are located to select other leaders. Delta leader should be second least congested hypercubes and if scenarios are the same they have singular solutions.

AGWO continuously directs searches towards uncharted/unexposed regions of search spaces since leader selections favour least congested hypercubes and offers leaders from other segments if there aren't enough (less than 3) in least packed segments.

Algorithm 1. Pseudocode of proposed adaptive GWO

Initialise the Xi populations of grey wolves Xi (i = 1, 2, ..., n)

Set up a, A, and C at zero.

Compute search agent objective values.

Locate the non-dominated solutions, then use them to initialisethe archive.

Xα=SelectLeader(archive)

Xδ= SelectLeader(archive)

Alpha will not be included in the archive ifX=SelectLeader(archive)

Remove beta from the repository.

SelectLeader(archive) = X

t=1;

while (t < Max number of iterations)

for each search agent

Equations (3) through (8) update the location of the currentsearch agent.

end for

Revise a, A, and C.

The objective values of each search agent should be determined.Search for the undominated solutions.

Revisit the archive in light of the discovered non-dominatedsolutions.

Whenever the repository is full

To delete one of the current archive's members, use the gridapproach.

Update the response and add it to the database.

end if

Whenever the repository is full

To delete one of the current archive's members, use the gridapproach.

Update the response and add it to the database.

end if

If any of the recently uploaded archived solutions are foundoutside of hypercubes

Adapt the grids to the new solution or solutions.

end if

Xα =SelectLeader(archive)

Keep alpha out of the archive.

Xβ= SelectLeader(archive)

Exclude beta from the archive

Xδ= SelectLeader(archive)

t=t+1

end while

return archive

3.3. Classification using M-AECNN

In this study, an SECNN-based classification for autism spectrum disorder prediction is established. The AE-CNN, however, has an overfitting issue. So, to address the aforementioned issue, this study proposed an Enhanced Dragonfly Optimisation technique.

3.3.1.AE-CNN (Auto Encoder Convolutional Neural Networks)

AE (Auto Encoder) - CNNs was implemented in the current study for the effective diagnosis of citrus illness. As an example, examine figure 5's three hidden layers in two-dimensional AEs based CNNs [27]. Unsupervised learning implies training ML model F to produce same data as inputs x in such a way that x F(x; w), where w denotes weights of ML models. Mathematically, weights w are optimised throughout training phases to reduce errors E, resulting in w = argminw[E(x, F(x, w)]. It's important to note that internal structures R, shown in figure 5 by green sections, are lesser than inputs or outputs R x. High-dimensional data have been effectively translated into low-dimensional latent vectors if models have been trained to generate data that are almost identical to inputs. These relationships can be expressed as,

$$\gamma = F_e(x), \qquad x \approx F_e(\gamma) \qquad (33)$$

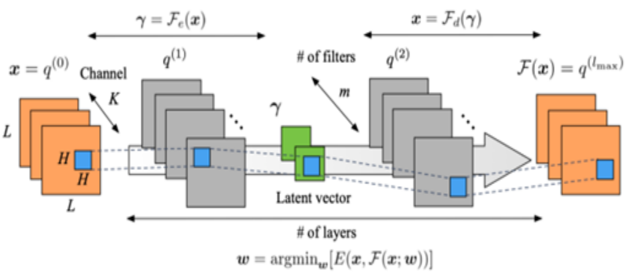

Fig. 5. 2D CNNs based AEs with three hidden layers.

where Fe and Fd are the AE's encoder and decoder components, respectively, as seen in figure 5. In passing, take note that the type of flow field being dealt with has a significant impact on how far we can suppress the dimension. The Adam optimizer is used to update the weights throughout the iterative training phase in the current study. The error function E is the L2 norm error. CNNs are instructed in the concept of weight sharing. Convoluting the filter h (l) (shown in figure 4 as the blue H H squares) with the output of the upstream layer q (l1) yields the output q (l) of a CNN node at layer l, location (i, j), and filter index m.

$$q_{ijm}^{(l)} = \varphi(b_m^{(l)} + \sum_{k=0}^{K-1}\sum_{p=0}^{H-1}\sum_{s=0}^{H-1} h_{pskm}^{(l)}\, q_{i+p-C\ j+s-Ck}^{(l-1)}) \qquad (9)$$

where C = floor(H/2), K stands for filter counts in convolution layers of input/output layers and correspond flow variables counts for points, b (l) m represents bias, and ϕ stands for activation functions, which are monotonically increasing nonlinear functions. Hyperbolic tangent functions ϕ(s) = (e s − e −s) · (e s + e −s) −1 are utilized for activations in this work, After studying different types of nonlinear activation functions, the hyperbolic tangent function was chosen. It was also demonstrated that the performance could be reached using either the sigmoid function or the ReLU function in its stead. The filter coefficients h (l) pskm are optimised as weights during CNN training to provide the appropriate output. Figure 5 depicts the communication of the improved filter coefficients inside the same network layer. Be mindful that CNNs operate on the assumption that neighbouring pixels in a image have no significant association.

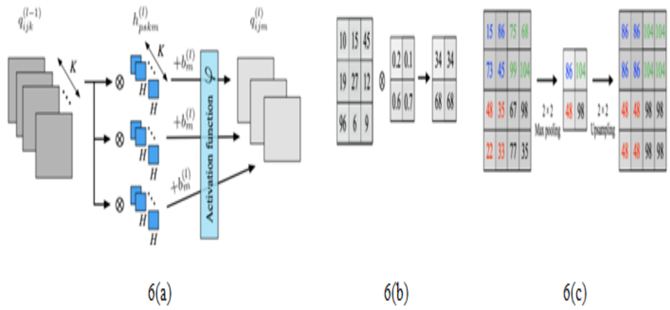

Fig. 6. Displays a single convolution layer's structure schematic CNNs (a) structure of a singular layer of CNNs; (b) convolution operation; (c) pooling and upsampling operations

Max pooling operations encode and upsampling operations are implemented for decodes and provide dimension reductions and extensions necessary for constructions of AE, as shown in the schematics of figure 6(c). AEs may minimise the size of the input photos while maintaining robustness against rotation and translation using the max pooling technique. In the decoder step of the upsampling procedure, closest neighbour interpolation is used to convert values of low dimensional maps into high dimensional images [28]. Although CNNs were initially utilised in computer science, they are currently being used to handle a wide range of high-dimensional data including fluid dynamics. According to reports, low dimensional maps have considerable advantages over methods that rely on linear theories because of setting up nonlinearities through the use of activations. However, AE-based approaches usually lack interpretability; in particular, understanding the physical relevance of the latent vector created by nonlinear filter operations can be difficult. Because AE-based modes do not have the same concept as eigenvalues or singular values in linear mode decomposition methodologies, we may understand the contribution of each latent vector by studying their energy containing ratio. This is due to the fact that these AE-based modes are not orthogonal to one another. To address this issue, this study proposed an Elman neural network for dealing with this sort of energy-containing ratio.

3.3.2. EDO (Enhanced Dragonfly Optimization) Algorithm

DOA is a population-based optimizer that has recently gained traction. The DOA algorithm is based on dragonfly migratory and hunting habits. In a static swarm (feeding), all of the swarm's members fly in close formation over a constrained area in search of food sources. According to a theory, dragonflies employ dynamic swarming as a component of their migration plan [29]. Because the dragonflies prefer to fly in larger groups, the swarm may now move. Figure 7's groupings are both static and dynamic. Furthermore, in other swarm-based systems, the operators of DA achieve two essential concepts: intensification, which is suggested by dynamic swarming activities, and diversification, which is prompted by static swarming activities.

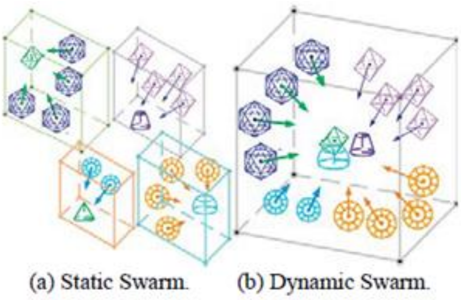

Fig 7. Dynamic and Static Dragonflies.

DA, where X are position vectors, Xj imply neighbours of X in positions j, and N refers to the neighbourhood sizes and are characterised by five behaviours:

Separations: Dragonflies employ this tactic to set themselves apart from other agents. This technique is written as follows (4):

$$S_i = \sum_{j=1}^{N} X - X_i \quad (10)$$

Alignments: Demonstrate how agents align their velocities with neighbouring dragonflies' velocities. This idea is based on (11) where Vj denotes the j-th neighbor's velocity vector:

$$A_i = \frac{\sum_{j=1}^{N} V_i}{N} \quad (11)$$

Cohesions: implies members nature to go towards the nearest mass centre. In (12), this step is characterised as follows:

$$C_i = \frac{\sum_{j=1}^{N} X_j}{N} - X \quad (12)$$

Attractions: demonstrates how group members gravitate towards the food source. It is carried out to evaluate the attraction potential of both the food supply and the i-th agent.

$$F_i = F_{loc} - X \quad (13)$$

Distractions: demonstrates how dragonflies will often turn away from a possible adversary. The opponent and the i-th dragonfly are distracted in the following manner where Eloc is the opponent's position

$$E_i = E_{loc} + X \quad (14)$$

The fitness of food supply and position vectors in DA is updated by the fittest agent yet known. The positions and fitness levels of the enemies are likewise estimations based on the worst dragonfly. This knowledge will allow DA to avoid less viable areas of the solution space and focus on more promising ones. To update the position vectors of dragonflies, two criteria are used: the position vector and the step vector (X). The step vector in (15) indicates the direction of the dragonflies' travel, which is determined as follows:

$$X_{t+1} = (sS_i + aA_i + cC_i + fF_i + eE_i) + wX_t \quad (15)$$

Where s, w, a, c, f, and e depict weights of components and locations are computed using Eq. (16), where t iterations are :

$$X_{t+1} = X_t + X_{t+1} \quad (16)$$

The separation, alignment, and cohesion optimisations enable the DA to often provide a range of local and global searches. Another two variables that enable dragonflies to take advantage of fortunate opportunities while avoiding unfavourable ones are attraction and distraction.

Because of these five swarming tendencies, the DA algorithm is better [30].DA employs a V-shape transfer function to calculate the changing probability of dragonfly position. Unlike other binary metaheuristics, BDA does not force dragonflies to choose between 1 and 0. As a consequence, BDA had great exploration skills, which aided it in locating the necessary search space.

a) **Mutation Learning strategy (MLS) based Dragonfly Algorithm**

To update its location, the Modified Dragonfly (MD) adopts a mutation learning method (MLS) that incorporates the ideas of personal best and personal worst solutions (see Fig. 8). The dragonflies in the traditional DA coordinate the global best solution (food supply) and the global worst solution (enemy) for attraction and diversion. Incorporating the personal best and personal worst dragonflies into these activities is intended to boost the chance of food hunting and opponent fleeing actions.

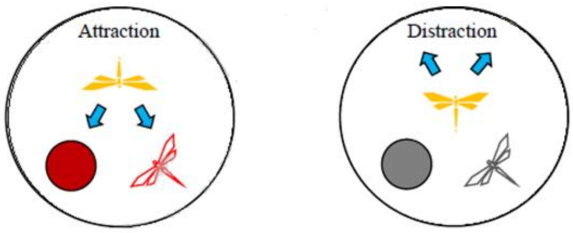

Fig. 8. Attractions and distractions behaviors in MLS-DA**.**

Unlike DA, the attraction and distraction of MLS are calculated using the following equations.

$$F_i = \frac{(Xpb_i - X_i) + (Xf - X_i)}{2} \quad (17)$$

$$E_i = \frac{(Xpw_i + X_i) + (Xe - X_i)}{2} \quad (18)$$

where Xpb_i refer to positions of best dragonflies, Xpw_i shows worst dragonflies, X refer to dragonflys' positions, Xf refer to food sources, and Xe refer to enemies. Furthermore, due of the hyper learning approach, the dragonflies may learn from both their own best and the greatest solutions accessible worldwide throughout the search phase. Figure 9 depicts the general concept of the learning approach. Instead of changing its location in reaction to swarming, the dragonfly attempts to replicate the best experiences it has had both locally and internationally.

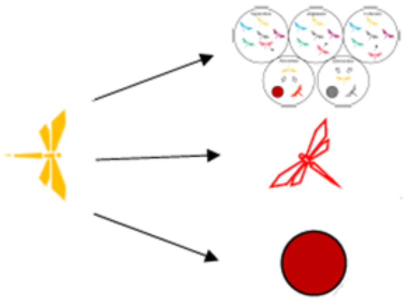

Fig. 9. General concept of the learning strategy**.**

MLS-DA's dragonfly positions are updated using:

$$X_i^d(t+1) = \begin{cases} \bar{X}_i^d & 0 \le r_1 < pl \\ Xpb_i^d(t) & pl \le r_1 < gl \\ Xf^d(t) & gl \le r_1 < 1 \end{cases} \tag{19}$$

$$\bar{X}_i^d = \begin{cases} 1 - X_i^d(t) r_2 < TF(\Delta X_i^d(t+1) \\ X_i^d(t) r_2 \ge TF(\Delta X_i^d(t+1) \end{cases} \tag{20}$$

where *X* represents positions of dragonflies, *Xpb* represents positions of best dragonflies, *Xf* represents food sources, *i* implies orders of dragonflies, *d* represents decision variable counts(dimensions), *t* implies current iterations, *r1* and *r2* stand for independent random values between 0 and 1. The global learning rate (gl) and the personal learning rate (pl) have fixed values between [0, 1].Equations (19) and (20) demonstrate the significance of the pl and gl in the learning process. The algorithm becomes prone to local optima trapping when pl and gl are too low and instead of searching for collective and individual optimal solutions. If pl and gl have excessive values, the position updating method will imitate DA. As a result, the selections of pl and gl are critical. The MLS-DOA pseudocode is presented in Algorithm 2.

Algorithm 2. Mutation Learning Strategy based Dragonfly Algorithm for parameter tuning

Input: Enter X, S, A, and E.

Output: Optimized learning parameters

1) Initialise the placements of N dragonflies and X at random.

2) Set the step vectors' initial value of *ΔX* to zeros

3) **while** (Not equal to Max iterations)

4) **for** *i* = 1 to number of dragonflies, *N*

5) Determine the dragonfly's fitness value (i-th).

6) Update Xpbi, your own best dragonfly.

7) Correct the Xpwi, my personal worst dragonfly.

8) **end for**

9) Update the opponent (Xe) and food source (Xf).

10) Update *s, α, c, f, e, and w*

11) **for** *i* = 1 to number of dragonflies, *N*

12) Use (10), (11) and (12) to calculate S, A, and C.

13) Use (13) and (14) to calculate F and E.

14) Use Mutation Learning Strategy to update step vectors in

(17) and (18).

15) Refresh the location of the dragonfly (*i*-th) using (19 & 20)

16) **end for**

17) **end while**

To determine if the suggested strategy is effective in identifying distinct ASD dataset groups (DATASET: https://www.kaggle.com/fabdelja/autism-screening-for+toddlers?select=Toddler+Autism+dataset+July+2018.csv). Four datasets—ADULT, CHILD, ADOLESCENT, and TODDLER—are utilised in this study for testing. The performance of the classifier is assessed using Conditional Mutual Information Maximisation (CMIM), Sequential Minimal Optimisation with SVM (SMO-SVM), Adaptive Grey Wolf Optimisation with SVM (AGWO-SVM), and a model based on M-AECNN.

In experiments, validation sets fine-tuned model's and training hyperparameters, while training sets optimized model's parameters. In order to evaluate the effectiveness of the various techniques in the prediction of ASD data, the research made use of a number of criteria that are frequently used in binary classification. Determine the rates of true positive (TP), false positive (FP), true negative

(TN), and false negative (FN) before computing the different performance indicators. Precision, which is the proportion of pertinent events retrieved among all events recovered, was the initial performance metric. Recall, which is characterised as the proportion of pertinent events recovered, is the second performance parameter. The effectiveness of a prediction technique can be evaluated using both accuracy and recall criteria, despite the fact that these measurements are typically incompatible. As a result, these two metrics may be combined and given equal weight to create the F-measure, a single metric. The last performance parameter was accuracy, which is defined as the percentage of properly anticipated occurrences compared to all instances projected. Table 1 shows a performance comparison of the recommended and current procedures.

Precisions are ratios of correctly found positive observations to total expected positive observations.

$$Precision = TP/(TP + FP) \quad (21)$$

Sensitivities or Recalls are ratios of correctly identified positive observations to total observations.

$$Recall = TP/(TP + FN) \quad (22)$$

F – measures are weighed averages of Precisions and Recalls and hence consider false positives and false negatives.

$$F - measure = 2 * (Recall * Precision)/(Recall + Precision) \quad (23)$$

Accuracies, computed in terms of positives and negatives as:

$$Accuracy = (TP + FP)/(TP + TN + FP + FN) \quad (24)$$

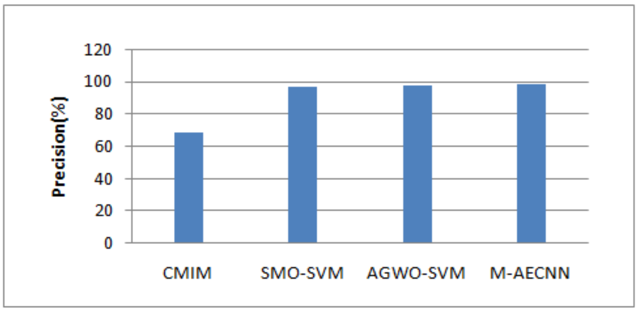

Fig.9. Comparative values of precisions between suggested and existing methods in classifications of ASDs

Figure 9 depicts an accuracy comparison of the proposed and current approaches for recognising ASD data. Overall, the findings demonstrated that the proposed classification model outperformed the machine-learning methods under consideration on the datasets, which is consistent with the earlier error rate and may be ascribed to the proposed model's non-redundant rule sets. According to the data, the proposed M-AECNN approach beats other current classification methods in terms of precision.

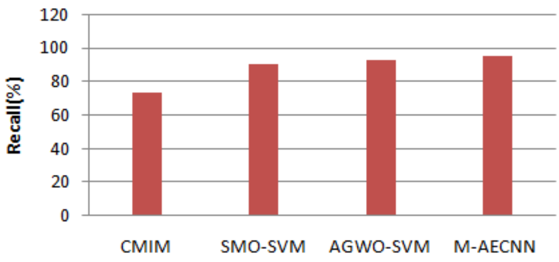

Fig.10. Comparative values of recall between suggested and existing methods in classifications of ASDs

Figure 10 depicts a memory comparison of the proposed and current approaches for detecting ASD data. The data used in this study is primarily concerned with diagnosing patients with ASD symptoms utilising a variety of variables that frequently impact the diagnosis. As a result, the prediction model is considered as a classification problem that can be addressed whether or not the person has ASD. As a result, the recommended supervised models were applied to the allotted task, and the results were analysed and evaluated. The feature selection method must be used prior to deploying these ML models in order to support the assessment findings and the models' correctness by eliminating the weak variables from the databases. The supervised classification models and proposed AGWO-based feature selection strategies were deemed appropriate for the job of diagnosing ASDs.

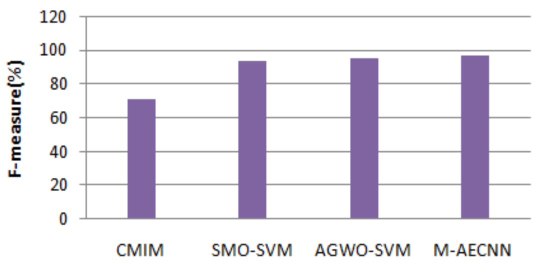

Fig.11. Comparative values of F-measure between suggested and existing methods in classifications of ASDs

As an example, Fig.11 depicts the F-measure comparison between the proposed and current approaches for categorising ASD data. According to both feature-selection and classification approaches, the toddler database variable, which derives from the ASD test, had the greatest correlation with the target class. The above comparison shows that, when compared to the other machine-learning models, the proposed model has the highest accuracy rate measurement in each database. When compared to the teenage databases, the toddler database provides the best f-measure findings. It can be seen from the graph that the suggested M-AECNN model outperforms the current approaches in terms of the f-measure.

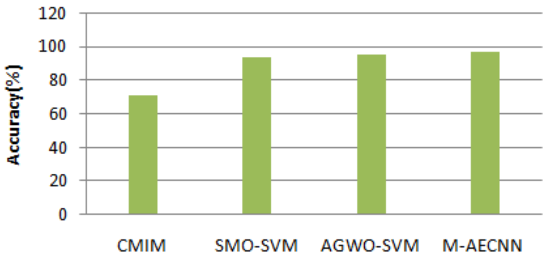

Fig.12. Accuracy comparison results between the proposed and existing method for classifying the ASD data

Figure 12 depicts an accuracy comparison of the proposed and current approaches for categorising ASD data. A successful supervised machine learning model is one that can reliably anticipate the objective and generalise fresh instance predictions. The simulation results show that the proposed M-AECNN model has a high accuracy of 97.47%, compared to the present AGWO-SVM model's accuracy of 95.67%, SMO-SVM model's accuracy of 93.9%, and CMIM model's accuracy of 71.41%. The findings show that, when compared to the current classification approaches, the suggested M-AECNN technique offers good accuracy results.

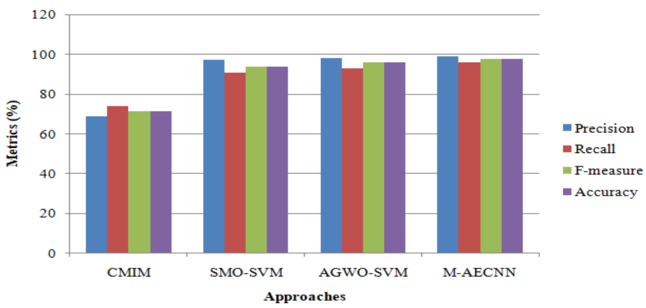

Fig.13. Overall comparison results between the proposed and existing method for classifying the ASD data

The overall comparison between the proposed and current methods for identifying the ASD data is shown in Fig.13. According to the results, the suggested M-AECNN approach performs with excellent accuracy when compared to other classification methods.

4. Conclusion

An sophisticated computer-aided detection method for the diagnosis of ASD was given in this study paper.For the effective identification of ASD, dimensionality reduction and classification methods based on M-AECNN have been developed. This study developed an AGWO to find the most important characteristics and effective classification methods in ASD datasets. To achieve the highest classification accuracy, the suggested AECNN classifier. With the suggested enhancement, the proposed improved DOA effectively addresses the overfitting issue. Obtaining findings that are acceptable and overcoming the difficulty of locating datasets with enough samples in the field of medical imaging are only a few advantages of this paradigm. Researchers and experts will also be able to automatically extract characteristics and categorise photos in a distinctive structure as opposed to the earlier techniques. In this study, ASD diagnosis rates rose in comparison to those obtained using earlier techniques.we might get the conclusion that the suggested design can be regarded as a useful tool for detecting ASD in early children. From a different angle, the additional data pertaining to brain dysfunctions can be examined using the suggested technique.

References

[1] Raj, S., & Masood, S. (2020). Analysis and detection of autism spectrum disorder using machine learning techniques. *Procedia Computer Science*, *167*, 994-1004.

[2] McCarty, P., & Frye, R. E. (2020, October). Early detection and diagnosis of autism spectrum disorder: Why is it so difficult?. In *Seminars in Pediatric Neurology* (Vol. 35, p. 100831). WB Saunders.

[3] Daniels, A. M., Halladay, A. K., Shih, A., Elder, L. M., & Dawson, G. (2014). Approaches to enhancing the early detection of autism spectrum disorders: a systematic review of the literature. *Journal of the American Academy of Child & Adolescent Psychiatry*, *53*(2), 141-152.

[4] Robins, D. L., & Dumont-Mathieu, T. M. (2006). Early screening for autism spectrum disorders: update on the modified checklist for autism in toddlers and other measures. *Journal of Developmental & Behavioral Pediatrics*, *27*(2), S111-S119.

[5] Nadel, S., & Poss, J. E. (2007). Early detection of autism spectrum disorders: screening between 12 and 24 months of age. *Journal of the American Academy of Nurse Practitioners*, *19*(8), 408-417.

[6] Pinto-Martin, J. A., Young, L. M., Mandell, D. S., Poghosyan, L., Giarelli, E., & Levy, S. E. (2008). Screening strategies for autism spectrum disorders in pediatric primary care. *Journal of Developmental & Behavioral Pediatrics*, *29*(5), 345-350.

[7] Nygren, G., Sandberg, E., Gillstedt, F., Ekeroth, G., Arvidsson, T., & Gillberg, C. (2012). A new screening programme for autism in a general population of Swedish toddlers. *Research in developmental disabilities*, *33*(4), 1200-1210.

[8] Toh, T. H., Tan, V. W. Y., Lau, P. S. T., & Kiyu, A. (2018). Accuracy of Modified Checklist for Autism in Toddlers (M-CHAT) in detecting autism and other developmental disorders in community

clinics. *Journal of autism and developmental disorders*, *48*(1), 28-35.

[9] Chlebowski, C., Robins, D. L., Barton, M. L., & Fein, D. (2013). Large-scale use of the modified checklist for autism in low-risk toddlers. *Pediatrics*, *131*(4), e1121-e1127.

[10] Robins, D. L. (2008). Screening for autism spectrum disorders in primary care settings. *Autism*, *12*(5), 537-556.

[11] Factor, R. S., Arriaga, R. I., Morrier, M. J., Mathys, J. B., Dirienzo, M., Miller, C. A., ... & Ousley, O. Y. (2022). Development of an interactive tool of early social responsiveness to track autism risk in infants and toddlers. *Developmental Medicine & Child Neurology*, *64*(3), 323-330.

[12] Sherkatghanad, Z., Akhondzadeh, M., Salari, S., Zomorodi-Moghadam, M., Abdar, M., Acharya, U. R., ... & Salari, V. (2020). Automated detection of autism spectrum disorder using a convolutional neural network. *Frontiers in neuroscience*, *13*, 1325.

[13] Oosterling, I. J., Wensing, M., Swinkels, S. H., Van Der Gaag, R. J., Visser, J. C., Woudenberg, T., ... & Buitelaar, J. K. (2010). Advancing early detection of autism spectrum disorder by applying an integrated two-stage screening approach. *Journal of Child Psychology and Psychiatry*, *51*(3), 250-258.

[14] Khowaja, M., Robins, D. L., & Adamson, L. B. (2018). Utilizing two-tiered screening for early detection of autism spectrum disorder. *Autism*, *22*(7), 881-890.

[15] Guo, X., Dominick, K. C., Minai, A. A., Li, H., Erickson, C. A., & Lu, L. J. (2017). Diagnosing autism spectrum disorder from brain resting-state functional connectivity patterns using a deep neural network with a novel feature selection method. *Frontiers in neuroscience*, *11*, 460.

[16] Abdolzadegan, D., Moattar, M. H., & Ghoshuni, M. (2020). A robust method for early diagnosis of autism spectrum disorder from EEG signals based on feature selection and DBSCAN method. *Biocybernetics and Biomedical Engineering*, *40*(1), 482-493.

[17] Kosmicki, J. A., Sochat, V., Duda, M., & Wall, D. P. (2015). Searching for a minimal set of behaviors for autism detection through feature selection-based machine learning. *Translational psychiatry*, *5*(2), e514-e514.

[18] Alzubi, R., Ramzan, N., & Alzoubi, H. (2017, August). Hybrid feature selection method for autism spectrum disorder SNPs. In *2017 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB)* (pp. 1-7). IEEE.

[19] Hossain, M. D., Kabir, M. A., Anwar, A., & Islam, M. Z. (2021). Detecting autism spectrum disorder using machine learning techniques. *Health Information Science and Systems*, *9*(1), 1-13.

[20] Liu, W., Yu, X., Raj, B., Yi, L., Zou, X., & Li, M. (2015, September). Efficient autism spectrum disorder prediction with eye movement: A machine learning framework. In *2015 International conference on affective computing and intelligent interaction (ACII)* (pp. 649-655). IEEE.

[21] Küpper, C., Stroth, S., Wolff, N., Hauck, F., Kliewer, N., Schad-Hansjosten, T., ... & Roepke, S. (2020). Identifying predictive features of autism spectrum disorders in a clinical sample of adolescents and adults using machine learning. *Scientific reports*, *10*(1), 1-11.

[22] Duda, M., Kosmicki, J. A., & Wall, D. P. (2014). Testing the accuracy of an observation-based classifier for rapid detection of autism risk. *Translational psychiatry*, *4*(8), e424-e424.

[23] Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, *16*, 321-357.

[24] Han, H., Wang, W. Y., & Mao, B. H. (2005, August). Borderline-SMOTE: a new over-sampling method in imbalanced data sets learning. In *International conference on intelligent computing* (pp. 878-887). Springer, Berlin, Heidelberg.

[25] Mirjalili, S., Mirjalili, S. M., & Lewis, A. (2014). Grey wolf optimizer. *Advances in engineering software*, *69*, 46-61.

[26] Al-Tashi, Q., Md Rais, H., Abdulkadir, S. J., Mirjalili, S., & Alhussian, H. (2020). A review of grey wolf optimizer-based feature selection methods for classification. *Evolutionary machine learning techniques*, 273-286.

[27] Masci, J., Meier, U., Cireşan, D., & Schmidhuber, J. (2011, June). Stacked convolutional auto-encoders for hierarchical feature extraction. In *International conference on artificial neural networks* (pp. 52-59). Springer, Berlin, Heidelberg.

[28] Sairam, K., Naren, J., Vithya, G., & Srivathsan, S. (2019). Computer aided system for autism spectrum disorder using deep learning methods. *Int. J. Psychosoc. Rehabil*, *23*(01).

[29] Mafarja, M., Heidari, A. A., Faris, H., Mirjalili, S., & Aljarah, I. (2020). Dragonfly algorithm: theory, literature review, and application in feature selection. *Nature-inspired optimizers*, 47-67.

[30] Rahman, C. M., & Rashid, T. A. (2019). Dragonfly algorithm and its applications in applied science survey. *Computational Intelligence and Neuroscience*, *2019*.

[31] Kumar, D. ., & Sonia, S. (2023). Resources Efficient Dynamic Clustering Algorithm for Flying Ad-Hoc Network. International Journal on Recent and Innovation Trends in Computing and

Communication, 11(2s), 106–117. https://doi.org/10.17762/ijritcc.v11i2s.6034

[32] Juan Lopez, Machine Learning-based Recommender Systems for E-commerce , Machine Learning Applications Conference Proceedings, Vol 2 2022.

[33] Pandey, J.K., Ahamad, S., Veeraiah, V., Adil, N., Dhabliya, D., Koujalagi, A., Gupta, A. Impact of call drop ratio over 5G network (2023) Innovative Smart Materials Used in Wireless Communication Technology, pp. 201-224.