Annals of Emerging Technologies in Computing (AETiC)
Vol. 10, No. 1, 2026

Research Article

Enhancing Intrusion Detection System Performance Using a Modified Grey Wolf Optimizer

Abdullah Al Mosuli[1], Mosleh Abualhaj[1,*], Ahmad Abu-Shareha[1], Mohamed Yousif[2] and Mohammad Daoud[3]

[1]Al Ahliyya Amman University, Jordan
moslehabualhaj@yahoo.com; m.abualhaj@ammanu.edu.jo; a.abushareha@ammanu.edu.jo
[2]Cardiff Metropolitan University, UK
MYousif@cardiffmet.ac.uk
[3]Al Ain University, United Arab Emirates
mohammad.daoud@aau.ac.ae
*Correspondence: m.abualhaj@ammanu.edu

Received: 10 November 2025; Accepted: 22 December 2025; Published: 1 January 2026

Abstract: Cybersecurity is one of the main worries of organizations, businesses, and even individuals. The problems facing cybersecurity are increasing on daily basis as a result of the increased reliance on electronic services and technologies and the associated increase in the number of cyberattacks. The prevention of cyberattacks has become a serious challenge due to the vast increase in cybersecurity threats. Intrusion Detection System (IDS) acts as one of the first line of defence against cyberattacks, protecting computer networks and users' data. However, the efficiency and effectiveness of IDS can be challenged by the enormous data monitored by the IDS, and the irrelevant features in the data. This study presents a Machine Learning (ML) model for intrusion detection and aims to enhance the model by employing the proposed Modified-Grey Wolf Optimizer (GWO) for feature selection. A new mutation function and an effective initialization method are introduced to the GWO, enhancing its exploration of the solution space and reducing convergence time. The proposed modified-GWO is then applied to the NSL-KDD dataset for feature selection, identifying the most relevant features for intrusion detection. The ML model will be tested using various ML classifiers. These classifiers are XGBoost, RF, HGB, and MNB. The proposed model achieved remarkable results with the XGBoost classifier reaching an accuracy of 99.52%, a precision of 99.47%, and a recall of 99.46%.

Keywords: ***Feature Selection; Grey Wolf Optimizer;*** **Machine Learning*****; NSL-KDD Dataset***

1. Introduction

The rapid advancement in information technology has produced many digital services and applications to make life easier for organizations, businesses, and individuals [1]. However, the increased reliance on these services opened opportunities for cyberattackers to exploit them. Cyberattacks (Intrusions) are unauthorized access or attempt to access a computer or a network. Cyberattackers (Intruders) can steal sensitive information, disrupt network operations, or launch further attacks from the victim's computer [2, 3]. These attacks may result in reputational damage and financial loss for both individuals and companies. It is estimated that a total of $3 trillion was lost due to cyberattacks in 2015, meanwhile, this figure is projected to increase to $10.5 trillion annually by 2025[1]. Cyberattackers are constantly finding new ways to exploit vulnerabilities in computer systems and networks [4, 5].

[1] https://www.globenewswire.com/news-release/2020/11/18/2129432/0/en/Cybercrime-To-Cost-The-World-10-5-Trillion-Annually-By-2025.html

Abdullah Al Mosuli, Mosleh Abualhaj, Ahmad Abu-Shareha, Mohamed Yousif and Mohammad Daoud, "Enhancing Intrusion Detection System Performance Using a Modified Grey Wolf Optimizer", *Annals of Emerging Technologies in Computing (AETiC)*, Print ISSN: 2516-0281, Online ISSN: 2516-029X, pp. 21-44, Vol. 10, No. 1, 1 January 2026, Published by International Association for Educators and Researchers (IAER), DOI: 10.33166/AETiC.2026.01.002, Available: http://aetic.theiaer.org/archive/v10/v10n1/p2.html.

The prevention of intrusions has become a serious challenge due to the vast increase in cybersecurity threats [6]. So many defense mechanisms have been employed, including firewalls, cryptography, anti-malware software, and Intrusion Detection System (IDS) [7]. IDS is software or hardware that serves as a safeguard to computer networks. The IDS's main role is to monitor network data and host activities for indicators of compromise (suspicious activities) and then produce warnings and reports about them. IDS can also block these suspicious activities if configured by the administrator [8-10]. Fig. 1 shows a simple representation of the IDS functionality in intrusion detection.

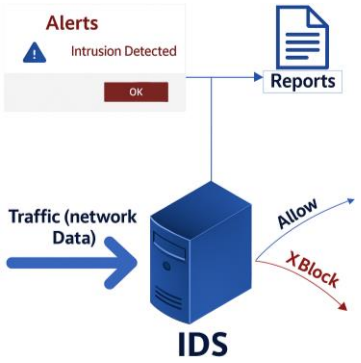

Figure 1. Simple representation of the IDS functionality

Various challenges have an impact on the IDS's effectiveness and resources; these challenges can be summarized as follows: 1) the continuous changing of the intruder's techniques, 2) the large quantity of data in the network activity that should be handled by the IDS, and 3) the commonly duplicated and irrelevant features contained in the analyzed data [11-13]. Accordingly, rather than using simple IDS to detect intrusions, an ML-based IDS is required to detect the intruder's suspicious activity [14].

ML-based IDS is implemented commonly as a classification task, in which a supervised ML model is trained by a set of samples with their true labels (e.g., intrusion vs. normal flows). Then, the trained model is used with samples of unknown labels with the purpose of assigning the correct label to those samples [15, 16]. One of the most influential steps in ML-IDS is the ability to handle large quantity of data using feature selection, which eliminates irrelevant features that in turns increase the IDS's effectiveness and save computational resources [17, 18]. Feature selection is an important step in ML as it involves identifying and selecting the most crucial features out of a larger set of features. Feature selection plays a crucial role in the performance of an IDS when used with ML. Precisely selecting the relevant and important features will considerably increase the IDS's performance. This in turn directly impacts the efficiency of the ML model [19, 20].

Recently, various metaheuristic optimization algorithms have been used for feature selection. Metaheuristic algorithms are used with complex problems to find the optimal solution in a large space of possible solutions. The metaheuristic algorithms demonstrated high efficiency in resolving many problems, including scientific and engineering problems. The advantage of these algorithms is the ability to find nearly-optimal solutions in a relatively short time [21-23]. Therefore, these algorithms can be applied for feature selection in case the number of features is high. Accordingly, metaheuristic algorithms are widely used in feature selection because they can handle high-dimensional datasets and can be computationally efficient. These algorithms replaced the weak filter-based feature selection technique, which selects the features with the assumption that these features are independent of each other [24- 26].

GWO is a metaheuristic algorithm that has demonstrated excellent performance, and it is frequently used to enhance ongoing applications, including cluster analysis, engineering problems, and neural network training [27, 28]. GWO has been adopted by many researchers in feature selection operations as it can handle high-dimensional datasets, which makes it appropriate for IDS systems that often deal with large data to detect intrusions [29]. In this work, a modified version of GWO that suits the IDS will be utilized to increase the IDS model performance. This study will evaluate the proposed method performance using the NSL KDD dataset. The dataset is considered a suitable benchmark for testing intrusion detection techniques because it contains various network intrusion scenarios, and its size is appropriate for conducting experiments and evaluating the model's performance using multiple ML algorithms.

2. Background

This section outlines the key concepts underlying the proposed intrusion detection framework. It reviews ML-based detection methods in IDS, focusing on classifiers such as Random Forest (RF), Histogram-based gradient Boosting (HGB), XGBoost, and Naive Bayes (NB). It also introduces the GWO algorithm for feature selection and provides an overview of the dataset used to evaluate the model's performance.

2.1. ML Detection in IDS

Incorporating ML in IDS provides a powerful technique for detection, using ML algorithms to analyze network traffic and learn to recognize patterns that may indicate an intrusion. This is beneficial in detecting new attacks that do not match any known signature [30].

Supervised Learning is a type of ML in which the algorithm is trained on a labeled datasetand then used to classify new data. For example, in the case of IDS, a dataset may consist of network traffic data, where each piece of data is labeled as either "normal" or "attack". This labeled dataset serves as a training set for the ML model using an ML algorithm. The algorithm learns from this data by finding patterns and correlations. Once the model is trained, it can be used to classify new and unseen data as either normal or attack based [31, 32]. Many classification algorithms can be used to train IDSs such as RF, HGB, XGBoost, and NB. The following sections describe each one briefly.

2.1.1. RF

RF is an ensemble learning method that works by constructing multiple Decision Trees (DTs) during training. Each tree in the forest generates an output when a new input is introduced into the system. The outputs are then collected, and the most frequently seen output is selected as the final result. RF can handle missing values without reducing accuracy. It is known for its accurate results in classification. Additionally, it is capable of handling large, multidimensional data sets. Fig. 2 outlines the main components for the RF classifier [33].

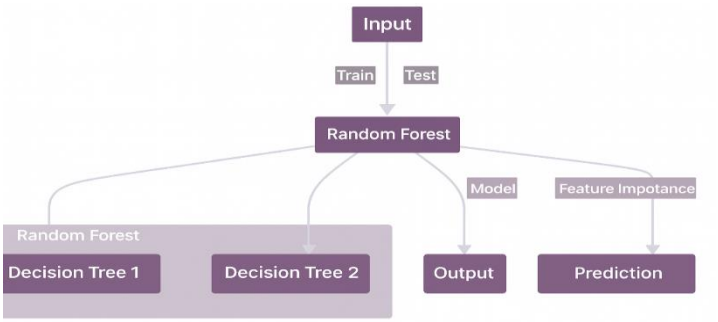

Figure 2. Main components of RF classifier

2.1.2. HGB

HGB is an ML algorithm that builds upon the principles of Gradient Boosting with DT. It's specifically designed to be faster and more scalable by utilizing Histogram-based techniques. Gradient Boosting is a powerful ensemble learning algorithm known for its effectiveness in reducing bias and variance in supervised learning. It builds the model in a stage-wise way, and it generalizes them by allowing optimization of an arbitrary differentiable loss function. Fig. 3 outlines the main components for the GB classifier [34].

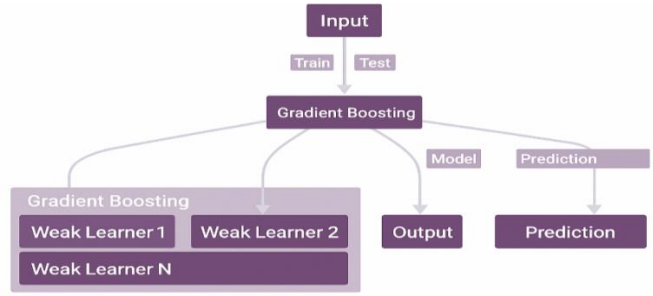

Figure 3. Main component of GB classifier

2.1.3. XGBoost

Extreme Gradient Boosting, also known as XGBoost, is a variance of Gradient Boosting that adds plenty of performance and speed-enhancing features. It uses a library of Gradient-Boosted DTs to make up the bulk of it. It adds parallel processing, which Gradient Boosting noticeably lacks, making it faster for training and prediction and more resource-efficient in terms of memory usage. Fig. 4 outlines the main components for the XGBoost classifier [33, 35].

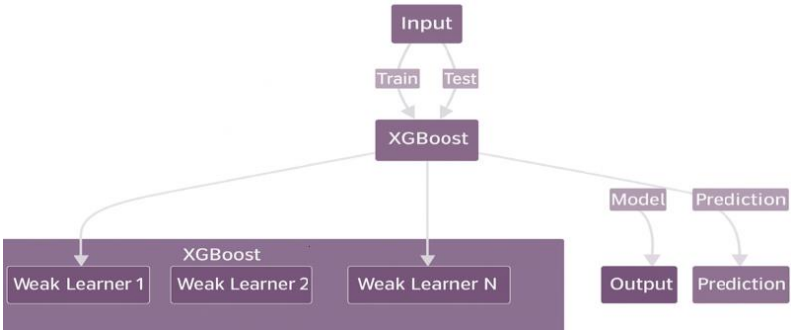

Figure 4. Main component of XGBoost classifier

2.1.4. NB

NB classifiers are a family of simple "probabilistic classifiers" based on applying Bayes' theorem with strong independence assumptions between the features. In simple terms, a NB classifier assumes that the presence of a particular feature in a class is unrelated to the presence of any other feature. Multinomial Naive Bayes (MNB) is used for discrete features and models data with a multinomial distribution. Fig. 5 outlines the main components for the NB classifier [36, 37].

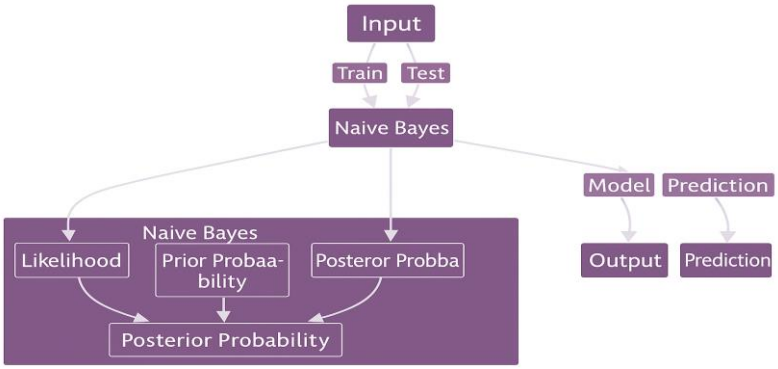

Figure 5. Main components of NB classifier

2.2. GWO Algorithm

The GWO algorithm works by simulating the social structure and hunting behavior of grey wolves in nature. The method begins by randomly distributing a population of search agents, simulating a pack of grey wolves, across the search space, which represents the hunting field. The search agents are then rated based on their level of fitness, with the fittest solution designated as the alpha wolf, the next-best choice as the beta wolf, and the third as the delta wolf. The remaining search agents are classified as omega wolves. The locations of the search agents are adjusted in each iteration based on the locations of the alpha, beta, and delta wolves. This is accomplished by imitating the encircling behavior of grey wolves, who encircle and attack their victims. While examining the search space, the search agents move closer to the prey, which is portrayed by the greatest solution discovered so far. Several parameters are used by the algorithm to control the balance of exploration and exploitation. For example, to manage the exploration/exploitation trade-off, a parameter 'a' is employed with its value decreasing linearly from 2 to 0 over the duration of iterations. This enables the algorithm to begin with more exploration and progressively move to exploitation as it approaches the global optimum. The GWO keeps updating the locations of the search agents until a requirement is fulfilled, such as attaining a maximum number of cycles or reaching a desirable degree of convergence. The best answer produced by the algorithm is presented as the final outcome at that point [38-40]). Algorithm 1 shows the pseudo code for the GWO algorithm.

Algorithm 1. Pseudo code for the GWO algorithm

1. **Random initialization** for the group of search agents (grey wolves).
   1.1 Initialize *a, A, C.*
2. **Evaluate** each feature subset **fitness**.
3. **While** (iteration < T):
   3.1 **Update** the alpha, beta, and delta wolves' positions using Eq. (2.1).
   3.2 **For each** search agent, adjust all the wolves' positions based on the positions of the alpha, beta, and delta.
   3.3 **Update** *a, A, C.*
   3.4 **Determine** the fitness for all wolves.
   3.5 **Iteration = iteration + 1.**
4. **Return** the best solution found by the algorithm.

The GWO Algorithm works as follows:

Step 1: Randomly create a population of search agents (grey wolves), each representing a potential solution to the optimization issue.

Step 1.1: Initialize a, A and C. where the vectors A is calculated using Equation (1), C is computed using Equation (2), 'a' is a parameter that decreases linearly from 2 to 0 over the course of iterations, and r1 and r2 are random vectors in [0, 1].

$$A = 2a * r1 - \quad (1)$$

$$C = 2 * r2 \quad (2)$$

Step 2: Compute the fitness value of each search agent and order them accordingly. The best search agent is known as the alpha wolf (α) Xα, the second best is known as the beta wolf (β) Xβ, and the third best is known as the delta wolf (δ) Xδ.

Step 3: Repeat Step 3 until a stopping requirement, such as reaching the maximum number of iterations is reached or acquiring the desirable degree of convergence.

Step 3.1: Update the positions of alpha Dα, beta Dβ, and delta Dδ using Equation (3):

$$\begin{cases} D\alpha = |C1 * X\alpha - X|, X1 = X\alpha - A1 * D\alpha \\ D\beta = |C2 * X\beta - X|, X2 = X\beta - A2 * D\beta \\ D\delta = |C3 * X\delta - X|, X3 = X\delta - A3 * D\delta \end{cases} \quad (3)$$

$$X(t+1) = \frac{(X1+X2+X3)}{3}$$

Which t is the current iteration, A and C are coefficient vectors, Xα, Xβ, and Xδ are the alpha, beta, and delta wolf position vectors, accordingly, and X is the location vector of a search agent.

Step 3.2: Update the positions for each search agent based on the current positions of the alpha, beta, and delta wolves.

Step 3.3: Update a, A and C.

Where A is a random value in the interval [-2a, 2a], and C is a random value assigned each iteration between [0, 2].

Step 3.4: Compute the fitness value of each search agent and order them accordingly.

Step 3.5: Increase the iteration number.

Step 4: When the stopping criteria is met return the best solution.

2.3. Dataset

The NSL-KDD dataset was used in this study. It contains diverse network attacks which make it suitable for building a solid intrusion detection model. The NSL KDD is an improved version of the KDD Cup 99 dataset. The NSL KDD dataset records were simulated in a military network environment, and it contains around 150,620 records, with four main attack categories. These categories are DoS (Denial of Service), Probe, U2R (User to Root), and R2L (Remote to Local). In total, the dataset contains 39 distinct attack types and a "normal" class, summing up to 40 classes. NSL KDD contains 41 features, each representing distinct attributes of network connections. An explanation of each feature, as well as its data type, is provided in Table 1. The NSL KDD was selected for this study due to the dataset's diversity of attack types and network traffic scenarios; it offers an extensive environment for developing and testing IDS models. The use of this dataset enables a thorough assessment of how well the suggested model performs in identifying network intrusions [41, 42].

Table 1. NSL-KDD Feature Description

Number	Feature Name	Data Type	Feature Description
1	duration	Numeric	Length of the connection
2	protocol type	Nominal	Connection protocol
3	service	Nominal	Destination service
4	flag	Nominal	Status flag of the connection
5	src bytes	Numeric	Bytes sent from source to destination
6	dst bytes	Numeric	Bytes sent from destination to source
7	land	Nominal	1 if is from/to the same host/port; 0 otherwise
8	wrong fragment	Numeric	Number of wrong fragments
9	urgent	Numeric	Number of urgent packets
10	hot	Numeric	Number of hot indicators
11	num failed logins	Numeric	Number of failed login in attempts
12	logged in	Nominal	1 if successfully logged in; 0 otherwise
13	num compromised	Numeric	Number of compromised conditions
14	root shell	Numeric	1 if root shell is obtained; 0 otherwise
15	su attempted	Numeric	1 if "su root" command attempted; 0 otherwise
16	num root	Numeric	Number of root accesses
17	num file creations	Numeric	Number of file creation operations
18	num shells	Numeric	Number of shell prompts
19	num access files	Numeric	Number of operations on access control files
20	num outbound cmds	Numeric	Number of outbound commands in an ftp session
21	is host login	Nominal	1 if the login belongs to the hot list; 0 otherwise
22	is guest login	Nominal	1 if the login is a guest login; 0 otherwise
23	count	Numeric	Number of conn. to the same host as the current conn. in the past two sec.
24	srv count	Numeric	Number of conn. to the same service as the current conn. in the past two sec.
25	serror rate	Numeric	% of conn. that have "SYN" errors (same-host conn.)
26	srv serror rate	Numeric	% of conn. that have "SYN" errors (same-service conn.)
27	rerror rate	Numeric	% of conn. that have "REJ" errors (same-host conn.)
28	srv rerror rate	Numeric	% of conn. that have "REJ" errors (same-service conn.)
29	same srv rate	Numeric	% of conn. to the same service (same service conn.)
30	diff srv rate	Numeric	% of conn. to different services
31	srv diff host rate	Numeric	% of conn. to different hosts (same-service conn.)
32	dst host count	Numeric	% Count of conn. having the same destination host
33	dst host srv count	Numeric	% Count of conn. having the same destination host and using the same service
34	dst host same srv rate	Numeric	% of conn. having the same destination host and using the same service
35	dst host diff srv rate	Numeric	% of different services on the current host
36	dst host same src port rate	Numeric	% of conn. to the current host having the same port
37	dst host srv diff host rate	Numeric	% of conn. to the same service coming from different hosts
38	dst host serror rate	Numeric	% of conn. to the current host that have an SO error
39	dst host srv serror rate	Numeric	% of conn. to the current host and specified service that have an SO error
40	dst host rerror rate	Numeric	% of conn. to the current host that have an RST error
41	dst host srv rerror rate	Numeric	% of conn. to the current host and specified service that have an RST error

3. Related Works

In Alzubi *et al.* [43], proposed an improved IDS through the usage of a modified binary GWO (MBGWO) algorithm to choose significant features for intrusion detection (feature selection). They modified the GWO by including the omega wolf in the decision-making process and using a random probability distribution crossover strategy. The algorithm was able to significantly reduce the number of features from 41 to 14, while still achieving an accuracy of up to 99.22% when using the NSL-KDD dataset. Support Vector Machine (SVM) with radial basis function kernel was employed as a classifier.

Meanwhile, Yerriswamy & Murtugudde [44] proposed a modified GWO with a Genetic Algorithm (GA) crossover and sigmoid function for feature selection for IDS. Using the NSL-KDD datasets, the proposed GB-EGWO outperforms other algorithms like GWO, MGWO, and MBGWO, achieving an average accuracy of 98.62% with 14 selected features.

As for Shakya [45], on the same NSL-KDD dataset, the study discussed an improved IDS for Wireless Sensor Network (WSN) that utilizes SVM classifier with the ML- MLGWO, by adding more wolves and

including a multi-objective fitness function, the algorithm was modified by increasing the number of wolves to 14 achieving an accuracy of 97.00%.

On the other hand, the study done by Safaldin *et al.* [46] proposed using the SVM classifier and a modified GWO to improve the IDS in WSNs. The algorithm used binary encoding, binarization, stochastic crossover, and 7 wolves. The study showed that the suggested methods outperform Particle Swarm Optimization (PSO)-IDS and GWO-IDS using the NSL KDD'99 dataset, with the best performance coming from GWOSVM-IDS with an accuracy of 96%.

Another study on the NSL-KDD was done by Almazini & Ku-Mahamud [47], the algorithm was modified with binary representation, stochastic crossover operation, and a sigmoidal function for position updates. The enhanced binary GWO EBGWO uses adaptive parameter control with search process indicators. The SVM classifier usesa Radial Basis Function kernel and achieves a classification accuracy of 87.46%.

On the other hand, Madhavi & Nethravathi [48] proposed a model for intrusion detection using the GWO for feature selection and Gradient Boosting Decision Tree (GBDT) for classification. The dataset used for evaluation was the KDD99 dataset, and the researchers also discussed how to create attack rules by using a KDD99 dataset to look for anomalies in network audit data.

As for Chatterjee *et al.* [49] a Multi-Stage IDS was introduced that incorporates the GWO for feature selection. The proposed framework employs a Stacked Autoencoder to classify incoming data packets as either benign or malicious. The GWO algorithm is employed to identify and extract the most relevant features from network packets, after which each packet is classified as either malicious or benign. These attributes are then fed into an RF classifier to determine if the attack is present in the existing knowledge base. If the attack is detected, the LightGBM classifier is used to identify the specific type of attack. If the attack is not found in the knowledge base, it is classified as a zero-day attack. For the evaluation of their proposed framework, two publicly available datasets were used, namely UNSW-NB15 and CIC-IDS-2017. The results of the evaluation showed that the proposed framework achieved an accuracy of 90.94% and 99.67% on the respective datasets.

Existing studies on GWO-based intrusion detection systems indicate that differences in reported performance among modified variants are mainly influenced by how each approach adjusts the exploration–exploitation balance and binary search behavior of the optimizer. The literature consistently reports that baseline GWO is vulnerable to premature convergence and stagnation due to rapid population clustering, and in some cases exhibits slow convergence in later iterations [43–47]. These limitations have motivated several enhancement strategies.

Crossover-based variants, such as MBGWO by Alzubi *et al.* [43] and GB-EGWO by Yerriswamy & Murtugudde [44], strengthen exploitation by recombining promising feature subsets; however, once population diversity decreases, crossover largely recombines similar binary patterns and becomes ineffective at escaping stagnation. Binary encoding and transfer-function-based approaches, as adopted by Safaldin *et al.* [46] and Almazini & Ku-Mahamud [47], stabilize discretization but may cause early fixation of feature selections, limiting late-stage exploration. Increasing the number of wolves or using multi-objective fitness functions, as in Shakya's work [45], improves early exploration but does not explicitly restore diversity after convergence. Other studies, such as Madhavi & Nethravathi [48], further show that performance gains may depend on classifier choice rather than optimizer design alone.

From a practical perspective, most GWO-based IDS frameworks adopt single-stage wrapper architectures, where computational cost scales with population size and fitness evaluations [43–48]. In contrast, the multi-stage IDS proposed by Chatterjee *et al.* [49] achieves high accuracy through cascading learning stages but incurs significantly higher computational and deployment complexity. Overall, existing GWO modifications improve convergence guidance or discretization but remain limited in late-stage diversity recovery, motivating the proposed bit-flip mutation, which directly perturbs binary feature selections while preserving a computationally efficient single-stage framework.

Although bit-flip mutation is a standard operator in Genetic Algorithms, its use here is not intended as generic GA–GWO hybridization. Rather, the contribution lies in the targeted integration of mutation within the GWO search process to mitigate late-stage diversity loss in binary feature selection, while preserving the original leadership-driven dynamics of GWO. This focused design enables diversity

recovery with minimal computational overhead and maintains the efficiency of a single-stage wrapper framework.

Table 2 further contextualizes these studies by summarizing their primary optimization mechanisms and associated computational complexity, highlighting the absence of explicit late-stage diversity recovery strategies.

Table 2. Summary of the related works on IDS using the GWO algorithm

Reference	Feature Selection Method	Primary Mechanism	Dataset	Classifier	Accuracy	Cost / Complexity Note
Safaldin *et al.*, 2021	GWOSVM	Binary encoding + binarization + stochastic crossover	NSL KDD	SVM	96%	Single-stage wrapper; cost ∝ population size and fitness evaluations
Alzubi *et al.*, 2020	MBGWO	Omega wolf participation + probabilistic crossover	NSL-KDD	SVM	99.22%	Single-stage wrapper; moderate cost due to crossover and added leadership
Yerriswamy & Murtugudde, 2021	GB-EGWO	GA crossover + sigmoid transfer function	NSL KDD	SVM	98.62%	Single-stage wrapper; added cost from crossover and transfer-function mapping
Shakya, 2021	MLGWO	Increased wolf population + multi-objective fitness	NSL-KDD	SVM	97.00%	Single-stage wrapper; higher cost due to increased population size
Almazini & Ku-Mahamud, 2021	EBGWO	Binary representation + stochastic crossover + adaptive control	NSL KDD	SVM with RBF	87.46%.	Single-stage wrapper; additional overhead from adaptive parameter control
Madhavi & Nethravathi, 2022	NA	Standard GWO feature selection + strong classifier	KDD99 incursion dataset	GBDT	92.17%	Single-stage wrapper; classifier-driven cost dominates
Chatterjee *et al.*, 2023	NA	Multi-stage stacking (FS + SAE + RF + LightGBM)	UNSW-NB15, CIC-IDS-2017	RF, LightGBM	90.94%, 99.67%	Multi-stage pipeline; high computational and deployment complexity

4. Proposed Methodology

The proposed method will incorporate GWO with multiple classification algorithms using the NSL KDD dataset. NSL KDD contains a total of 41 features, and it may contain irrelevant features that could impact the model performance negatively. Therefore, feature selection is necessary for identifying the most relevant features from the dataset and to use these features as inputs to various classification algorithms. Fig. 6 shows the main stages for the proposed ML model.

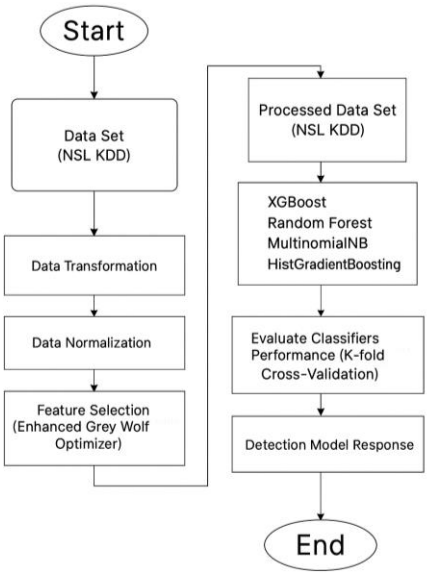

Figure 6. The proposed ML Model

4.1. Data Transformation

Data transformation played an important role in preparing the NSL-KDD for the ML model. This dataset contains many categorical features such as protocol type, service, flag and output. These features may hinder the performance of the ML algorithms due to the fact that many ML algorithms tend to work better with numerical data. Therefore, the numerical representation of these categorical features becomes critical to the successful implementation of ML models. This study used the transformation methodology known as "label encoding" which converts categories into a set of integers; each distinct category is assigned a distinct integer. Label encoding was chosen due to its simplicity to apply and computationally efficient. In addition, unlike other encoding methods such as one-hot encoding, it does not increase the dimensionality of the dataset making, which makes it a suitable solution for the NSL-KDD. Table 3 shows label encoder representation of attack categories [41, 50, 51].

Table 3. Label Encoder Representation of Attack Categories

Original Values	Label Encoded Values	Original Values	Label Encoded Values	Original Values	Label Encoded Values
apache2	0	neptune	14	smurf	27
back	1	nmap	15	snmpgetattack	28
buffer_overflow	2	normal	16	snmpguess	29
ftp_write	3	perl	17	spy	30
guess_passwd	4	phf	18	sqlattack	31
httptunnel	5	pod	19	teardrop	32
imap	6	portsweep	20	udpstorm	33
ipsweep	7	processtable	21	warezclient	34
land	8	ps	22	warezmaster	35
loadmodule	9	rootkit	23	worm	36
mailbomb	10	saint	24	xlock	37
mscan	11	satan	25	xsnoop	38
multihop	12	sendmail	26	xterm	39
named	13				

4.2. Data Normalization

Data normalization is a necessary step in preparing the NSL-KDD for the model. Normalization is a technique used to change the scale of a variable to a standard range. Min-Max Scaler is one of the scaling techniques that transforms variables to a specific range between 0 and 1. This technique is useful for the ML model due to the fact that most classification algorithms assume that the input variables are on the same scale and may not perform well if the scales are vastly different. Additionally, the GWO algorithm assumes that all features have the same scale, so Min-Max Scaler was used for this model to ensure that all features have the same scale. The Min-Max Scaling is performed using Equation (4) [52, 53].

$$Yscaled = \frac{y - Y_{Min}}{Y_{Max} - Y_{Min}} \quad (4)$$

Where y_{Scaled} is the result for the Min-Max Scaling, y is the original value, y_{Min} is the is the minimum value in the column, and y_{Max} is the maximum value in the column.

Min-Max scaler was used on many features including (duration, src_bytes, dst_bytes, wrong_fragment, urgent, hot, num_failed_logins, num_compromised, su_attempted, num_root, num_file_creations, num_shells, num_access_files, Count, srv_count, dst_host_count and dst_host_srv_count).

4.3. Feature Selection

Feature selection is a method of selecting the most relevant features from a bigger set of features to be utilized in an ML model. Feature selection aims to improve the performance of this model by selecting only the most relevant and informative features. GWO optimizer has demonstrated its effectiveness in feature selection, especially when applied to high-dimensional datasets which make it appropriate for IDS [54-56]. This study proposed an improved version of GWO for feature selection. The GWO was improved using a mutation function, which improves the exploration of the solution space by GWO. Additionally, the study used an effective initialization method for the GWO by employing a knowledge-guided initialization in which the Flag and Service features were included in the initial population only, while allowing all features

to be selected or discarded during optimization, due to their importance in determining an attack. This ensures a faster convergence time for the algorithm and reduces computational complexity. Feature selection is a very necessary step in this model as it aims to increase the detection accuracy of the model.

4.4. Classification

The classification stage is a crucial part of the proposed model; it is based on the features that were chosen to be the most relevant features during the feature selection process. Classification is important because it determines whether monitored network activity is normal or an intrusion. Classification usually splits data into two stages training and testing. In first stage the training is done for each classifier using the training data. And in the second stage the classifiers effectiveness is assessed in predicting the class of new data (testing data) [57-59].

This research utilized multiple classifiers (RF, MNB, HGB and XGBoost) in the proposed model to take advantage of the pros of various ML algorithms. Every classifier has a unique method for gathering knowledge from the data and making predictions. Some classifiers might work better with particular types of problems or certain types of data. Utilizing a variety of classifiers increases the likelihood of discovering a model that performs well on the specific task [57-59].

5. The Proposed Modified-GWO

ML-based IDS handles an enormous amount of data such as network data and system logs for intrusion detection. These data may contain irrelevant features that may affect the performance of the detection negatively. Feature selection filters out the irrelevant features from the data, allowing the model to operate only on the most relevant features, improving the detection performance and reducing computational demand. The GWO has proven its capability in feature selection tasks [61, 62, 63]. However, a common issue with the algorithm is that it tends to get stuck in local optima, which is a common problem in ML. Local optima is where the algorithm reaches a solution in the explored space and label it as the best solution. However, exploring the unexplored areas of the solution space may yield a better solution [18, 24]. This issue necessitated the modification of the GWO to enhance its ability in exploring the solution space by introducing a mutation function and using an effective initialization methodology. A mutation function and effective mutation function can help to prevent being trapped in local optimum and increase the exploration of the solution space. Additionally, the use of an effective initialization method can make the convergence speed faster, which in turn may reduce computational complexity. Fig. 7 shows the original GWO flow diagram and outlines the steps for the algorithm [18, 24, 63]. In the proposed modified-GWO, a new mutation function and an effective initialization technique were implemented for feature selection as shown in Fig. 8.

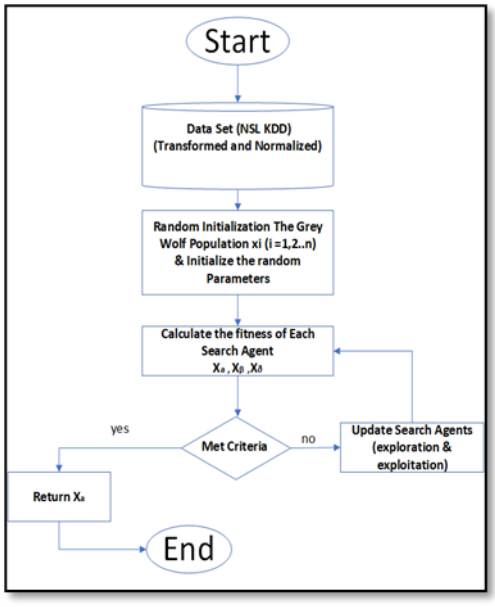

Figure 7. Original GWO

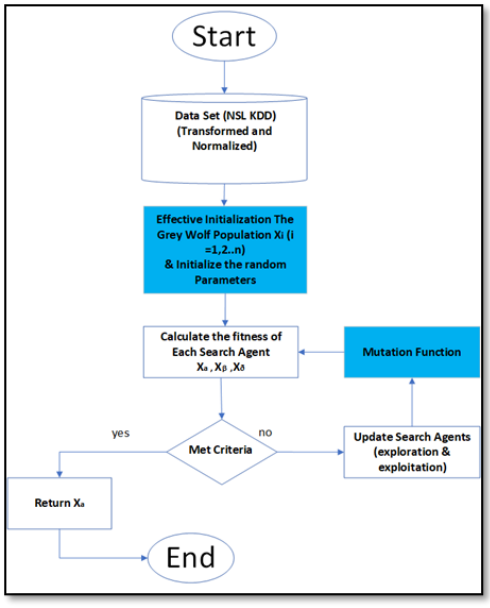

Figure 8. Modified GWO

5.1. The Mutation Function

An effective mutation function increases the exploration of the solution space for finding a better feature subset and could prevent the GWO from getting stuck in local optima. The proposed modified-GWO uses the bit-flip mutation function, a popular mutation method used in binary genetic algorithms for feature selection [64]. This mutation function's goal is to increase the exploration of the solution space, which can help keep the algorithm from getting stuck in local optima. Each bit in the binary representation of a solution (in this case, a feature subset) has a specific probability of being flipped in the bit-flip mutation. As a result, a bit in the solution that is currently set to '0' could potentially become '1' and vice versa. The probability that a bit will be flipped is called mutation rate which is a parameter that can be changed depending on the particular problem and dataset. In this method it was set to 0.2. The aim of using the mutation was to explore a wider range of potential solutions in the feature space and increasing the likelihood of discovering an ideal or nearly ideal feature subset for the intrusion detection system.

5.2. The Effective Initialization Technique

The knowledge-guided initialization with random and Latin Hypercube Sampling (LHS) are the initialization techniques used in the proposed modified-GWO. This strategy is intended to guarantee a diverse and representative initial population, which is essential for any metaheuristic algorithm to succeed [65, 66]. Knowledge-guided initialization means including a feature subset or an important feature in the initial population only, which guides the early search process without constraining the optimization. This will aid in pointing the search process in the direction of better search space solutions. Also, the statistical technique LHS was used to create a subset of features. LHS guarantees that solution space is evenly and thoroughly sampled and aids in producing a more representative and diverse initial population during GWO initialization, improving the algorithm's capacity for exploration. Combining these techniques for initialization will increase the efficiency and effectiveness of the GWO for feature selection in the intrusion detection system by producing an initial population that is diverse and potentially close to the optimal solution.

On the other hand, there are features in the data that contribute mostly to the identification of malicious activities such as Flag and Service. Consequently, including these features in the initial population only, while allowing them to be selected or discarded during optimization, will contribute to the detection performance and lead to faster convergence time.

5.2.1. The Flag (Status Flag of the Connection) Feature

The flag feature in the dataset represents the status of the connection, such as whether the connection was established successfully or if it was rejected. Malicious activities such as scans, DoS attacks, or unauthorized access attempts often manifest themselves through unusual connection statuses. This feature can be crucial in detecting abnormal patterns, as malicious activities often exhibit anomalous connection statuses [67].

By analyzing the "flag" feature, it is possible to identify these anomalies and flag suspicious activities. For example, rapid changes in connection statuses or patterns that deviate from the norm can trigger alerts, allowing for quick intervention. Fig. 9 shows the TCP connection flags and their placement in the header.

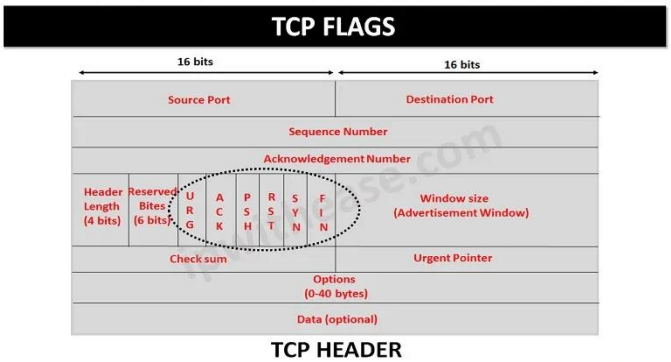

Figure 8. TCP header and flags

In the NSL KDD dataset the "S0" flag values indicate that a connection attempt was initiated (a SYN packet was sent), but no response was received from the other side (no SYN-ACK packet was received).

5.2.2. The Service (Destination Service) Feature

The "service" feature identifies the network service on the destination, such as HTTP, FTP, SMTP, etc. It categorizes the type of service that the connection is attempting to access or utilize. Analyzing the service feature can reveal unusual patterns related to specific services, which could be indicative of an attack on that particular service. The service feature importance lies in identifying an attack that is targeting a certain service. For example, an unusually high number of requests to the FTP service might indicate an attempt to brute-force FTP credentials [68].

The combination for these features could be useful in detecting instances of attacks. For example, a flag value of S0 indicates that a connection attempt was initiated but no response was received from the destination host, while the service value of FTP denotes that the attempted connection targeted the FTP service. This combination may suggest a possible scanning activity or an attempted attack on the FTP service. Additionally, a series of "S0" flags (connection attempts without response) followed by "S2" (connection established and closed) on the FTP service might indicate a brute force attack where an attacker is attempting to guess the password. Therefore, both Flag and Service features could be useful for the identification of malicious behavior, and in the feature selection process. Including these features in the initial population through a knowledge-guided initialization could lead to faster convergence times and better detection performance.

6. Implementation and Result

This section describes the experimental setup and findings of the proposed IDS framework. It details the implementation environment, operations, and evaluation metrics used to assess performance. The results include analyses of the proposed modified-GWO model, its comparison with the original GWO, and benchmarking against other existing models.

6.1. Implementation Environment

The research was conducted in a Windows environment, using Visual Studio Code IDE. Visual Studio Code offers a complete set of tools for creating, testing, and debugging code. It also offers easy integration with Anaconda. Anaconda is a Python distribution that comes with a number of well-known data science and ML libraries. This configuration ensures that the Python dependencies are in a controlled environment, enabling effective management of the packages and libraries used in the study. Python extensively supports ML and contains several libraries for data analysis. Some of the ML libraries we used are NumPy, Pandas, Scikit-learn, SciPy, Math, and Time. A high-performance computer (Intel Core i7-7700HQ CPU @ 2.80GHz (8 CPUs), 16384MB RAM memory, and NVIDIA GeForce GTX 1050 Ti graphic card) was used to handle the large datasets and complicated computations.

6.2. Implementation Operations

The implementation of the proposed IDS model included numerous essential steps, each step assisted in the development of an effective and robust system. The first step for preprocessing was transformation using label encoder method. The second step for preprocessing was normalization using Min-Max Scaler method. After that, feature selection was implemented using the proposed modified-GWO algorithm. Finally, the model's performance was evaluated using XGBoost, RF, HGB, and MNB classification algorithms. Stratified K-Folding was used, where k was set to 5 in order to split the dataset into five parts (five Folds), and the model was trained and tested five times, each time with a different fold ensuring a better evaluation for the overall performance of the model. Algorithm 2 shows the proposed modified-GWO pseudo code.

Algorithm 2. Modified-GWO pseudo code

Input:

- Dataset: D={X, y}
- Number of wolves (population size): N
- Maximum number of iterations: T_{max}

- Feature dimension: d
- Mutation probability: p_m=0.2
- Fitness weighting parameters: w
- Classifier C.

Output:

- Best feature subset X_α

Initialization

1. Initialize wolves: X_i, i = [1, N], X_{ij}, j∈[1, d], $X_{ij} \in \{0,1\}$
2. Initialize control parameter: a=2
3. For each wolf X_i, compute fitness using, Fitness(Xi)=w·(1−Accuracy(Xi))+(1-w)*|Xi|
4. Identify: X_α (best fitness), X_β (second best), X_δ (third best)

Main Loop

5. While (iteration<T_{max} && improvement > threshold)
6. Generate random vectors r1,r2~U(0,1)
7. Update coefficient vectors: A=2a·r1−a, C=2·r_2
8. For each wolf X, compute:

$$D\alpha = |C1 * X\alpha - X|, X1 = X\alpha - A1 * D\alpha$$
$$D\beta = |C2 * X\beta - X|, X2 = X\beta - A2 * D\beta$$
$$D\delta = |C3 * X\delta - X|, X3 = X\delta - A3 * D\delta$$

9. Update wolf position:

$$X(t+1) = \frac{(X1 + X2 + X3)}{3}$$

10. Apply binarization
11. For each bit Xij in Xi: $Xij_{new} = \begin{cases} 1 - Xij, & if\ rand() < pm \\ Xij, & otherwise \end{cases}$
12. Recalculate fitness for all wolves.
13. Improvement = $X\alpha_{new}$ - $X\alpha$
14. Update $X\alpha$, $X\beta$, and $X\delta$.
15. Calculate a=2−(2t/T_{max})
16. Return $X\alpha$

The implementation of the proposed modified GWO algorithm operates as follows:

- **Population Initialization:** Feature subsets (wolves) are initialized using a combination of random initialization and Latin Hypercube Sampling (LHS) to ensure diversity in the search space.
- **Knowledge-Guided Seeding:** The *Flag* and *Service* features are included only in the initial population as part of a knowledge-guided initialization, while all features remain free to be selected or discarded during subsequent optimization.
- **Parameter Configuration:** The population size is set to 20 wolves, and the maximum number of iterations is fixed at 200.
- **Fitness Evaluation:** Each feature subset is evaluated using a fitness function based on a Decision Tree (DT) classifier, which jointly considers classification accuracy and feature subset size.
- **Fitness Objective Formulation:** The error rate combines prediction accuracy and feature reduction, weighted by control parameters α and β, as defined in Equation (5).

$$error\ rate = alpha * (1 - Accuracy) + beta * \left(\frac{number\ of\ selected\ features}{maximum\ number\ of\ features}\right) \quad (5)$$

- **Leader Identification:** In each iteration, the three best-performing solutions are identified as the alpha, beta, and delta wolves.
- **Position Update Mechanism:** The positions of all wolves are updated based on the mean influence of the alpha, beta, and delta solutions, following the standard GWO update strategy.
- **Diversity Enhancement:** A bit-flip mutation operator with a mutation rate of 0.2 is applied to the updated feature subsets to enhance exploration and mitigate premature convergence.
- **Fitness Re-evaluation:** After mutation, the fitness of each feature subset is recalculated to reflect the updated solutions.
- **Termination and Output:** After completing 200 iterations, the algorithm outputs the alpha wolf as the optimal feature subset.

The proposed GWO returns the 16 features selected out of 41 features. The selected features are shown in Fig. 10 which outlines the feature permutation importance for each of the selected features and their role and significance as individual features in predictions for the model.

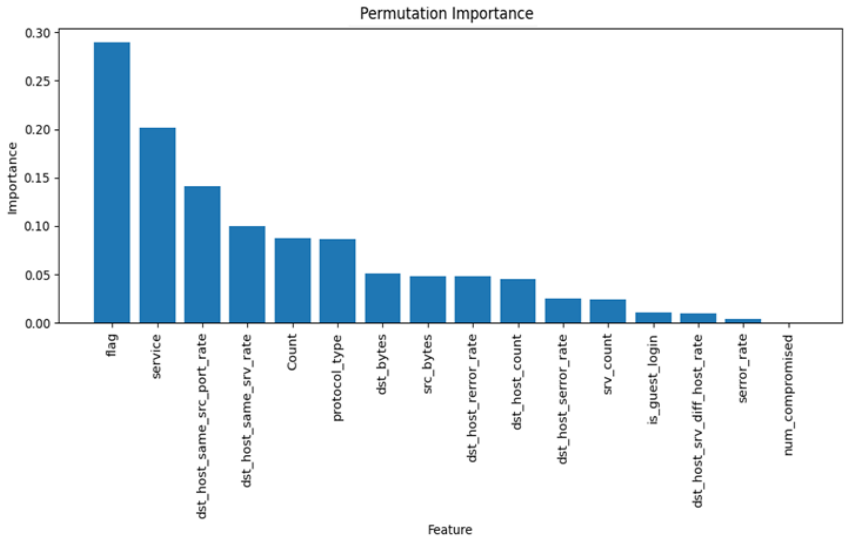

Figure 9. Feature permutation importance

The proposed modified-GWO convergence for the 200 iterations is shown in Fig. 11 which outlines the fitness during these iterations.

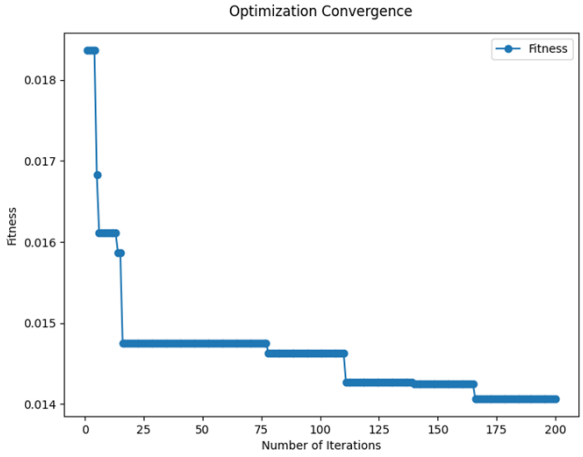

Figure 10. GWO convergence for the 200 iterations

6.3. Evaluation Metrics

The proposed model was evaluated using various evaluation metrics commonly utilized in research to assess the performance of IDSs. The following metrics provide a detailed insight into how well the model is classifying data its correctness and its efficiency in classifying intrusions [69-72].

- False Negative (FN): The number of samples that are in the intrusions class in the dataset and are incorrectly predicted in the normal class.
- False Positive (FP): The number of samples that are in the normal class in the dataset and are incorrectly predicted in the intrusions class.
- True Positive (TP): The number of samples that are in the intrusions class in the dataset and are correctly predicted in the intrusions class.
- True Negative (TN): The number of samples that are in the normal class in the dataset and are correctly predicted in the normal class.
- Convergence Time: This metric was chosen to measure the speed of the model's training process.
- Accuracy: Accuracy serves as a fundamental measure of a model's overall correctness. It is computed as the ratio of correctly classified instances to the total number of instances, encompassing both true positives and true negatives as seen in Equation (6).

$$Accuracy = \frac{(TP+TN)}{(TP+TN+FP+FN)} \quad (6)$$

- Precision: Precision is a key metric to understand the model's performance regarding true positives. It is calculated as the ratio of correctly predicted positive observations to the total predicted positives as seen in Equation (7).

$$Precision = \frac{(TP)}{(TP+FP)} \quad (7)$$

- F1-score: The F1-score serves as a balanced measure of a model's precision and recall. It is especially significant when handling imbalanced datasets where one class might be underrepresented. It is calculated as seen in Equation (8).

$$F1 - Score = \frac{2*(Precision*Recall)}{(Precision+Recall)} \quad (8)$$

By evaluating the model using these various metrics provide a comprehensive overview into the model's effectiveness and correctness using the NSL-KDD dataset.

6.4. Results

6.4.1. The Proposed Modified-GWO Model

The model was evaluated using XGBoost, RF, HGB, and MNB classifiers. Table 4 provides the results of these classifiers.

Table 4. Classification results for all classifiers

Classifier	Accuracy	Precision	F1-Score	Convergence Time
XGBoost	99.52%	99.47%	99.46%	0.578023434
RF	99.23%	99.18%	99.15%	0.882414341
HGB	90.30%	94.98%	92.4%	0.015620708
MNB	86.77%	85.49%	82.07%	0.288537502

XGBoost delivered the highest performance, with 99.52% accuracy, 99.47% precision, and 99.46% F1-score, converging in 0.578 seconds. The RF classifier also performed well, achieving 99.23% accuracy and 99.15% F1-score, but required a longer convergence time of 0.882 seconds. HGB achieved moderate results, with 90.30% accuracy, 94.98% precision, and 92.4% F1-score, and was the fastest to converge at 0.0156 seconds. MNB had the lowest performance, with 86.77% accuracy and 82.07% F1-score, converging in 0.289 seconds. Figs. 12 to 15 summarize the classifiers' rankings by accuracy, precision, F1-score, and convergence time.

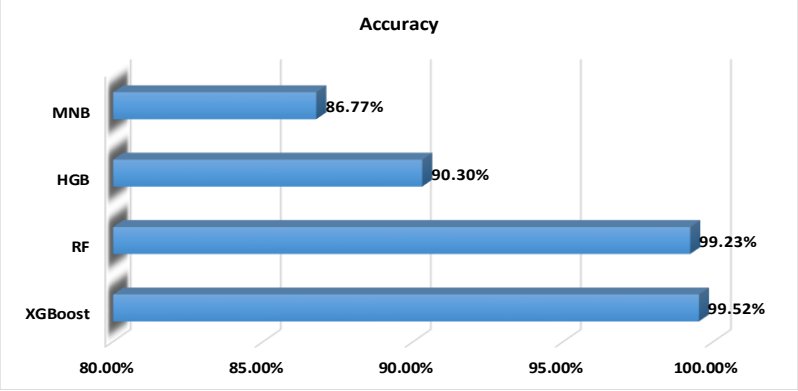

Figure 11. Classifiers Accuracy Comparison

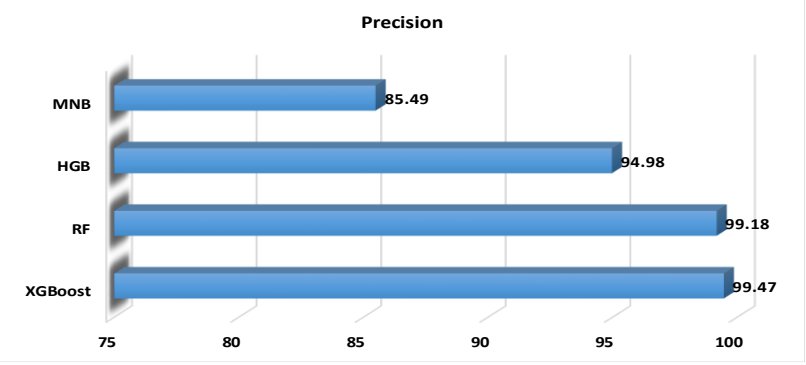

Figure 12. Classifiers precision comparison

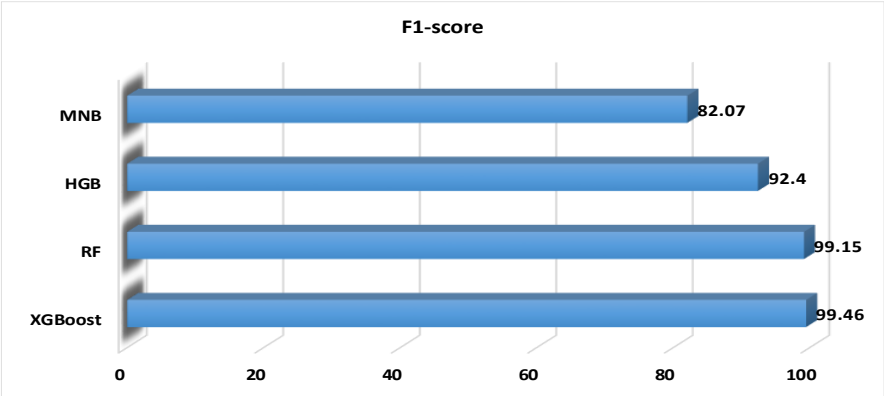

Figure 13. Classifiers F1-score comparison

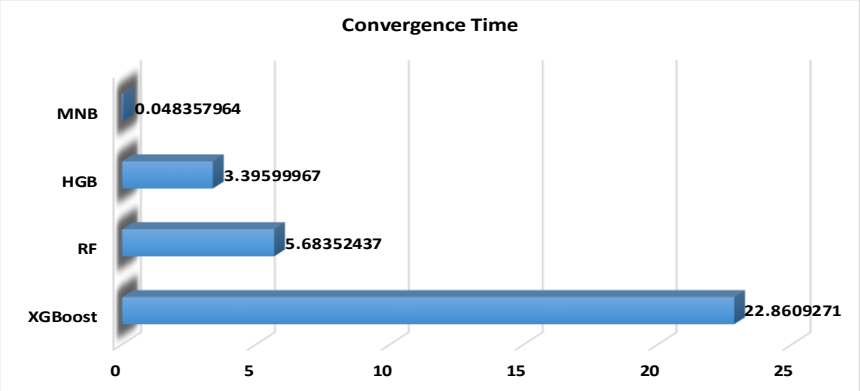

Figure 14. Classifiers convergence time comparison

Table 5 presents the mean accuracy, standard deviation, and 95% confidence intervals obtained across five paired evaluations for all classifiers. The proposed XGBoost-based model achieves the highest mean accuracy with low variance, indicating stable performance under the adopted evaluation setting. Statistical significance was assessed using a paired non-parametric Wilcoxon signed-rank test on accuracy values, as summarized in Table 6. Although the proposed model consistently outperformed the baseline classifiers, the differences did not reach statistical significance at the 0.05 level ($p = 0.0625$), indicating consistent but not statistically significant performance gains.

Table 5. Classification results for all classifiers

Classifier	Mean Accuracy (%)	Std (%)	95% CI
XGBoost	99.52	0.076	[99.43, 99.61]
RF	99.23	0.083	[99.13, 99.33]
HGB	90.30	0.158	[90.10, 90.50]
MNB	86.77	0.114	[86.63, 86.91]

Table 6. Classification results for all classifiers

Comparison	Test	p-value
XGBoost vs RF	Wilcoxon signed-rank	0.0625
XGBoost vs HGB	Wilcoxon signed-rank	0.0625
XGBoost vs MNB	Wilcoxon signed-rank	0.0625

6.4.2. Original GWO VS The Proposed Modified-GWO

XGBoost showed the best results, reaching 99.52% accuracy with the modified GWO, slightly higher than 99.40% with the original. The RF classifier also performed well, with 99.23% accuracy using the modified GWO, compared to 99.05% before. The HGB classifier improved from 89.64% to 90.30%, and the MNB classifier increased from 85.09% to 86.77%. Fig. 16 presents a comparison of accuracy for all classifiers using both versions of GWO. The results indicate that the modified GWO consistently leads to better accuracy, supporting the effectiveness of the proposed optimization method.

Fig. 17 demonstrates that the modified GWO outperforms the original GWO in precision for most classifiers. XGBoost precision increased from 99.30% to 99.47%, and RF improved from 99.01% to 99.18%. Precision for HGB decreased from 97.10% to 94.98%, and MNB declined from 86.68% to 85.49%. Overall,

the modified GWO enhanced precision in most cases, especially among high-performing classifiers, supporting its effectiveness in identifying positive instances.

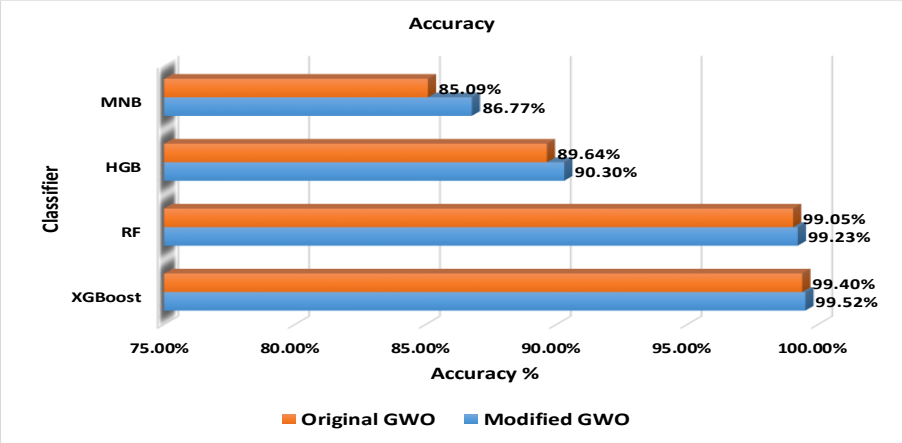

Figure 15. Original vs proposed modified-GWO Accuracies

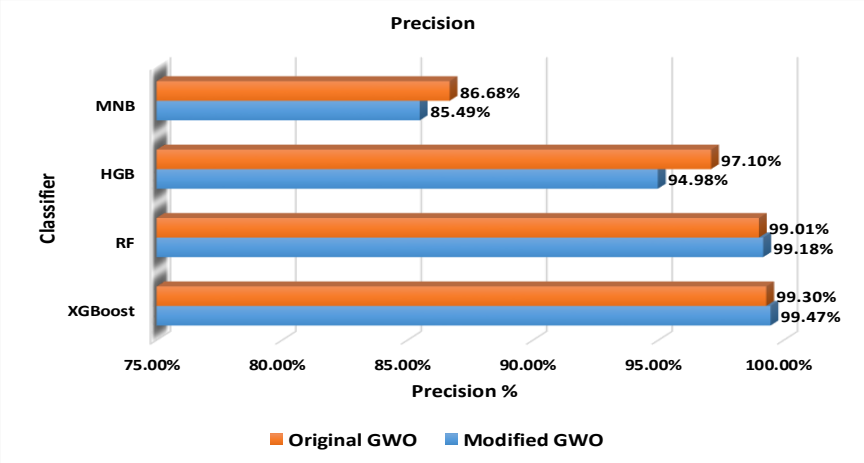

Figure 16. Original vs proposed modified-GWO precision

The proposed modified GWO algorithm yields measurable improvements in F1-score for most classifiers, as shown in Fig. 18. The F1-score for the XGBoost classifier increased from 99.30% to 99.46%, while the RF classifier improved from 98.94% to 99.15%. The HGB classifier experienced a minor decrease from 93.07% to 92.40%. In contrast, the MNB classifier improved from 80.04% to 82.07%. These results indicate that the modified-GWO enhances the balance between precision and recall for most classifiers, thereby improving overall predictive performance.

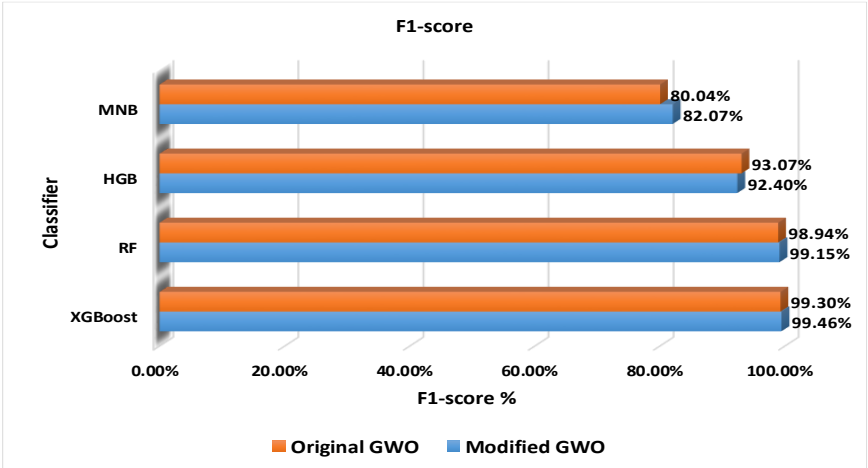

Figure 17. Original vs proposed modified-GWO F1-Score

6.4.3. The Proposed Modified-GWO VS Other Models

The comparison presented in Fig. 19 provides a contextual performance overview, demonstrating that the proposed GWO-based model achieves competitive and high accuracy relative to other intrusion detection systems reported in the literature. It achieved the highest accuracy of 99.52%, exceeding Alzubi *et al.* (99.22%) by 0.30% and Yerriswamy & Murtugudde (98.62%) by 0.90%. The improvement becomes more substantial when compared with Shakya (97%) and Safaldin *et al.* (96%), showing gains of 2.52% and 3.52%, respectively. The largest difference is observed with Almazini & Ku-Mahamud (87.46%), where the proposed model achieved an impressive 12.06% higher accuracy. It should be noted that these comparative results are drawn from the original studies and may involve differences in data splits, preprocessing steps, and evaluation protocols. Accordingly, Fig. 19 intended to contextualize reported performance trends rather than to represent a strictly controlled head-to-head comparison. These results indicate the strong potential of the proposed GWO framework, highlighting its enhanced optimization capability and its effectiveness in improving intrusion detection accuracy within the adopted experimental setting.

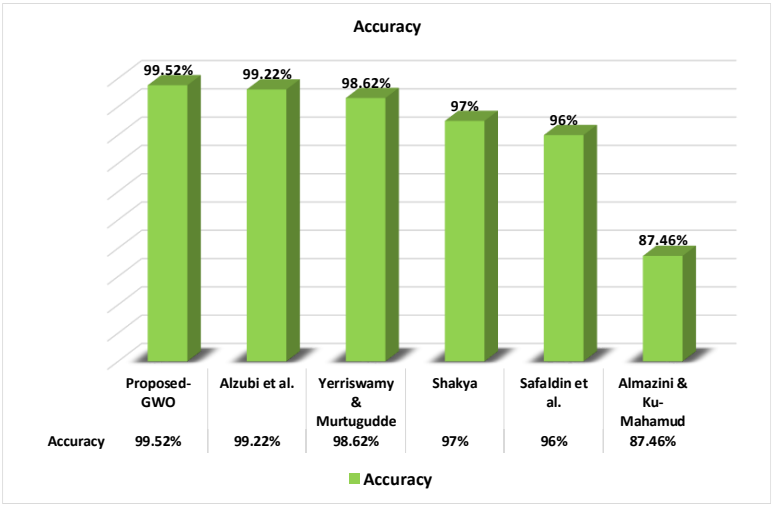

Figure 19. Accuracy of the proposed GWO vs other existing IDS models

7. Limitations and Future Work

While the proposed framework demonstrates strong performance on the NSL-KDD dataset, certain limitations merit consideration. NSL-KDD remains a well-established and widely adopted benchmark comprising 150,620 labeled records and diverse attack categories; however, it does not fully reflect characteristics of contemporary network environments, such as encrypted traffic and evolving attack strategies. Nevertheless, its structured labeling and attack diversity provide a controlled experimental setting for systematically analyzing the behavior and effectiveness of optimization-driven feature selection mechanisms, which is the primary focus of this study.

In addition, the mutation rate in the proposed algorithm was fixed at 0.2 based on empirical stability observed during preliminary experimentation and within commonly adopted ranges for binary evolutionary optimization. While this setting provided consistent performance, a comprehensive sensitivity analysis across different mutation rates (e.g., 0.1–0.3) was not conducted and may offer further insight into parameter robustness and dataset-dependent behavior.

Future work will therefore focus on extending the experimental validation to more recent and realistic intrusion detection datasets, such as CIC-IDS-2017, UNSW-NB15, and ToN-IoT, to assess generalization to contemporary attack patterns. Moreover, a systematic parameter sensitivity analysis, including mutation-rate tuning and adaptive mutation strategies, will be explored to further enhance robustness and optimize performance across diverse network environments.

8. Conclusion

This study has presented significant contributions to the field of IDS by introducing a proposed modified-GWO algorithm that enhances both optimization efficiency and detection accuracy. The modification, which incorporates an improved initialization strategy and a new mutation function, enabled

more balanced exploration and exploitation of the search space, resulting in faster and more reliable convergence. When applied for feature selection on the NSL-KDD dataset and evaluated with multiple ML classifiers, the proposed approach demonstrated superior performance across metrics. Notably, the XGBoost classifier achieved an accuracy rate of 99.52%, 99.47% precision rate, and 99.46% recall rate, outperforming existing IDS models and highlighting the robustness of the proposed method. Compared with other IDS frameworks, the proposed GWO achieved up to 12% improvement in accuracy, setting a new benchmark for intelligent intrusion detection. Overall, the combination of the modified-GWO and XGBoost offers a powerful, efficient, and scalable solution for modern IDS applications, capable of maintaining high accuracy while reducing computational overhead.

CRediT Author Contribution Statement

Abdullah Al Mosuli: Conceptualization, Methodology, Software, Data curation, Formal analysis, Writing – original draft; Mosleh Abualhaj: Conceptualization, Methodology, Supervision, Validation, Project administration, Writing – review & editing, Corresponding author; Ahmad Abu-Shareha: Methodology, Investigation, Formal analysis, Writing – review & editing; Mohamed Yousif: Validation, Resources, Visualization, Writing – review & editing; Mohammad Daoud: Formal analysis, Investigation, Writing – review & editing.

References

[1] Khaleel Ibrahim Al-Daoud and Ibrahim A. Abu-AlSondos, "Robust AI for Financial Fraud Detection in the GCC: A Hybrid Framework for Imbalance, Drift, and Adversarial Threats", *Theoretical and Applied Electronic Commerce Research*, Online ISSN: 0718-1876, Vol. 20, No. 2, 1 June 2025, pp. 1-25, Published by MDPI, DOI: 10.3390/jtaer20020121, Available: https://www.mdpi.com/0718-1876/20/2/121.

[2] Yahya Alhaj Maz, Mohammed Anbar, Selvakumar Manickam and Mosleh M. Abualhaj, "Utilizing Deep Learning, Ensemble Learning, and Transfer Learning for Enhancing Security in Internet of Things Networks through Intrusion Detection Systems", in *Proceeding of the 2025 12th International Conference on Information Technology (ICIT)*, 27-30 May 2025, Amman, Jordan, Online ISBN: 979-8-3315-0894-4, pp. 73-76, Published by IEEE, DOI: 10.1109/ICIT64950.2025.11049182, Available: https://ieeexplore.ieee.org/document/11049182.

[3] Werisha Ibrar, Danish Mahmood, Ahmad Sami Al-Shamayleh, Ghufran Ahmad, Salman Z. Alharthi *et al.*, "Generative AI: a double-edged sword in the cyber threat landscape", *Artificial Intelligence Review*, Print ISSN: 0269-2821, Online ISSN: 1573-7462, Vol. 58, No. 9, 1 July 2025, Article No. 285, Published by Springer Nature, DOI: 10.1007/s10462-025-11285-9, Available: https://link.springer.com/article/10.1007/s10462-025-11285-9.

[4] Mohammad Al-Mousa, Waleed Amer, Mosleh Abualhaj, Sultan Albilasi, Ola Nasir *et al.*, "Agile Proactive Cybercrime Evidence Analysis Model for Digital Forensics", *International Arab Journal of Information Technology (IAJIT)*, Print ISSN: 1683-3198, e-ISSN: 2309-4524, Vol. 22, No. 3, May 2025, Published by Zarqa University, DOI: https://doi.org/10.34028/iajit/22/3/15, Available: https://www.iajit.org/upload/files/Agile-Proactive-Cybercrime-Evidence-Analysis-Model-for-Digital-Forensics.pdf.

[5] Yehia Aldaaja, Mahmoud Odeh, Sawsan S. Badrakhan, Mohammad Sabri, Yousif Abdelrahim *et al.*, "Quantifying the Managerial and Practical Benefits of Cloud Computing on Information Security in Higher Education Institutions", *Applied Mathematics & Information Sciences (AMIS)*, Print ISSN: 1935-0090, Online ISSN: 2325-0399, Vol. 19, No. 1, Jan. 2025, pp. 15-24, Published by Natural Sciences Publishing, DOI: 10.18576/amis/190102, Available: https://www.naturalspublishing.com/Article.asp?ArtcID=29256.

[6] Omar Almomani, Adeeb Alsaaidah, Ahmad Adel Abu-Shareha, Abdullah Alzaqeba, Mohammed Amin Almaiah *et al.* "Enhance URL Defacement Attack Detection Using Particle Swarm Optimization and Machine Learning", *Journal of Computational and Cognitive Engineering*, Print ISSN: 2810-9570, Online ISSN: 2810-9503, Vol. 4, No. 3, Feb. 2025, Published by Bon View Publishing Pte Ltd, DOI: 10.47852/bonviewJCCE52024668, Available: https://ojs.bonviewpress.com/index.php/JCCE/article/view/4668.

[7] Mosleh Abualhaj, Ahmad Abu-Shareha, Sumaya Al-Khatib, Adeeb Alsaaidah and Mohammed Anbar, "Improving firewall performance using hybrid of optimization algorithms and decision trees classifier", *International Journal of Artificial Intelligence (IJ-AI)*, Print ISSN: 2089, Online ISSN: 4872/2252-8938, Vol. 14, No. 4, August 2025 ,pp. 2839-2848, published by the Institute of Advanced Engineering and Science (IAES), DOI: 10.11591/ijai.v14.i4.pp2839-2848, Available: https://ijai.iaescore.com/index.php/IJAI/article/view/27023.

[8] Mohammad Hiari, Yousif Alraba'nah and Iyas Qaddara, "A Deep Learning-Based Intrusion Detection System using Refined LSTM for DoS Attack Detection", *Engineering, Technology & Applied Science Research*, Print ISSN: 2241-4487, Online ISSN: 1792-8036, Vol. 15, No. 4, August 2025, pp. 25627-25633, Published by Dr D. Pylarinos, DOI: 10.48084/etasr.11499, Available: https://etasr.com/index.php/ETASR/article/view/11499.

[9] Amaal Rateb Shorman, Maen Alzubi, Mohammad Almseidin and Roqia Rateb, "Adaptive Intrusion Detection for IoT Networks using Artificial Immune System Techniques: A Comparative Study", *Journal Robotics and Control (JRC)*, Print ISSN: 2715-5056, Online ISSN: 2715-5072, Vol. 6, No. 2, March 2025, pp. 570-582, Published by Department of Agribusiness, Universitas Muhammadiyah Yogyakarta, DOI: 10.18196/jrc.v6i2.23645, Available: https://journal.umy.ac.id/index.php/jrc/article/view/23645.

[10] Mosleh M. Abualhaj, Sumaya N. Alkhatib, Mahran Al-Zyoud, Iyas Qaddara, Mohammad O. Hiari *et al.*, "Enhanced Network Communication Security Through Hybrid Dragonfly-Bat Feature Selection for Intrusion Detection", *Journal of Communications (JCM)*, Print ISSN: 2374-4367, Online ISSN: 1796-2021, Vol. 20, No. 5, October 2025, pp. 607-618, Published by Engineering and Technology Publishing, DOI: 10.12720/jcm.20.5.607-618, Available: https://www.jocm.us/show-320-2111-1.html.

[11] Sami. Qawasmeh, Ahmad Habboush, Bassam Elzaghmouri, Qasem Kharma and Da'ad Albalawneh, "Hybrid Convolutional Neural Network-Based Intrusion Detection System for Secure IoT Networks", *Tikrit J. Eng. Sci*, Print ISSN: 1813-162X, Online ISSN: 2312-7589, Vol. 32, No. SP1, 12 Aug. 2025, pp. 1-11, Published by Tikrit University, DOI: 10.25130/tjes.sp1.2025.2, Available: https://tj-es.com/ojs/index.php/tjes/article/view/2526.

[12] Sharif Naser Makhadmeh, Salam Fraihat, Mohammed Awad, Yousef Sanjalawe, Mohammed Azmi Al-Betar *et al.*, "A crossover-integrated Marine Predator Algorithm for feature selection in intrusion detection systems within IoT environments", *Internet of Things*, Print ISSN: 2543-1536, Online ISSN: 2542-6605, Vol. 31, May 2025, Art. 101536, Published by Elsevier, DOI: 10.1016/j.iot.2025.101536, Available: https://www.sciencedirect.com/science/article/abs/pii/S2542660525000496.

[13] Mosleh M. Abualhaj, Sumaya N. Al-khatib, Mahran Al Zyoud, Iyas Qaddara and Mohammed Anbar, "Enhancing Intrusion Detection System Performance Using a Hybrid of Harris Hawks and Whale Optimization Algorithms", *Engineering, Technology & Applied Science Research*, Print ISSN: 2241-4487, Online ISSN: 1792-8036, Vol. 15, No. 4, August 2025, pp. 24354-24361, Published by Dr D. Pylarinos, DOI: 10.48084/etasr.10919, Available: https://etasr.com/index.php/ETASR/article/view/10919/5161.

[14] Areen. M. Arabiat and Yousef. G. Eljaafreh, "Intrusion Detection in Wireless Sensor Networks Using ML Based Classification of Denial of Service (DoS) Attacks", *Journal of Communications* (JCM), Print ISSN: 2374-4367, Online ISSN: 1796-2021, Vol. 20, No. 4, 8 August 2025, pp. 501-514, Published by Engineering and Technology Publishing, DOI: 10.12720/jcm.20.4.501-514, Available: https://www.jocm.us/show-319-2099-1.html.

[15] Laith Abualigah and Aseel Smerat, "Future Directions in Artificial Intelligence: Trends, Challenges, and Human Implications", In *Mastering the Minds of Machines*, CRC Press, July 2025, pp 194-200, ISBN: 9781003516385, Published by Taylor & Francis, DOI: 10.1201/9781003516385-25, Available: https://www.taylorfrancis.com/chapters/edit/10.1201/9781003516385-25/

[16] Saleh Ali Alomari, Mahmoud Abdel-Salam, Ali Raza, Canan Batur Sahin, Raed Abu Zitar *et al.*, "The Evolution of Machine Learning: From Traditional Algorithms to Deep Learning Paradigms", In *Mastering the Minds of Machines*, Print ISBN: 9781032834832, Online ISBN: 9781003516385, CRC Press, 2025, pp 9-15, Published by Taylor & Francis, DOI: 10.1201/9781003516385-2, Available: https://www.taylorfrancis.com/chapters/edit/10.1201/9781003516385-2/.

[17] Mosleh Abualhaj, Sumaya Al-Khatib, Ahmad Abu-Shareha, Omar Almomani, H. Al-Mimi, A. Allawee *et al.*, "Spam detection boosted by firefly-based feature selection and optimized Classifiers", *International Journal of Advances in Soft Computing and its Applications*, Print ISSN: 2710-1274, Online ISSN: 2074-8523, Vol. 17, No. 3, Nov. 2025, pp. 1-19., Published by Al-Zaytoonah University of Jordan, DOI: https://doi.org/10.15849/IJASCA.251130.02, Available: https://journals.zuj.edu.jo/ijasca/PapersUploaded/2025.3.2.pdf

[18] Rawan D. Alabdallat, Moslem M. Abualhaj and Ahmad Abu-Shareha, "Enhanced Multiclass Android Malware Detection Using a Modified Dwarf Mongoose Algorithm", *International Journal of Analysis and Applications*, Online ISSN: 2291-8639, Vol. 23, October 2025, pp. 1-19, Published by Semnan University, Center of Excellence in Nonlinear Analysis and Applications, DOI: 10.28924/2291-8639-23-2025-248, Available: https://etamaths.com/index.php/ijaa/article/view/4452.

[19] Tao Yang and Dandan Song, "Research on Unmanned Path Planning of Intelligent Vehicle Based on Swarm Intelligence Algorithm", *Annals of Emerging Technologies in Computing (AETiC)*, Print ISSN: 2516-0281, Online ISSN: 2516-029X, Vol. 9, No. 1, 1st January 2025, pp. 73–86, Published by International Association for Educators and Researchers (IAER), DOI: 10.33166/AETiC.2025.01.005. Available: https://aetic.theiaer.org/archive/v9/v9n1/p5.html.

[20] Sumbul Azeem, Shazia Javed, Iftikhar Naseer, Oualid Ali and Taher M. Ghazal, "A New Hybrid PSO-HHO Wrapper Based Optimization for Feature Selection", *IEEE ACCESS*, Online ISSN: 2169-3536, Vol. 11, January 2025, pp. 1234-1245. Published by IEEE, DOI: 10.1109/ACCESS.2025.3570901, Available: https://ieeexplore.ieee.org/document/11005976.

[21] Yousef Sanjakawe, Salam Al-E'mari, Mosleh Abualhaj, Sharif Naser Makhadmeh, Mohammad A. Alsharaiah *et al.*, "Recent advances in secretary bird optimization algorithm, its variants and applications", *Evolutionary Intelligence*, Print ISSN: 1864-5909, Online ISSN: 1864-5917, Vol. 18, No. 65, 30 May 2025, Published by Springer Nature, DOI: 10.1007/s12065-025-01054-6, Available: https://link.springer.com/article/10.1007/s12065-025-01054-6.

[22] Khaled Houssam Mahfouz, Mohammed Azmi Al Betar and Sharif Naser Makhadmeh, "Mitigating the task scheduling problem in fog computing environments using improved marine predators optimization algorithm", *Computing*, Print ISSN: 0010-485X, Online ISSN: 1436-5057, Vol. 107, No. 7, 13 June 2025, Published by Springer Nature, DOI: 10.1007/s00607-025-01502-2, Available: https://link.springer.com/article/10.1007/s00607-025-01502-2.

[23] Roqia Rateb, Amal Shorman, Ahmad Sami Al-Shamayleh, Areej Alshorman and Laith H. Baniata, "A Designing Framework for Ant Colony Algorithm for Managing Cognitive Insomnia", *Journal of Computer Science,* Print ISSN: 1549-3636, Online ISSN: 1552-6607, Vol. 21, No. 7, 7 July 2025, pp. 1554-1564, Published by Science Publications, DOI: 10.3844/jcssp.2025.1554.1564, Available: https://thescipub.com/abstract/jcssp.2025.1554.1564.

[24] Rawan Alaballat, Mosleh Abualhaj and Ahmad Abu-Shareha, "Android Malware Detection Using a Modified Dwarf Mongoose Algorithm", *International Journal of Intelligent Engineering & Systems*, Print ISSN: 2185-310X, Online ISSN: 2185-3118, Vol. 18, No. 8, October 2025, pp. 336-351, Published by Intelligent Network and Systems Society, DOI: 10.22266/ijies2025.0930.21, Available: https://oaji.net/articles/2023/3603-1755588454.pdf.

[25] Mazin Arabasy and Rehab Salaheldin Ghoneim, "Enhancing sustainable urban design using machine learning: comparative analysis of seven metaheuristic algorithms in energy-efficient digital architecture", *Frontiers in Built Environment*, Online ISSN: 2297-3362, Vol. 11, August 2025, Published by Frontiers Media S.A., DOI: 10.3389/fbuil.2025.1526209, Available: https://www.frontiersin.org/journals/built-environment/articles/10.3389/fbuil.2025.1526209/full.

[26] Laith Abualigah, Mohammad H. Almomani, Saleh Ali Alomari, Raed Abu Zitar, Hazem Migdady *et al.*, "Optimizing Intrusion Detection in Wireless Sensor Networks via the Improved Chameleon Swarm Algorithm for Feature Selection", *IET Communications,* Print ISSN: 1751-8628, Online ISSN: 1751-8636, Vol. 19, No. 1, March 2025, art. E70029, Published by John Wiley & Sons, DOI: 0.1049/cmu2.70029, Available: https://ietresearch.onlinelibrary.wiley.com/doi/full/10.1049/cmu2.70029.

[27] Premkumar Manoharan, Sowmya Ravichandran, Jagarapu S. V. Siva Kumar, Mustafa Abdullah, Tan Ching Sin and Tengku Juhana Tengku Hashim, "Electrical equivalent circuit parameter estimation of commercial induction machines using an enhanced grey wolf optimization algorithm", *Biomimetics,* Online ISSN: 2313-7673, Vol. 10, No. 4, April 2025, art. 228, Published by MDPI, DOI: 10.3390/biomimetics10040228, Available: https://www.mdpi.com/2313-7673/10/4/228.

[28] Feriel Abdelmalek, Hamza Afghoul, Fateh Krim, Mohit Bajaj and Vojtech Blazek, "Experimental validation of novel hybrid Grey Wolf Equilibrium Optimization for MPPT to improve the efficiency of solar photovoltaic system", *Results in Engineering*, Online ISSN: 2590-1230, Vol. 25, March 2025, art. 103831, Published by Elsevier, DOI: 10.1016/j.rineng.2024.103831, Available: https://www.sciencedirect.com/science/article/pii/S2590123024020747.

[29] Mosleh Mohammd Abualhaj, Sumaya Al-Khatib, Nida Al Shafi, Iyas Qaddara and Abdualla Hyassat, "Utilizing Gray Wolf Optimization Algorithm in Malware Forensic Investigation", *Journal of Computational and Cognitive Engineering*, Print ISSN: 2810-9570, Online ISSN: 2810-9503, May 2025, Published by Bon View Publishing Pte Ltd, DOI: 10.47852/bonviewJCCE52025053, Available: https://ojs.bonviewpress.com/index.php/JCCE/article/view/5053.

[30] Mohammed Otair, Faten Ali Qawaqzeh, Saleh Ali Alomari, Raed abu Zitar, Hazem Migdady *et al.*, "Enhanced Arithmetic Optimization Algorithm for Intrusion Detection in Wireless Sensor Networks", *International Journal of Communication Systems*, Print ISSN: 1074-5351, Online ISSN: 1099-1131, Vol. 38, No. 10, 10 Jul. 2025, art. e70122, Published by John Wiley & Sons, DOI: 10.1002/dac.70122, Available: https://onlinelibrary.wiley.com/doi/full/10.1002/dac.70122.

[31] Christopher Ifeanyi Eke, Ahmad Sami Al-Shamayleh, M. Phiri, K. Maswadi, D. K. Kwaghtyo *et al.*, "Machine Learning Based Mobile Big Data Analytics: State-of-the-art Applications, Taxonomy, Challenges and Future Research Directions", *Nigerian Journal of Technological Development*, Print ISSN: 0189-9546, Online ISSN: 2437-2110, Vol. 22, No. 4, September 2025, pp. 65–89, Published by University of Ilorin, Faculty of Engineering and Technology, DOI: 10.63746/njtd.v22i4.3385, Available: https://journal.njtd.com.ng/index.php/njtd/article/view/3385.

[32] Osama I. Ramadan, Lashin S. Ali, Yasser Ramadan, Randa M. Abobaker, Hoda M. Flifel *et al.*, "Enhancing Breast Cancer Classification based on BPSO Feature Selection and Machine Learning Techniques", *Engineering, Technology & Applied Science Research*, Print ISSN: 2241-4487, Online ISSN: 1792-8036, Vol. 15, No. 3, June 2025, pp. 23907-23916, Published by Dr D. Pylarinos, DOI:10.48084/etasr.10900, Available: https://etasr.com/index.php/ETASR/article/view/10900.

[33] Prashant G. Sawarkar, Amol Pote, Mohammad Amir Khan, Mukhtar Hamid Abed, Ahmad Adnan Hadi *et al.*, "Sustainable Geopolymer Concrete Evaluation Using Machine Learning and PDP Analysis: Comparative Insights into Strength Prediction", *Hybrid Advances*, Print ISSN: 2773-207X, October 2025, art. 100569, Published by Elsevier, DOI: 10.1016/j.hybadv.2025.100569, Available: https://www.sciencedirect.com/science/article/pii/S2773207X25001939.

[34] Prasannavenkateson Theerthagiri, "Liver disease classification using histogram-based gradient boosting classification tree with feature selection algorithm", *Biomedical Signal Processing and Control*, Print ISSN: 1746-8094, Online ISSN: 1746-8108, Vol. 100, February 2025, art. 107102, Published by Elsevier, DOI: 10.1016/j.bspc.2024.107102, Available: https://www.sciencedirect.com/science/article/abs/pii/S1746809424011601.

[35] Yousif Alraba'nah, Saleh Al-Sharaeh and Ghosoun Al Hindi, "Enhancing Intrusion Detection Using Hybrid Long Short-Term Memory and XGBoost", *Journal of Soft Computing and Data Mining*, Online ISSN: 2716-621X, Vol. 6, No. 1, June 2025, pp. 247-261, Published by Penerbit UTHM, DOI: 10.30880/jscdm.2025.06.01.016, Available: https://publisher.uthm.edu.my/ojs/index.php/jscdm/article/view/20781/7361.

[36] Md Abdur Rahim, Md Alfaz Hossain, Md Najmul Hossain, Jungpil Shin and Keun Soo Yun, "Stacked Ensemble-Based Type-2 Diabetes Prediction Using Machine Learning Techniques", *Annals of Emerging Technologies in Computing (AETiC)*, Print ISSN: 2516-0281, Online ISSN: 2516-029X, Vol. 7, No. 1, Jan 203, pp. 30-39, Published by International Association for Educators and Researchers (IAER), DOI: 10.33166/AETiC.2023.01.003, Available: https://aetic.theiaer.org/archive/v7/v7n1/p3.html.

[37] Amjed Zraiqat, Belal Batiha, Maha Al Soudi, Ibraheem Kasim Ibraheem, Aseel Smerat *et al*., "Detective Algorithm: A Novel Human-inspired Metaheuristic for Solving Real-world Constrained Optimization Problems", *International Journal of Intelligent Engineering & Systems*, Print ISSN: 2185-310X, Online ISSN: 2185-3118, Vol. 18, No. 10, November 2025, pp. 33-46, Published by Intelligent Network and Systems Society, DOI: 10.22266/ijies2025.1130.03, Available: https://oaji.net/articles/2023/3603-1760661318.pdf.

[38] Mosleh M. Abualhaj, Mahran Al-Zyoud, Adeeb Alsaaidah, Ahmad Abu-Shareha and Sumaya Al-Khatib, "Enhancing malware detection through self-union feature selection using Firefly Algorithm with Random Forest classification", *International Journal of Intelligent Engineering & Systems*, Print ISSN: 2185-310X, Online ISSN: 2185-3118, Vol. 17, No. 4, May 2024, pp. 376-389, Published by Intelligent Network and Systems Society, DOI: 10.22266/ijies2024.0831.29, Available: https://inass.org/wp-content/uploads/2024/03/2024083129-2.pdf.

[39] Yunyun Liu, Azizan As'arry, Mohd Khair Hassan, Abdul Aziz Hairuddin and Hesham Mohamad, "Review of the grey wolf optimization algorithm: variants and applications", *Neural Computing and Applications*, Print ISSN: 0941-0643, Online ISSN: 1433-3058, Vol. 36, No. 6, November 2023, pp. 2713–2735, Published by Springer Nature, DOI: 10.1007/s00521-023-09202-8, Available: https://dl.acm.org/doi/abs/10.1007/s00521-023-09202-8.

[40] Qian Zhang, Shaoyong Han, Mohammed A. El-Meligy and Mehdi Tlija, "Active control vibrations of aircraft wings under dynamic loading: Introducing PSO-GWO algorithm to predict dynamical information", *Aerospace Science and Technology*, Print ISSN: 1270-9638, Vol. 153, 31 July 2024, art. 109430, Published by Elsevier, DOI: 10.1016/j.ast.2024.109430, Available: https://www.sciencedirect.com/science/article/abs/pii/S1270963824005613.

[41] Mosleh M. Abualhaj, Ahmad Adel Abu-Shareha, Mohammad O. Hiari, Yousif Alrabanah, Mahran Al-Zyoud *et al*., Print ISSN: 2158-107X, Online ISSN: 2156-5570, "A Paradigm for DoS Attack Disclosure using Machine Learning Techniques", *International Journal of Advanced Computer Science and Applications (IJACSA)*, Vol. 13, No. 3, October 2022, pp. 192–200, Published by Science and Information Organization, DOI: 10.14569/IJACSA.2022.0130325, Available: https://thesai.org/Downloads/Volume13No3/Paper_25-A_Paradigm_for_DoS_Attack_Disclosure.pdf.

[42] Ünal Çavuşoğlu, "A new hybrid approach for intrusion detection using machine learning methods", *Applied Intelligence*, Print ISSN: 0924-669X, Online ISSN: 1573-7497, Vol. 49, No. 7, February 2019, pp. 2735–2761, Published by Springer Nature, DOI: 10.1007/s10489-018-1310-8, Available: https://link.springer.com/article/10.1007/s10489-018-01408-x.

[43] Qusay M. Alzubi, Mohammed Anbar, Zakaria N. M. Alqattan, Mohammed Azmi Al-Betar and Rosni Abdullah, "Intrusion detection system based on a modified binary grey wolf optimisation", *Neural Computing and Applications*, Print ISSN: 0941-0643, Online ISSN: 1433-3058, Vol. 32, February 2020, pp. 6125–6137, Published by Springer Nature, DOI: 10.1007/s00521-019-04360-3, Available: https://link.springer.com/article/10.1007/s00521-019-04103-1.

[44] Yerriswamy T and Gururaj Murtugudde, "An efficient algorithm for anomaly intrusion detection in a network", *International Conference on Computing System and its Applications (ICCSA- 2021)*, Vol. 2, No. 2, November 2021, pp. 255–260, Published by Elsevier, DOI: https://doi.org/10.1016/j.gltp.2021.08.066, Available: https://www.sciencedirect.com/science/article/pii/S2666285X21000947.

[45] Subarna Shakya, "Modified gray wolf feature selection and machine learning classification for wireless sensor network intrusion detection", *IRO Journal on Sustainable Wireless Systems*, Online ISSN: 2582-3167, Vol. 3, No. 2, 14 June 2021, pp. 118–127, Published by Inventive Research Organization, DOI: 10.36548/jsws.2021.2.006, Available: https://irojournals.com/irosws/article/view/3/2/6.

[46] Mukaram Safaldin, Mohammed Otair and Laith Abualigah, "Improved binary gray wolf optimizer and SVM for intrusion detection system in wireless sensor networks", *Journal of Ambient Intelligence and Humanized Computing*, Print ISSN: 1868-5137, Online ISSN: 1868-5145, Vol. 12, June 2021, pp. 1559–1576, Published by Springer Nature, DOI: 10.1007/s12652-020-02562-1, Available: https://link.springer.com/article/10.1007/s12652-020-02228-z.

[47] Hussein Almazini and Ku Ruhana Ku-Mahamud, "Grey Wolf Optimization Parameter Control for Feature Selection in Anomaly Detection", *International Journal of Intelligent Engineering & Systems*, Print ISSN: 2185-310X, Online ISSN: 2185-3118, Vol. 14, No. 2, January 2021, pp. 474–483, Published by Intelligent Network and Systems Society, DOI: 10.22266/ijies2021.043, Available: https://oaji.net/articles/2021/3603-1614237950.pdf.

[48] Maduri Madhavi and DR. Nethravathi, "Gradient Boosted Decision Tree (GBDT) and Grey Wolf Optimization (GWO) based Intrusion Detection Model", *Journal of Theoretical and Applied Information Technology*, Print ISSN: 1992-8645, Online ISSN: 1817-3195, Published by Little Lion Scientific, Vol. 100, No. 16, 31 Aug. 2022, pp. 4937–4951, DOI: 10.14569/JTAT.2022.100.16.4937, Available: https://www.jatit.org/volumes/Vol100No16/5Vol100No16.pdf .

[49] Somnath Chatterjee, Vaibhav Shaw and Ranit Das, "Multi-Stage Intrusion Detection System aided by Grey Wolf optimization algorithm", *Cluster Computing*, Print ISSN: 1386-7857, Vol. 27, No. 3, 10 November 2023, pp. 3819–3836, Published by Springer Nature, DOI: 10.1007/s10586-023-04179-4, Available: https://link.springer.com/article/10.1007/s10586-023-04179-4.

[50] Muntadher Idrees Ali, Savitha K, Aseel Smerat, Kolluru Suresh Babu, Sivakumar G *et al.* "Multimodal Deep Learning Model for Measuring the Impact of Social Media Advertising Using Visual-Linguistic Representation Learning", *Journal of Machine and Computing*, Online ISSN: 2788-7669, Vol. 5, No. 4, October 2025, pp. 2566-2573, Published by AnaPub Publications, DOI: 10.53759/7669/jmc202505197, Available: https://anapub.co.ke/journals/jmc/jmc_pdf/2025/jmc_volume_5-issue_4/JMC202505197.pdf.

[51] Zeeshan Saleem Mufti, Kashaf Mahboob, Muhammad Nauman Aslam, Sadaf Hussain, Abdoalrahman S.A. Omer *et al.*, "Spectral analysis of Cupric oxide (CuO) and Graphene Oxide (GO) via machine learning techniques", *Egyptian Informatics Journal*, Print ISSN: 1110-8665, Online ISSN: 2090-4754, Vol. 29, March 2025, art. 100632, Published by Elsevier, DOI: 10.1016/j.eij.2025.100632, Available: https://www.sciencedirect.com/science/article/pii/S1110866525000258 .

[52] Ahmad Yousef Areiqat, "Leveraging AI and FinTech: Driving Business Innovation in the Fourth Industrial Revolution", In *Achieving Sustainable Business through AI, Technology Education and Computer Science*: Volume 1: Computer Science, Business Sustainability, and Competitive Advantage, Allam Hamdan, Ed., Springer Nature, Cham, Switzerland, November 2024, pp. 703–709, DOI: 10.1007/978-3-031-70855-8_60., Available: https://link.springer.com/book/10.1007/978-3-031-70855-8.

[53] Yahya Alhaj Maz, Mohammed Anbar, Selvakumar Manickam, Mosleh M. Abualhaj, Sultan Ahmed Almalki *et al.*, "Transfer Learning-Based Approach with an Ensemble Classifier for Detecting Keylogging Attack on the Internet of Things", *Computers, Materials & Continua*, Print ISSN 1546-2218, Online ISSN: 1546-2226, Vol. 85, No. 3, October 2025, pp. 5287-5307, Published by Tech Science Press, DOI: 10.32604/cmc.2025.068257, Available: https://www.techscience.com/cmc/v85n3/64174/html.

[54] Mazin Arabasy and Rehab Salaheldin Ghoneim, "Enhancing sustainable urban design using machine learning: comparative analysis of seven metaheuristic algorithms in energy-efficient digital architecture", *Frontiers in Built Environment*, Print ISSN 2297-3362, Vol. 11, 18 June 2025, Published by Frontiers Media S.A., DOI: 10.3389/fbuil.20251526209, Available: https://www.frontiersin.org/journals/built-environment/articles/10.3389/fbuil.2025.1526209/full .

[55] Nasim Matar, Feras Alnaimat, Ahmad Al-Qerem, Shadi Nashwan and Issam Jebreen, "Optimizing Multimodal Feature Alignment for Enhanced Text-to-Video Retrieval Performance", in *Proceeding of the 12th International Conference on Information Technology, ICIT 2025*, 27-30 May 2025, Amman, Jordan, pp 242-246, Published by IEEE, DOI: 10.1109/ICIT64950.2025.11049110, Available: https://ieeexplore.ieee.org/document/11049110.

[56] Mosleh M. Abualhaj, Qusai Y. Shambour, Ahmad Adel Abu-Shareha, Sumaya N. Al-Khatib *et al.*, "Enhancing malware detection through self-union feature selection using Gray Wolf Optimizer", *Indonesian Journal of Electrical Engineering and Computer Science*, Online ISSN: 2502-4752, Vol. 37, No. 1, January 2025, pp. 197–205, Published by Institute of Advanced Engineering and Science, DOI:10.11591/ijeecs.v37.i1.pp197-205, Available: https://ijeecs.iaescore.com/index.php/IJEECS/article/view/38458/18848 .

[57] Sulieman Ibraheem Mohammad, Ala'a M. Al-Momani, Asokan Vasudevan, Mousa Masadeh, Aktham Al sarayreh *et al.*, "Innovation Through Intelligence: Mapping the Adoption of Artificial Intelligence with Bibliometric Methods", In *Artificial Intelligence, Sustainable Technologies, and Business Innovation: Opportunities and Challenges of Digital Transformation*, March 2025, pp 187-198, Published by Springer Nature, DOI: 10.1007/978-3-031-77925-1_17, Available: https://link.springer.com/chapter/10.1007/978-3-031-77925-1_17.

[58] Iyas Qaddara and Yousef Alraba'nah, "Enhancing Requirements Classification Using Machine Learning Techniques", *SN Computer Science*, Print ISSN: 2662-995X, Online ISSN: 2661-8907, Vol. 6, No. 6, 15 July 2025, art. 649, Published by Springer Nature, DOI: 10.1007/s42979-025-04158-z, Available: https://link.springer.com/article/10.1007/s42979-025-04158-z.

[59] Areen Arabiat, "Intelligent model for detecting GAN-generated images based on multi-classifier and advanced data mining techniques", *International Journal of Electrical and Electronic Engineering & Telecommunications*, Online ISSN: 2317-2518, Vol. 14, No. 3, May 2025, pp. 147–157, Published by Engineering and Technology Publishing, DOI: 10.18178/ijeetc.14.3.147-157, Available: https://www.ijeetc.com/show-249-1887-1.html.

[60] Ahmad Al-Qerem, Ali Mohd Ali, Issam Jebreen, Ahmad Nabot, Mohammed Rajab *et al.*, "Feature selection in socio-economic analysis: A multi-method approach for accurate predictive outcomes", *International Journal of Crowd*

Science, Print ISSN: 2398-7294, Vol. 9, No. 1, January 2025, pp. 64–78, Published by Tsinghua University Press, DOI: 10.26599/ijcs.2023.9100035, Available: https://www.sciopen.com/article/pdf/10.26599/IJCS.2023.9100035.pdf.

[61] Gulnara Bektemyssova, Zeinab Montazeri, Mohammad Dehghani, Ibraheem Kasim Ibraheem, Aseel Smerat *et al.*, "Antarctic Ice Worm Algorithm: A Novel Nature-inspired Metaheuristic for Optimization Tasks", *International Journal of Intelligent Engineering & Systems*, Print ISSN: 2185-310X, Online ISSN: 2185-3118, Vol. 18, No. 10, 30 July 2025, pp. 318-332, Published by Intelligent Network and Systems Society, DOI: 10.22266/ijies2025.1130.21, Available: https://oaji.net/articles/2023/3603-1760662783.pdf.

[62] Saifeddin Alghlayini, Mohammed Azmi Al-Betar and Mohammed Atef, "Enhancing non-invasive blood glucose prediction from photoplethysmography signals via heart rate variability-based feature selection using metaheuristic algorithms", *Algorithms*, Online ISSN: 1999-4893, Vol. 18, No. 2, 8 February 2025, art. 95, Published by MDPI, DOI: 10.3390/a18020095, Available: https://www.mdpi.com/1999-4893/18/2/95.

[63] Mojtaba Ghasemi, Mohammad Amin Akbari, Mohsen Zare, Seyedali Mirjalili, Mohamed Deriche *et al.*, "Birds of prey-based optimization (BPBO): a metaheuristic algorithm for optimization", *Evolutionary Intelligence*, Print ISSN: 1864-5909, Online ISSN: 1864-5917, Vol. 18, No. 88, 29 July 2025, Published by Springer Nature, DOI: 10.1007/s12065-025-01052-8, Available: https://link.springer.com/article/10.1007/s12065-025-01052-8.

[64] Miriam Ugarte, Pablo Valle, Miren Illarramendi and Aitor Arrieta, "Enhancing multi-objective test case selection through the mutation operator", *Automated Software Engineering*, Print ISSN 0928-8910, Online ISSN: 1573-7535, Vol. 32, No. 1, 30 January 2025, Published by Springer Nature, DOI: 10.1007/s10515-025-00489-6, Available: https://link.springer.com/article/10.1007/s10515-025-00489-6.

[65] Karol Durczak, Piotr Rybacki and Agnieszka Sujak, "Application of a Selected Pseudorandom Number Generator for the Reliability of Farm Tractors", *Applied Sciences*, Online ISSN: 2076-3417, Vol. 12, No. 23, November 2022, art. 12452, Published by MDPI, DOI: 10.3390/app122312452, Available: https://www.mdpi.com/2076-3417/12/23/12452.

[66] Bektemyssova Gulnara, Zeinab Montazeri, Mohammad Dehghani, Mohammad Kasim Ibraheem, Aseel Smerat *et al.*, "Search and Rescue Algorithm (SRA): A Metaheuristic Approach for Efficient Constrained Engineering Optimization", *International Journal of Intelligent Engineering and Systems*, Print ISSN: 2185-310X, Online ISSN: 2185-3118, Vol. 18, No. 10, September 2025, pp. 607-621, Published by Intelligent Network and Systems Society, DOI: 10.22266/ijies2025.1130.39, Available: https://oaji.net/articles/2023/3603-1760664579.pdf.

[67] Jie Wang, Xuanrui Xiong, Gaosheng Chen, Ruiqi Quyang, Yunli Gao *et al.*, "Multi-Criteria Feature Selection Based Intrusion Detection for Internet of Things Big Data", *Sensors*, Print ISSN: 1424-8220, Vol. 23, No. 17, 25 August 2023, art. 7434, Published by MDPI, DOI: 10.3390/s23177434, Available: https://www.mdpi.com/1424-8220/23/17/7434.

[68] Mohammed M. Alani and Ali Miri, "Towards an Explainable Universal Feature Set for IoT Intrusion Detection", *Sensors*, Print ISSN: 1424-8220, Vol. 22, No. 15, 29 July 2022, art. 5690, Published by MDPI, DOI: 10.3390/s22155690, Available: https://www.mdpi.com/1424-8220/22/15/5690.

[69] Yousef Sanjalawe, Salam Fraihat, Salam Al-E'mari, Mosleh Abualhaj, Sharif Makhadmeh *et al.*, "Smart load balancing in cloud computing: Integrating feature selection with advanced deep learning models", *PLOS ONE*, Online ISSN: 1932-6203, Vol. 20, No. 9, 9 Sep. 2025, Art. no. e0329765, Published by Public Library of Science, DOI: 10.1371/journal.pone.0329765, Available: https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0329765.

[70] Shangying Guo and Jing Zhao, "Investigating the accuracy of the GPT-2 algorithm in classifying identified targets for an intelligent virtual assistant", *Annals of Emerging Technologies in Computing (AETiC)*, Print ISSN: 2516-0281, Online ISSN: 2516-029X, Vol. 9, No. 3, July 2025, pp. 22-32, Published by International Association for Educators and Researchers (IAER), DOI: 10.33166/AETiC.2025.03.002, Available: https://aetic.theiaer.org/archive/v9/v9n3/p2.htm.

[71] Long Yu, Jiarui Dai, Jiaqi Dai and Yanan Wang, "Deep Learning and Transformers Accuracy in Rumor Detection on Social Media", *Annals of Emerging Technologies in Computing (AETiC)*, Print ISSN: 2516-0281, Online ISSN: 2516-029X, Vol. 9, No. 3, July 2025, pp. 33-49, Published by International Association for Educators and Researchers (IAER), DOI: 10.33166/AETiC.2025.03.003, Available: https://aetic.theiaer.org/archive/v9/v9n3/p3.html.

[72] Mohammad Riyaz Belgaum, Harsha Charitha, Harini Munurathi, Bylla Anusha, Ala Jayasri Sai *et al.*, "Enhancing the efficiency of diabetes prediction through training and classification using PCA and LR model", *Annals of Emerging Technologies in Computing (AETiC)*, Print ISSN: 2516-0281, Online ISSN: 2516-029X, Vol. 7, No. 3, 1st July 2023, pp. 78-91, Published by International Association for Educators and Researchers (IAER), DOI: 10.33166/AETiC.2023.03.004, Available: https://www.researchgate.net/publication/371970187.