Computers, Materials & Continua
DOI: 10.32604/cmc.2023.033273
Article

Network Intrusion Detection Based on Feature Selection and Hybrid Metaheuristic Optimization

Reem Alkanhel[1], El-Sayed M. El-kenawy[2], Abdelaziz A. Abdelhamid[3,4], Abdelhameed Ibrahim[5], Manal Abdullah Alohali[6], Mostafa Abotaleb[7] and Doaa Sami Khafaga[8,*]

[1]Department of Information Technology, College of Computer and Information Sciences, Princess Nourah bint Abdulrahman University, P.O. Box 84428, Riyadh, 11671, Saudi Arabia
[2]Department of Communications and Electronics, Delta Higher Institute of Engineering and Technology, Mansoura, 35111, Egypt
[3]Department of Computer Science, Faculty of Computer and Information Sciences, Ain Shams University, Cairo, 11566, Egypt
[4]Department of Computer Science, College of Computing and Information Technology, Shaqra University, 11961, Saudi Arabia
[5]Computer Engineering and Control Systems Department, Faculty of Engineering, Mansoura University, Mansoura, 35516, Egypt
[6]Information Systems Department, College of Computer and Information Sciences, Princess Nourah bint Abdulrahman University, P.O. Box 84428, Riyadh, 11671, Saudi Arabia
[7]Department of System Programming, South Ural State University, Chelyabinsk, 454080, Russia
[8]Department of Computer Sciences, College of Computer and Information Sciences, Princess Nourah bint Abdulrahman University, P.O. Box 84428, Riyadh, 11671, Saudi Arabia
*Corresponding Author: Doaa Sami Khafaga. Email: dskhafga@pnu.edu.sa
Received: 13 June 2022; Accepted: 01 August 2022

Abstract: Applications of internet-of-things (IoT) are increasingly being used in many facets of our daily life, which results in an enormous volume of data. Cloud computing and fog computing, two of the most common technologies used in IoT applications, have led to major security concerns. Cyberattacks are on the rise as a result of the usage of these technologies since present security measures are insufficient. Several artificial intelligence (AI) based security solutions, such as intrusion detection systems (IDS), have been proposed in recent years. Intelligent technologies that require data preprocessing and machine learning algorithm-performance augmentation require the use of feature selection (FS) techniques to increase classification accuracy by minimizing the number of features selected. On the other hand, metaheuristic optimization algorithms have been widely used in feature selection in recent decades. In this paper, we proposed a hybrid optimization algorithm for feature selection in IDS. The proposed algorithm is based on grey wolf (GW), and dipper throated optimization (DTO) algorithms and is referred to as GWDTO. The proposed algorithm has a better balance between the exploration and exploitation steps of the optimization process and thus could achieve better performance. On the employed IoT-IDS dataset, the performance of the proposed GWDTO algorithm was assessed using a set of evaluation metrics and compared to other optimization approaches in

the literature to validate its superiority. In addition, a statistical analysis is performed to assess the stability and effectiveness of the proposed approach. Experimental results confirmed the superiority of the proposed approach in boosting the classification accuracy of the intrusion in IoT-based networks.

Keywords: Feature selection; grey wolf optimization; dipper throated optimization; intrusion detection; internet-of-things (IoT)

1 Introduction

New business process innovations are being pushed forward by the Internet of Things (IoT), a network of interconnected computers and objects that can communicate and engage with one another. Individuals and enterprises confront a wide range of difficulties linked to their reputation, funding, and company operations as the frequency of cybersecurity attacks on IoT equipment grow fast and extensively [1]. In the cloud computing paradigm, consumers may access a wide range of services and resources on-demand, with little effort from either the service provider or the client [2]. The vast majority of IoT applications rely on the cloud to store and analyze their data. Due to the vast volumes of data that are kept in the cloud, security is a huge problem. Many factors have contributed to an increase in cyberattacks on cloud computing: the availability and accessibility of hacking tools have made it easier for hackers to carry out an attack, requiring no advanced knowledge or special skills [3]. Due to the increasing need for cybersecurity development, corporations and academic institutions throughout the world have been paying particular attention. Despite the adoption of a range of security measures, such as firewalls, antivirus software, encryption of critical data, and biometric authentication of end-users, cyberattacks continue to plague organizations and businesses. An attack on a computer can result in the disclosure of sensitive information if it is carried out by someone who knows how to exploit a system's weaknesses. IoT systems' confidentiality, integrity, and availability are always under threat from such attacks.

There are several ways to secure IoT devices from a wide range of threats, but one of the most successful is using intrusion detection systems (IDS). When a malicious attack is detected, an IDS can take immediate action to stop it from spreading throughout the network [4–6]. IDS may be divided into two categories based on their detection mechanisms: anomaly and misuse detection. Analyzing variations from normal profile activity helps anomaly detection pick up on potentially dangerous activities. These IDS have a high rate of false positives (FPs), but they are better at spotting new and novel attacks. In contrast, misuse detection is able to distinguish between legitimate and harmful activity based on previously recognized patterns [7]. They can identify known attacks with certainty; however, these IDS cannot tell the difference between new and previously known attacks. The accuracy of IDS might be improved with the use of machine learning (ML) approaches, although this has yet to be accomplished [8–10]. An efficient IDS must be able to distinguish between valid and malicious network traffic, as well as detect the sort of attack that is taking place on the protected system while analyzing network data. For ML, the vast diversity of attack methods and network traffic properties increases the computational and temporal complexity of the issue search space [11].

In order to enhance the description of patterns in various classes, feature selection (FS) is a strategy that removes unnecessary and redundant features and selects the optimal subset of features. Wrappers and filters in the FS selection process are classified according to the application of learning algorithms. Filter algorithms use an independent criterion to analyze the link between a collection of features (such as information measures, distance measures or consistency measures). On the other hand, wrapper

algorithms employ unique algorithms to evaluate the value of a certain collection of attributes. If you're looking at classification accuracy, wrapper strategies outperform filter approaches since the proposed subset of features is reviewed using feedback from the learning process. They are, however, computationally more expensive than filters since their effectiveness is so heavily reliant on the learning process. Finding the nearly optimum collection of features is a key consideration while creating an FS algorithm. Traditional approaches such as breadth searches, depth searches, and others are infeasible when it comes to picking the optimal subset of attributes in large datasets; neural networks (NN), for example, are wrapper-based techniques that need 2 N separate subsets of a dataset with N features, each of which must be constructed and evaluated separately [12]. As a result, FS is considered an NP-hard optimization issue. Selecting a small number of features while yet maintaining a high level of accuracy is the primary focus of the algorithm's goal function. A multi-objective optimization problem or a single-objective optimization problem might be devised in order to obtain trade-off solutions between the two opposing purposes, as is commonly the case in feature selection literature.

Metaheuristics algorithms have recently shown good performance in a range of optimization contexts because of their dynamic search behavior and global search capabilities [13]. Particle swarm optimization (PSO) [14], grey wolf optimizer (GWO) [15], harmony search (HS) [16], seagull optimization algorithm (SOA) [17], multiverse optimization (MVO) [18], bowerbird optimization (SBO) [19], and dipper throated optimization (DTO) [20] are a few examples of how it has been frequently employed in the literature to give acceptable solutions to FS tasks. However, all metaheuristic optimization algorithms must balance the exploration and exploitation stages in order to avoid getting stuck in local optima or failing to converge. Randomness abounds in the search for solutions in metaheuristic algorithms, and this is to blame for the difficulties encountered. A combination of ideas from different scientific fields is required in order to solve this problem. Through the process of hybridization, it is possible to construct an algorithm with enhanced speed and accuracy by combining the best features of many existing approaches.

Hybrid algorithms outperformed single-algorithm approaches in the study of literature. It's still true that no algorithm is better than any other in every feature selection situation, according to the no-free-lunch (NFL) theory [21]. As a result, new algorithms need to be developed, or existing algorithms need to be enhanced by making modifications to their operators in order to better cope with feature selection challenges. A novel method to improve the performance of GWO metaheuristic algorithm has been presented in this paper based on DTO algorithm. In the proposed approach, DTO is used to alter the search operators of GWO to improve its performance through this hybridization. When compared to other well-known metaheuristic algorithms, the original GWO authors said that it is a very competitive method. However, GWO suffers from local optima issue stagnation and early convergence, much as other meta-heuristic algorithms. Therefore the proposed approach could overcome this limitation by improving the exploration and exploitation steps through the application of the DTO algorithm. On high-dimensional optimization problems, the proposed GWDTO algorithm surpassed all other competing algorithms in terms of solution quality and resilience, according to the results achieved.

2 Literature Review

Swarm intelligence-based evolutionary optimization methods were developed by Gauthama et al. in 2017 [22]. To tackle the feature selection problem, the swarm algorithms have been used in several research because of their ease of use and rapid convergence. Local optima and demographic variables are some of the limitations of this strategy. The upshot is that a number of studies have merged

swarm optimization with other algorithms to improve its efficiency and apply it to FS challenges. The swarm-based hybrid FS approach proposed by authors in [23] uses a local search strategy and is based on particle swarm optimization in which the local search approach is used to choose the subset of features that are least correlated and most important. Using correlation information, the particle swarm optimization search procedure aims to choose distinctive features with the help of the local search approach. On thirteen benchmark datasets it was evaluated against five current approaches for selecting features. For intelligent facial expression identification issues, the authors presented a micro-genetic algorithm integrated PSO feature selection strategy. Gaussian mutations in the equation for updating the particle's velocity were added to the original PSO approach to prevent premature convergence. For a successful global and local search, the technique for updating velocity is also dependent on the average user's experience. Using the K-nearest neighbors (kNN) classifier, the feature selection algorithms are usually tested in terms of benchmark datasets.

The Swarm intelligence algorithm is based on grasshopper foraging and flocking behavior. Genetic optimization algorithm (GOA) is a swarm intelligence algorithm in which the combination with the mutation operator results in a binary hybrid algorithm, as presented by authors in [24]. There are a number of functions that may be used for this purpose, such as the transfer function. In addition, a mutation operator with a realistic mutation rate was used to give a variety of options. GOA and evolutionary operators were integrated in the same way by authors in [25], in which an updated GOA with new evolutionary-based operators for constructing an efficient wrapper FS approach is presented. UCI provided the data sets used to test the suggested methodologies. In order to surpass other optimizers and locate the best solutions with higher convergence trends, we implemented the selection mechanism. As a solution to global optimization difficulties, the authors in [26] presented the Sine Cosine Algorithm (SCA), which makes use of the sine and cosine functions. Another population diversification technique known as a disruption operator was created by authors in [27] as a mix of SCA and Salp Swarm Algorithm (SSA). Therefore, when SSA and SCA were employed in tandem to create a combined population of prospective solution candidates, the quality of the solutions was not stagnated in any way. The results were encouraging when applied to feature selection issues for datasets with feature sizes ranging from 13 to more than 11,000. Authors in [28] have created a novel hybridization method based on SCA. After converting standard PSO to binary variations, the authors included SCA to further explore the datasets they had available. Clustering problems were solved using the k-means approach for seven datasets of 9 to more than 11,000 features, using ten typical benchmark test functions as the first testing. By including the SCA's position update equation, the suggested technique modifies the PSO's velocity equation. Additionally, the PSO's weighting factor was recalculated depending on the number of runs. A selection of iterations with the highest inertia weight was selected in order to improve the searchability of long-distance locations. Researchers found that using statistical t-tests, they were able to improve clustering accuracy significantly above previous natural-inspired optimization approaches. A novel SCA/Antlion Optimization (ALO) hybrid was introduced by authors in [29]. Random elements were also included in the position update equations to expand the diversity of people in each region. In addition, the Mayfly Algorithm (MA) is a novel algorithm based on the flying and mating behavior of mayflies [30]. MA and harmony search (HS) algorithms were suggested by authors in [31] for the feature selection difficulty. This approach was further developed by the HS after being acquired by the MA from various search locations. HS and MA were therefore combined only for the aim of enhancing the search intensification technique. Eighteen classification datasets were examined using the MA-HS method, with improved results.

3 Methodology

The proposed network intrusion identification system is based on feature selection applied through the proposed GWDTO optimization algorithm in order to identify attacks in RPL-based IoT networks. Fig. 1 depicts the proposed architecture. The design of the proposed approach includes data collection, analysis, and detection. The data collecting system is made up of sensors, traffic repositories, and sniffers. Using a sniffer, the 6LoWPAN network can monitor every packet that passes across it. Sensor events and packets that have been sniffed and forwarded can be accessed in the repository of sensor events/traffic collection. After that, the most useful features in the dataset are selected using the previously discussed feature selection strategy. To raise a warning if an attack is detected, the detection system has an alarm/attack notification module. Besides providing log reports to the user interface, it also analyzes traffic on a regular basis and gives log reports.

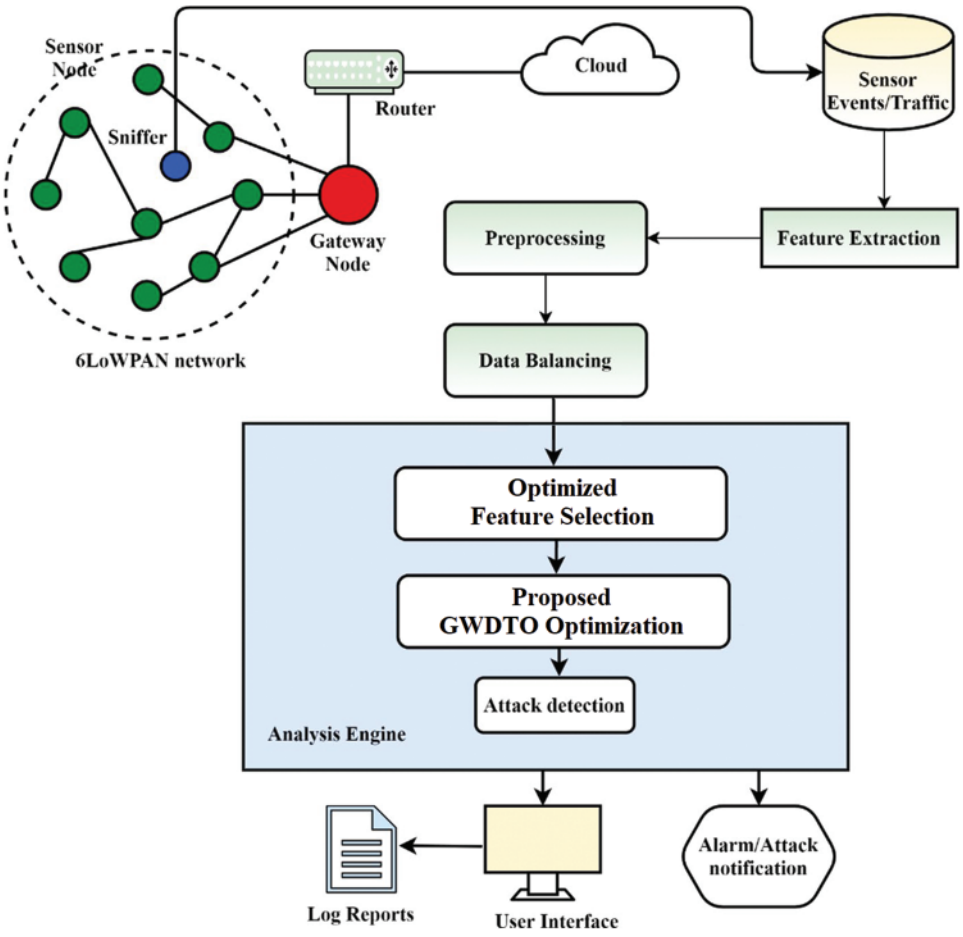


Figure 1: The proposed architecture of network intrusion detection

3.1 Dataset Collection

The RPL-NIDDS17 dataset is used to train the proposed approach [32]. The NetSim program was used to generate this data collection. NetSim is a widely-used program for simulating many kinds of network environments. A gateway, sensor nodes, a wired node, and a router make up the Internet

of Things network used to construct the dataset. Detailed information about each assault is stored in a separate CSV file. All the CSV files are combined into a single dataset. It is possible to label a total of 20 features in this dataset using time, basic, and flow characteristics. Routing attacks include Sybil, blackhole, sinkhole, and clone intrusion detection in addition to standard traffic patterns, as do hello flooding and selective forwarding. Only 33,337 routing attacks and 431,981 normal traffic are included in this dataset. Data is uneven as a result of the imbalance.

3.2 Dataset Preprocessing

Cleaning up data is a vital first step in the preparation process. This procedure includes removing duplicates, completing any gaps in the data, and encoding it. Despite the fact that machines can only read numeric data, the dataset contains both numeric and nominal information. As a consequence, the characters in the dataset are encoded into numerical values. Data scaling is used to speed things up after all of this. In terms of both size and unit, the dataset comprises a wide variety of attributes. Scaling is an option if you wish to maintain the integrity of your data over time. Another important step in data preprocessing is the dataset balancing, in which the number of samples in the dataset is equivalent for all classes. In this work, we adopted the locality-sensitive hashing and synthetic minority oversampling technique to achieve this goal. Tab. 1 presents the number of samples in the dataset before and after balancing.

Table 1: Dataset preprocessing using LSH-SMOTE balancing

Category	Total instances in dataset	Utilized instances	Using LSH-SMOTE
Normal	4,31,981	1,33,348	1,33,348
Attack	33,337	33,337	1,33,348

3.3 Dipper Throated Optimization

Dipper throated optimization (DTO) is based on tracking the locations and speeds of swimming and flying birds to simulate the genuine process of seeking food. Swimming birds' positions and speeds are updated using these equations.

$$BL_{nd}(t+1) = BL_{best}(t) - C_1.|C_2.BL_{best}(t) - BL_{nd}(t)| \tag{1}$$

where $BL_{nd}(t)$ and $BL_{best}(t)$ are the normal location and best location of the bird at iteration t, and C_1 and C_2 are adaptive values whose values are changed during the optimization process based on the iteration number and random values. The update of the flying bird's location is performed using the following equation.

$$BS(t+1) = C_3 BS(t) + C_4 r_1 (BL_{best}(t) - BL_{nd}(t)) + C_5 r_1 (BL_{Gbest} - BL_{nd}(t)) \tag{2}$$

$$BL_{nd}(t+1) = BL_{nd}(t) + BS(t+1) \tag{3}$$

where $BS(t+1)$ is the updated speed of each bird, r_1 is a random number in [0; 1], BL_{Gbest} is the global best location, and C_3 is a weight value, C_4 and C_5 are constants.

3.4 Grey Wolf Optimization

An alpha, beta, or omega wolf is regarded as a subpar wolf in the optimization of the Grey Wolf (or, according to some references, delta). The omega is ruled by delta wolves, who are superior to the

alphas and betas. Scouts, elders, and hunters make up this group. They rely on the hunters to help them search for food, and the hunters give the group food as a result. The security of the organization is the responsibility of the Sentinels. The primary duty of a scout is to keep an eye out for any dangers to the group's area and notify the rest of the unit accordingly. The wolves that have served as alpha or beta in the past are known as the pack's Elders. Even more intriguing than the social organization of grey wolves is the way they hunt as a group. As previously reported, the grey wolves are encircling their prey. Encircling behavior may be modeled using the following equations:

$$\vec{F}(t+1) = \vec{F}_p(t) - \vec{A}.\vec{D} \tag{4}$$

$$\vec{D} = \left|\vec{C}.\vec{F}_p(t) - \vec{F}(t)\right| \tag{5}$$

where $\vec{A}$ and $\vec{C}$ are vectors of coefficients, t represents the current iteration, $\vec{F}$ is the grey wolf's position vector, and $\vec{F}_p$ indicates the position vector of the prey. If there is a better solution in each iteration, $\vec{F}$ is updated to the best solution.

$$\vec{a} = 2 - t\left(\frac{2}{Max_{iter}}\right) \tag{6}$$

$$\vec{A} = 2\vec{a}.\vec{r}_1 - \vec{a} \tag{7}$$

$$\vec{C} = 2\vec{r}_2 \tag{8}$$

Loop counter t, Max_{iter} is the maximum number of iterations, $\vec{r}_1$ and $\vec{r}_2$ are random vectors in [0, 1], and $\vec{a}$ is linearly decreasing from 2 to 0 throughout the length of iterations. Consider a two-dimensional position vector and some of the potential neighbors depicted in Fig. 2 as a starting point for exploring the impact of Eqs. (1) and (2).

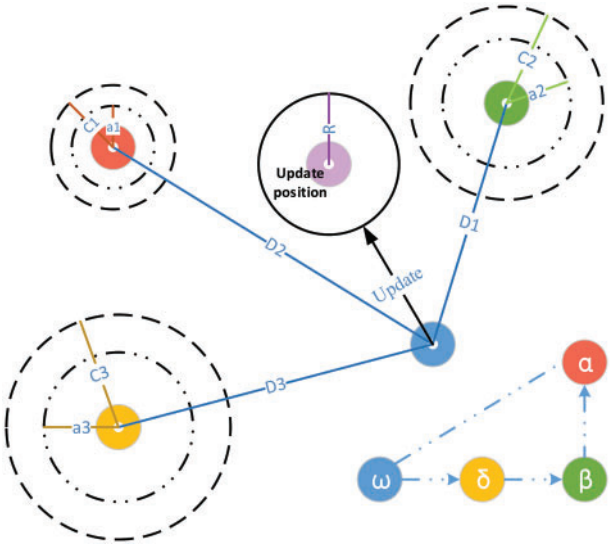


Figure 2: The hunting process of grey wolf optimization

Grey wolves may track down their victim and then completely engulf it. In most hunts, the alpha is in command of everything. During a hunting trip, the beta and delta occasionally join into the process.

However, the location of the prey within the 2D search region we have created (optimum) is unknown. A statistical model of gray wolf hunting behavior will assume that all three wolves are aware of where to seek prey. Consequently, we will maintain the top three search results and force all of the grey wolves (including omegas) to recalculate their positions based on them. Here are the formulas we devised to do this:

$$\vec{D}_{\alpha} = \left|\vec{D}_1 * \vec{F}_{\alpha} - \vec{F}\right|, \quad \vec{D}_{\beta} = \left|\vec{D}_2 * \vec{F}_{\beta} - \vec{F}\right|, \quad \vec{D}_{\delta} = \left|\vec{D}_3 * \vec{F}_{\delta} - \vec{F}\right| \tag{9}$$

$$\vec{F}_1 = \vec{F}_{\alpha} - \vec{A}_1 * \vec{D}_{\alpha}, \quad \vec{F}_2 = \vec{F}_{\beta} - \vec{A}_2 * \vec{D}_{\beta}, \quad \vec{F}_3 = \vec{F}_{\delta} - \vec{A}_3 * \vec{D}_{\delta} \tag{10}$$

$$\vec{F}(t+1) = \frac{\vec{F}_1 + \vec{F}_2 + \vec{F}_3}{3} \tag{11}$$

A grey wolf (a search agent) is seen in Fig. 2 moving around in a 2D search space according to alpha, beta, and delta. Fig. 2 shows that the ultimate location of a grey wolf (search agent) will be in a random location inside the search area given by the coordinates of alpha, beta, and delta. Prey's location is estimated by alpha, beta, and delta wolves; other wolves follow this guess and update their locations around the prey at random.

3.5 The Proposed Feature Selection Approach

The proposed GWDTO employs a KNN classifier to make sure that only decency-preserving features are used. Due to its primary goal of maximizing classification accuracy while minimizing the number of features picked and the error rate, the feature selection method employs the fitness function expressed by Eq. (12) for evaluating the performance of the proposed algorithm during the optimization process [33–39].

$$Fitness = \alpha\gamma_R(D) + \beta\frac{|C-R|}{|C|} \tag{12}$$

In which the condition attribute set R in relation to the decision D has a classification quality denoted by $\gamma_R(D)$. The total number of features is denoted by C, and the number of selected features is referred to as R. Additional parameters that affect the classification accuracy are $\alpha \in [0, 1]$ and $\beta = 1 - \alpha$. Based on the error rate and selected feature ratio, Eq. (1) can be converted into a minimization problem. Therefore, the minimization of the error rate can be performed using the following fitness function, where $E_R(D)$ is the error rate.

$$Fitness = \alpha E_R(D) + \beta\frac{|R|}{|C|} \tag{13}$$

To select the best set of features, the resulting best solution is converted into binary 0 or 1. To perform this conversion, the sigmoid function is usually employed as represented by the following equation where S_{best} refers to the best position at iteration t.

$$S_d(t+1) = \begin{cases} 0, & if\ Sigmoid(S_{best}) < 0.5 \\ 1, & otherwise \end{cases} \tag{14}$$

Algorithm 1: The proposed GWDTO algorithm

Initialize birds locations BL_i $(i = 1, 2, 3, \ldots, n)$ with size n, BS_i $(i = 1, 2, 3, \ldots, n)$,
Fitness function Fn, f_n, r_1, r_2, r_3, R, C_1, C_2, C_3, C_4, C_5, $t=1$, and max iterations $iter\_max$
Evaluate fitness function Fn for each BL_i
Find best bird BL_{best}
While t < $iter\_max$ **do**
    for ($i = 1$; $i \leq n$) **do**
        If (R < 0.5) **then**
            Update Location of the gey wolf agents using:
            $\vec{D}_\alpha = \left|\vec{D}_1 * \vec{F}_\alpha - \vec{F}\right|$, $\vec{D}_\beta = \left|\vec{D}_2 * \vec{F}_\beta - \vec{F}\right|$, $\vec{D}_\delta = \left|\vec{D}_3 * \vec{F}_\delta - \vec{F}\right|$
        else
            Update Speed of the flying bird using:
            $BS(t+1) = C_3 BS(t) + C_4 r_1 \left(BL_{best}(t) - BL_{nd}(t)\right) + C_5 r_1 (BL_{Gbest} - BL_{nd}(t))$
            Update Location of the swimming bird using:
            $BL_{nd}(t+1) = r_1 + z * r_2 + (1 - z) * r_3 + BS(t+1)$
        end for
    end for
    Evaluate fitness function Fn for each $\overrightarrow{BL_i}$
    Update R, r_1, r_2, r_3, c, C_1, C_2
    Find best bird BL_{best}
    Set $BL_{Gbest} = BL_{best}$
    Set t = t + 1
end while
return BL_{Gbest}

4 Experimental Results

A Windows 11 laptop with an Intel Core i5 CPU clocked at 2.33 GHz, and 16GB of RAM is used for the tests. In order to develop and evaluate the proposed framework, MATLAB R2020a was employed. MATLAB's Text Analytics Toolbox is utilized to do preprocessing on the dataset. This section evaluates and compares the performance of the proposed approach with other competing approaches.

4.1 Evaluation Metrics

The evaluation metrics employed in this research to assess the performance of the feature selection algorithm are presented in Tab. 2. These metrics include average fitness size, average error, standard deviation, worst, best, and average fitness. These metrics are used to evaluate the performance of feature selection methods [40–49].

Table 2: Evaluation metrics

Metrics		Equation
Average error	=	$\frac{1}{M}\sum_{j=1}^{M}\frac{1}{N}\sum_{i=1}^{N} mse(C_i\ ,\ L_i)$
Average fitness	=	$\frac{1}{M}\sum_{i=1}^{M} g_*^i$
Average fitness size	=	$\frac{1}{M}\sum_{i=1}^{M} size\ \left(g_*^i\right)$
Best fitness	=	$Min_{i=1}^{M}\ g_*^i$
Worst fitness	=	$Max_{i=1}^{M}\ g_*^i$
STD (Standard Deviation) fitness	=	$\sqrt{\frac{1}{M-1}\ \sum\left(g_*^i - Mean\right)^2}$

4.2 Evaluation Results

The achieved results based on the proposed feature selection approach are presented in Tab. 3. The NIDS performance with and without the LSH-SMOTE algorithm is shown in Tab. 2. With 98.1% accuracy, the proposed approach is far more accurate than the approach of traditional LSH-SMOTE and without LSH-SMOTE. There are also greater gains in terms of LSH-SMOTE algorithm accuracy, recall, F-score, specificity, and sensitivity than the other models. Using the proposed LSH-SMOTE, the performance of the NIDS is greatly enhanced.

Table 3: Evaluation of the results achieved by the feature selection methods

	Avg. error	Avg. select size	Avg. fitness	Best fitness	Worst fitness	Avg. std
bGWDTO	0.1192	0.0720	0.1824	0.0842	0.1827	0.0047
bGWO	0.1364	0.2720	0.1986	0.1189	0.1858	0.0094
bGWO_PSO	0.1757	0.4053	0.2069	0.1604	0.2704	0.0276
bPSO	0.1702	0.2720	0.1970	0.1773	0.2450	0.0088
bBA	0.1798	0.4114	0.2199	0.1096	0.2112	0.0187
bWAO	0.1700	0.4354	0.2048	0.1689	0.2450	0.0110
bBBO	0.1384	0.4358	0.2027	0.1924	0.2789	0.0537
bMVO	0.1469	0.3685	0.2267	0.1519	0.2699	0.0595
bSBO	0.1785	0.4423	0.2367	0.1798	0.2595	0.0697
bGWO_GA	0.1565	0.1948	0.2047	0.1825	0.2587	0.0100
bFA	0.1686	0.3065	0.2489	0.1676	0.2652	0.0456
bGA	0.1500	0.2144	0.2100	0.1133	0.2284	0.0110

The proposed GWDTO algorithm is used to optimize the feature selection process, and the results are evaluated using the criteria mentioned in the previous section. The selected features are used to train a KNN classifier to measure the effectiveness of the proposed approach. Tab. 4 presents the statistical analysis of the achieved results using the proposed approach and other approaches. As presented in this table, the results achieved using the proposed approach outperform the other methods. The mean error is (0.119, whereas the minimum mean error using the other approaches is (0.136), which reflects the superiority of the proposed approach.

Table 4: Statistical analysis of the results achieved by the feature selection methods

	# Val.	Min	25%	Median	75%	Max	Range	Mean	Std	Std. err	Sum
bGWDTO	12.000	0.119	0.119	0.119	0.119	0.119	0.000	0.119	0.000	0.000	1.431
bGWO	12.000	0.126	0.136	0.136	0.136	0.146	0.020	0.136	0.004	0.001	1.637
bGWO_PSO	12.000	0.166	0.176	0.176	0.176	0.186	0.020	0.176	0.004	0.001	2.109
bPSO	12.000	0.160	0.170	0.170	0.170	0.180	0.020	0.170	0.004	0.001	2.043
bBA	12.000	0.170	0.180	0.180	0.180	0.190	0.020	0.180	0.004	0.001	2.158
bWAO	12.000	0.160	0.170	0.170	0.170	0.180	0.020	0.170	0.004	0.001	2.040
bBBO	12.000	0.118	0.138	0.138	0.138	0.148	0.030	0.137	0.007	0.002	1.641
bMVO	12.000	0.137	0.147	0.147	0.147	0.157	0.020	0.147	0.004	0.001	1.763
bSBO	12.000	0.169	0.179	0.179	0.179	0.189	0.020	0.179	0.004	0.001	2.142
bGWO_GA	12.000	0.147	0.157	0.157	0.157	0.167	0.020	0.157	0.004	0.001	1.878
bFA	12.000	0.159	0.169	0.169	0.169	0.179	0.020	0.169	0.004	0.001	2.023
bGA	12.000	0.130	0.150	0.150	0.150	0.160	0.030	0.149	0.007	0.002	1.790

Tabs. 5 and 6 show the findings of the Wilcoxon signed-rank test, on the other hand. As can be seen from the tables, the proposed strategy is statistically significant, just like the other strategies. Therefore, the suggested method is suited to the task of selecting features. Tab. 7 shows the results of a one-way analysis of variance (ANOVA) test for validating the stability and effectiveness of the proposed approach. These tests stress the statistical significance and efficacy of the suggested method based on the hypotheses of these tests.

Table 5: Wilcoxon signed rank test results (Part 1)

	bGWDTO	bGWO	bGWO_PSO	bPSO	bBA	bWAO
Theoretical median	0	0	0	0	0	0
Number of values	12	12	12	12	12	12
Actual median	0.1192	0.1364	0.1757	0.1702	0.1798	0.17
P value (two tailed)	0.0005	0.0005	0.0005	0.0005	0.0005	0.0005
Sum of signed ranks (W)	78	78	78	78	78	78
Sum of positive ranks	78	78	78	78	78	78
Sum of negative ranks	0	0	0	0	0	0
Discrepancy	0.1192	0.1364	0.1757	0.1702	0.1798	0.17
Exact or estimate?	Exact	Exact	Exact	Exact	Exact	Exact
Significant (alpha = 0.05)?	Yes	Yes	Yes	Yes	Yes	Yes

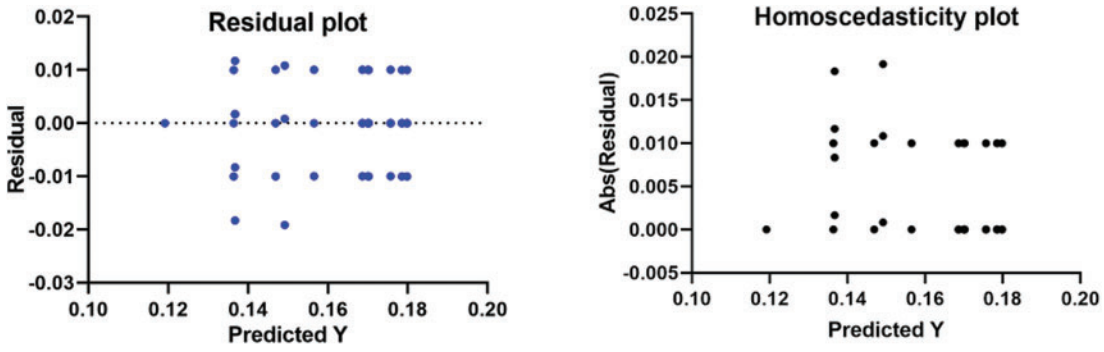
Table 6: Wilcoxon signed rank test results (Part 2)

	bBBO	bMVO	bSBO	bGWO_GA	bFA	bGA
Theoretical median	0	0	0	0	0	0
Number of values	12	12	12	12	12	12
Actual median	0.1384	0.1469	0.1785	0.1565	0.1686	0.15
P value (two tailed)	0.0005	0.0005	0.0005	0.0005	0.0005	0.0005
Sum of signed ranks (W)	78	78	78	78	78	78
Sum of positive ranks	78	78	78	78	78	78
Sum of negative ranks	0	0	0	0	0	0
Discrepancy	0.1384	0.1469	0.1785	0.1565	0.1686	0.15
Exact or estimate?	Exact	Exact	Exact	Exact	Exact	Exact
Significant (alpha = 0.05)?	Yes	Yes	Yes	Yes	Yes	Yes

Table 7: One-way analysis of variance test

ANOVA table	SS	DF	MS	F (DFn, DFd)	*P* value
Treatment (between columns)	0.05083	11	0.004621	F (11, 132) = 213.4	$P < 0.0001$
Residual (within columns)	0.002858	132	2.17E − 05		
Total	0.05369	143			

The attained outcomes are shown in Fig. 3 to demonstrate the approach's efficacy and superiority. It's easy to see that the proposed technique is highly accurate based on the data in this image because the residual error is so little. QQ, heatmaps, ROCs, and histogram plots are utilized to demonstrate the suggested method's efficiency. Plots like this demonstrate how superior the recommended strategy is to the alternatives.



(a) Residual plot (b) Homoscedasticity plot

Figure 3: (Continued)

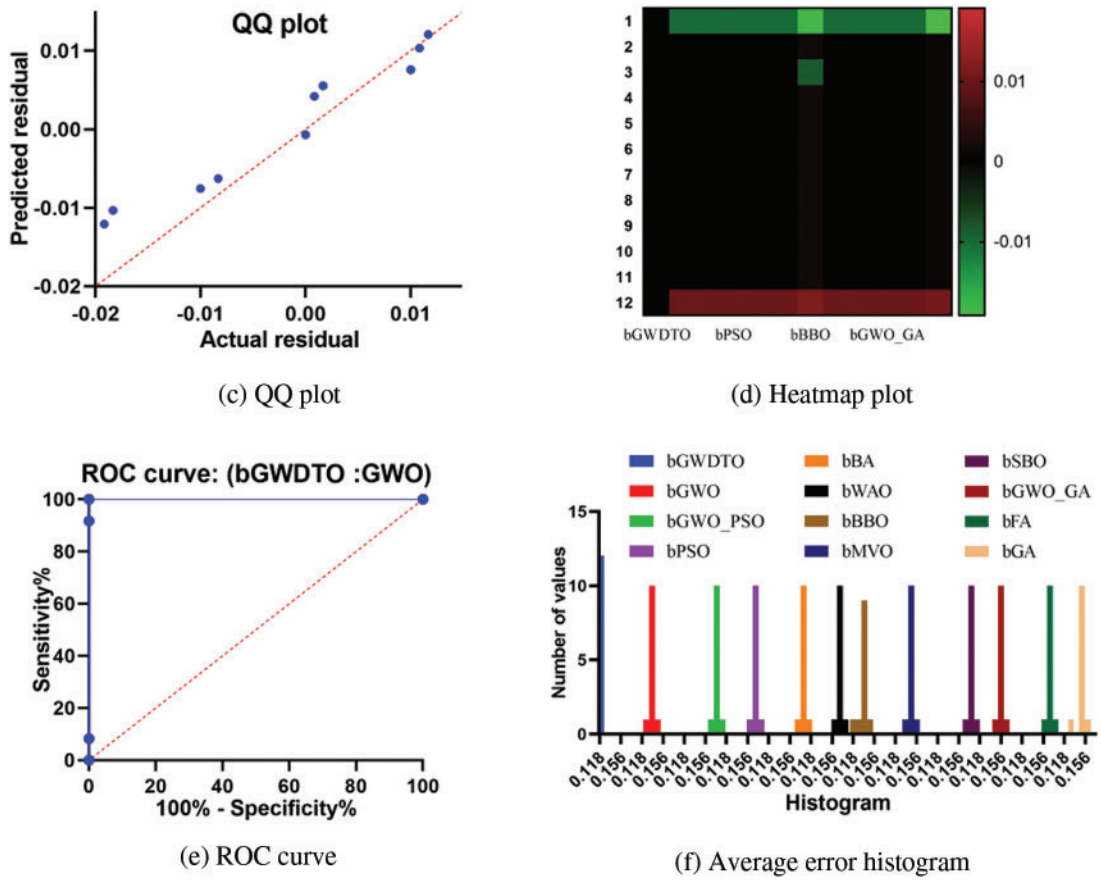


Figure 3: Visualization of the achieved results using the proposed feature selection approach

The ranges of the average error achieved by the proposed approach and other approaches are depicted in Fig. 4. In this figure, the proposed approach could achieve the smallest average error, which is better than the other approaches. These results emphasize the effectiveness and superiority of the proposed approach.

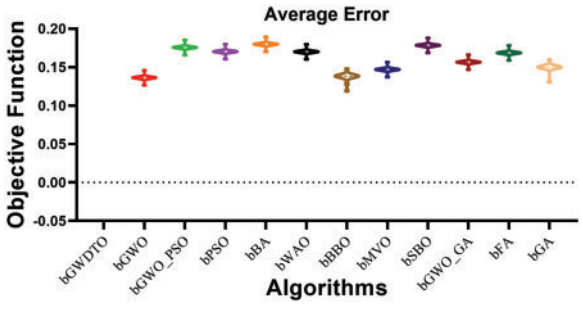


Figure 4: Histogram of the accuracy achieved by the proposed approach and other approaches

5 Conclusions

There is a large quantity of data generated by IoT applications because of their special nature. Furthermore, the safety and privacy of user data is jeopardized as a result of these applications. Machine learning (ML)-based security solutions, such as intrusion detection systems (IDS), have been introduced in recent years. ML algorithms are affected by the existence of duplicate or irrelevant data. A new feature selection (FS) approach referred to as GWDTO was developed to improve the efficiency of the GWO algorithm by utilizing the DTO algorithm. Adopting DTO, which has a high capacity to locate viable regions that give the optimum solution, improved GWO's performance. The results of the proposed algorithm were compared to other methods. The suggested GWDTO approach outperformed various current metaheuristic algorithms, including the original PSO, WOA, GWO, MVO, SBO, FA, and GA, according to the results of the testing. According to the RPL-NIDDS17 dataset, the proposed approach achieved a 1.18 average error. These results outperform those achieved by the other approaches, which confirm the superiority and effectiveness of the proposed approach.

Acknowledgement: Princess Nourah bint Abdulrahman University Researchers Supporting Project Number (PNURSP2022R323), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia.

Funding Statement: Princess Nourah bint Abdulrahman University Researchers Supporting Project Number (PNURSP2022R323), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] I. El-Hasnony, R. Mostafa, M. Elhoseny and S. Barakat, "Leveraging mist and fog for big data analytics in IoT environment," *Transactions of Emergence Telecommunication Technology*, vol. 32, pp. e4057, 2021.

[2] I. Lee, "Internet of things (IoT) cybersecurity: Literature review and IoT cyber risk management," *Future Internet*, vol. 12, pp. 157, 2020.

[3] G. Kushwah and V. Ranga, "Voting extreme learning machine based distributed denial of service attack detection in cloud computing," *Journal of Information Security Applications*, vol. 53, pp. 102532, 2020.

[4] P. Louvieris, N. Clewley and X. Liu, "Effects-based feature identification for network intrusion detection," *Neurocomputing*, vol. 121, pp. 265–273, 2013.

[5] O. Al-Jarrah, O. Alhussein, P. Yoo, S. Muhaidat, K. Taha *et al.,* "Data randomization and cluster-based partitioning for botnet intrusion detection," *IEEE Transactions on Cybernetics*, vol. 46, pp. 1796–1806, 2016.

[6] J. Ashraf, M. Keshk, N. Moustafa, M. Abdel-Basset, H. Khurshid *et al.,* "IoTBoT-IDS: A novel statistical learning-enabled botnet detection framework for protecting networks of smart cities," *Sustainable Cities Society*, vol. 72, pp. 103041, 2021.

[7] Y. Zhou, G. Cheng, S. Jiang and M. Dai, "Building an efficient intrusion detection system based on feature selection and ensemble classifier," *Computer Networks*, vol. 174, pp. 107247, 2020.

[8] K. Wang, M. Du, S. Maharjan and Y. Sun, "Strategic honeypot game model for distributed denial of service attacks in the smart grid," *IEEE Transactions on Smart Grid*, vol. 8, pp. 2474–2482, 2017.

[9] K. Wang, M. Du, Y. Sun, A. Vinel and Y. Zhang, "Attack detection and distributed forensics in machine-to-machine networks," *IEEE Networks*, vol. 30, pp. 49–55, 2016.

[10] K. Wang, M. Du, D. Yang, C. Zhu, J. Shen *et al.,* "Game-theory-based active defense for intrusion detection in cyber-physical embedded systems," *ACM Transactions on Embedded Computer Systems*, vol. 16, pp. 1–21, 2016.
[11] E. Hoz, A. Ortiz, J. Ortega and B. Prieto, "PCA filtering and probabilistic SOM for network intrusion detection," *Neurocomputing*, vol. 164, pp. 71–81, 2015.
[12] M. Du, K. Wang, Y. Chen, X. Wang and Y. Sun, "Big data privacy preserving in multi-access edge computing for heterogeneous internet of things," *IEEE Communication Magazine*, vol. 56, pp. 62–67, 2018.
[13] M. Du, K. Wang, Z. Xia and Y. Zhang, "Differential privacy preserving of training model in wireless big data with edge computing," *IEEE Transactions on Big Data*, vol. 6, pp. 283–295, 2018.
[14] P. Mishra, V. Varadharajan, U. Tupakula and E. Pilli, "A detailed investigation and analysis of using machine learning techniques for intrusion detection," *IEEE Communication Survey and Tutorials*, vol. 21, pp. 686–728, 2019.
[15] S. Aljawarneh, M. Aldwairi and M. Yassein, "Anomaly-based intrusion detection system through feature selection analysis and building hybrid efficient model," *Journal of Computer Science*, vol. 25, pp. 152–160, 2018.
[16] M. Ambusaidi, X. He, P. Nanda and Z. Tan, "Building an intrusion detection system using a filter-based feature selection algorithm," *IEEE Transactions on Computers*, vol. 65, pp. 2986–2998, 2016.
[17] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *Journal of Machine Learning Research*, vol. 3, pp. 1157–1182, 2003.
[18] B. Xue, M. Zhang, W. Browne and X. Yao, "Survey on evolutionary computation approaches to feature selection," *IEEE Transactions on Evolutionary Computing*, vol. 20, pp. 606–626, 2016.
[19] I. El-Hasnony, S. Barakat, M. Elhoseny and R. Mostafa, "Improved feature selection model for big data analytics," *IEEE Access*, vol. 8, pp. 66989–67004, 2020.
[20] M. Nguyen and K. Kim, "Genetic convolutional neural network for intrusion detection systems," *Future Generation Computer Systems*, vol. 113, pp. 418–427, 2020.
[21] D. Wolpert and W. Macready, "No free lunch theorems for optimization," *IEEE Transactions on Evolutionary Computing*, vol. 1, pp. 67–82, 1997.
[22] M. Gauthama, N. Somu, K. Kirthivasan, R. Liscano and V. Shankar, "An efficient intrusion detection system based on hypergraph-genetic algorithm for parameter optimization and feature selection in support vector machine," *Knowledge Based Systems*, vol. 134, pp. 1–12, 2017.
[23] S. Malhotra, V. Bali and K. Paliwal, "Genetic programming and K-nearest neighbour classifier based intrusion detection model," in *Proc. of the 2017 7th Int. Conf. on Cloud Computing, Data Science & Engineering-Confluence*, Noida, India, pp. 42–46, 2017.
[24] P. Ghosh, A. Karmakar, J. Sharma and S. Phadikar, "CS-PSO based intrusion detection system in cloud environment," in *Emerging Technologies in Data Mining and Information Security*, New York, NY, USA: Springer, pp. 261–269, 2019.
[25] J. Seth and S. Chandra, "MIDS: Metaheuristic based intrusion detection system for cloud using k-NN and MGWO," in *Proc. of the Int. Conf. on Advances in Computing and Data Sciences*, Dehradun, India, New York, NY, USA, Springer, pp. 411–420, 2018.
[26] S. RM, P. Maddikunta, M. Parimala, S. Koppu, T. Gadekallu *et al.,* "An effective feature engineering for DNN using hybrid PCA-GWO for intrusion detection in IoMT architecture," *Computer Communication*, vol. 160, pp. 139–149, 2020.
[27] M. Mayuranathan, M. Murugan and V. Dhanakoti, "Best features based intrusion detection system by RBM model for detecting DDoS in cloud environment," *Journal of Ambient Intelligence and Humanized Computing*, vol. 12, pp. 3609–3619, 2021.
[28] A. Ewees, R. Mostafa, R. Ghoniem and M. Gaheen, "Improved seagull optimization algorithm using lévy flight and mutation operator for feature selection," *Neural Computing Applications*, vol. 34, pp. 7437–7472, 2022.
[29] J. Del, E. Osaba, D. Molina, X. Yang, S. Salcedo-Sanz *et al.,* "Bio-inspired computation: Where we stand and what's next," *Swarm Evolutionary Computing*, vol. 48, pp. 220–250, 2019.

[30] B. Abdollahzadeh, F. Soleimanian and S. Mirjalili, "Artificial gorilla troops optimizer: A new nature-inspired metaheuristic algorithm for global optimization problems," *International Journal of Intelligent Systems*, vol. 36, pp. 5887–5958, 2021.
[31] X. Meng, X. Gao, L. Lu, Y. Liu and A. Zhang, "A new bio-inspired optimisation algorithm: Bird swarm algorithm," *Journal of Expert Theory and Artificial Intelligence*, vol. 28, pp. 673–687, 2016.
[32] A. Verma and V. Ranga, "Evaluation of Network Intrusion Detection Systems for RPL Based 6LoWPAN Networks in IoT," *Wireless Personal Communications*, vol. 108, pp. 1571–1594, 2019.
[33] E. -S. M. El-Kenawy, S. Mirjalili, A. Ibrahim, M. Alrahmawy, M. El-Said *et al.,* "Advanced meta-heuristics, convolutional neural networks, and feature selectors for efficient COVID-19 X-ray chest image classification," *IEEE Access*, vol. 9, no. 1, pp. 36019–36037, 2021.
[34] A. Abdelhamid and S. Alotaibi, "Optimized two-level ensemble model for predicting the parameters of metamaterial antenna," *Computers, Materials & Continua*, vol. 73, no. 1, pp. 917–933, 2022.
[35] A. Abdelhamid and S. R. Alotaibi, "Robust prediction of the bandwidth of metamaterial antenna using deep learning," *Computers, Materials & Continua*, vol. 72, no. 2, pp. 2305–2321, 2022.
[36] D. Sami Khafaga, A. Ali Alhussan, E. M. El-kenawy, A. E. Takieldeen, T. M. Hassan *et al.,* "Meta-heuristics for feature selection and classification in diagnostic breast cancer," *Computers, Materials & Continua*, vol. 73, no. 1, pp. 749–765, 2022.
[37] D. Sami Khafaga, A. Ali Alhussan, E. M. El-kenawy, A. Ibrahim, S. H. Abd Elkhalik *et al.,* "Improved prediction of metamaterial antenna bandwidth using adaptive optimization of LSTM," *Computers, Materials & Continua*, vol. 73, no. 1, pp. 865–881, 2022.
[38] E. -S. M. El-Kenawy, S. Mirjalili, F. Alassery, Y. Zhang, M. Eid *et al.,* "Novel meta-heuristic algorithm for feature selection, unconstrained functions and engineering problems," *IEEE Access*, vol. 10, pp. 40536–40555, 2022.
[39] A. Abdelhamid, E. -S. M. El-kenawy, B. Alotaibi, M. Abdelkader, A. Ibrahim *et al.,* "Robust speech emotion recognition using CNN + LSTM based on stochastic fractal search optimization algorithm," *IEEE Access*, vol. 10, pp. 49265–49284, 2022.
[40] N. Abdel Samee, E. M. El-Kenawy, G. Atteia, M. M. Jamjoom, A. Ibrahim *et al.,* "Metaheuristic optimization through deep learning classification of COVID-19 in chest X-ray images," *Computers, Materials & Continua*, vol. 73, no. 2, pp. 4193–4210, 2022.
[41] H. Nasser AlEisa, E. M. El-kenawy, A. Ali Alhussan, M. Saber, A. A. Abdelhamid *et al.,* "Transfer learning for chest X-rays diagnosis using dipper throated algorithm," *Computers, Materials & Continua*, vol. 73, no. 2, pp. 2371–2387, 2022.
[42] S. S. M. Ghoneim, T. A. Farrag, A. A. Rashed, E. -S. M. El-Kenawy and A. Ibrahim, "Adaptive dynamic meta-heuristics for feature selection and classification in diagnostic accuracy of transformer faults," *IEEE Access*, vol. 9, pp. 78324–78340, 2021.
[43] H. Hassan, A. I. El-Desouky, A. Ibrahim, E. -S. M. El-Kenawy and R. Arnous, "Enhanced QoS-based model for trust assessment in cloud computing environment," *IEEE Access*, vol. 8, no. 1, pp. 43752–43763, 2020.
[44] M. M. Eid, E. -S. M. El-Kenawy and A. Ibrahim, "A binary sine cosine-modified whale optimization algorithm for feature selection," in *4th National Computing Colleges Conf. (NCCC 2021)*, Taif, Saudi Arabia, pp. 1–6, 2021.
[45] E. -S. M. El-Kenawy, S. Mirjalili, S. S. M. Ghoneim, M. M. Eid, M. El-Said *et al.,* "Advanced ensemble model for solar radiation forecasting using sine cosine algorithm and newton's laws," *IEEE Access*, vol. 9, pp. 115750–115765, 2021.
[46] A. Salamai, E. -S. M. El-kenawy and A. Ibrahim, "Dynamic voting classifier for risk identification in supply chain 4.0," *Computers Materials & Continua*, vol. 69, no. 3, pp. 3749–3766, 2021.
[47] A. Ibrahim, S. Mirjalili, M. El-Said, S. S. M. Ghoneim, M. Al-Harthi *et al.,* "Wind speed ensemble forecasting based on deep learning using adaptive dynamic optimization algorithm," *IEEE Access*, vol. 9, pp. 125787–125804, 2021.

[48] K. Albulayhi, Q. A. Al-Haija, S. A. Alsuhibany, A. A. Jillepalli, M. Ashrafuzzaman *et al.,* "IoT intrusion detection using machine learning with a novel high performing feature selection method," *Applied Sciences*, vol. 12, no. 10, pp. 1–30, 2022.

[49] K. Albulayhi and F. T. Sheldon, "An adaptive deep-ensemble anomaly-based intrusion detection system for the internet of things," in *2021 IEEE World AI IoT Congress (AIIoT)*, Seattle, WA, USA, pp. 187–196, 2021.