Journal of Computing and Security

2020, Volume 7, Number 1 (pp. 63–79)

http://www.jcomsec.org

A Lightweight Anomaly Detection Model using SVM for WSNs in IoT through a Hybrid Feature Selection Algorithm based on GA and GWO

Azam Davahli [a], Mahboubeh Shamsi [b,*], Golnoush Abaei [c]

[a] *Department of Computer Engineering, Qom Branch, Islamic Azad University, Qom, Iran.*
[b] *Faculty of Electrical and Computer Engineering, Qom University of Technology, Qom, Iran.*
[c] *Faculty of Electrical, Computer, and Biomedical Engineering, Shahabdanesh University, Qom, Iran.*

ARTICLE INFO.

Article history:
Received: 22 October 2019
Revised: 10 February 2020
Accepted: 24 April 2020
Published Online: 10 July 2020

Keywords:
Wrapper Feature Selection, Metaheuristic Algorithms, Grey Wolf Optimizer (GWO), Genetic Algorithm (GA), Wireless Networks, Internet of Things (IoT), Anomaly Detection, Support Vector Machine (SVM).

ABSTRACT

As a result of an incredibly fast growth of the number and diversity of smart devices connectable to the internet, commonly through open wireless sensor networks (WSNs) in internet of things (IoT), the access of attackers to the network traffic in the form of intercepting, eavesdropping and rebroadcasting has become much easier. Anomaly or intrusion detection system (IDS) is an efficient security mechanism, however despite the maturity of anomaly detection technologies for wired networks, current technologies with high computational complexity are improper for resource-limited WSNs in IoT and they also fail to detect new WSN attacks. Furthermore, dealing with the huge amount of intrusion wireless traffic collected by sensors, causing slow detecting process, higher resource usage and inaccurate detection. Hence, considering WSN limitations for developing an IDS in IoT, establishes a significant challenge for security researchers. This paper proposes a new model to develop a support vector machine (SVM)-based lightweight IDS (LIDS) using combination concepts of genetic algorithm (GA) and mathematical equations of grey wolf optimizer (GWO) which is called GABGWO. The GABGWO through applying two new crossover and mutation operators tries to find the most relevant traffic features and eliminate worthless ones, in order to increase the performance of the LIDS. The performance of LIDS is evaluated using AWID real-world wireless dataset under two scenarios with and without using GABGWO. The results showed a promising behavior of the proposed GABGWO algorithm in choosing optimal traffics, decreasing the computational costs and providing high accuracies for LIDS. The hybrid algorithm is also compared to pure GA and GWO and other recent methods and it is found that its performance is better than them.

1 Introduction

A network of things that is equipped with sensors and try to exchange data commonly through open wireless sensor networks (WSNs) is called internet of things (IoT) [1]. Beside of many advantages which has been provided by IoT, these networks have some security challenges [2, 3] and face some new attacks [4, 5]. Anomaly detection system or intrusion detection system (IDS) is an efficient security technique that by gathering traffic data and analyzing them identifies the attacks [6–8]. The numerous existing IDS techniques in wired networks are not able to handle the zero-day wireless intrusions in IoT [9–11]. Additionally, most traditional IDSs have high computational costs and don't consider resource limitations in wireless networks. Therefore, the advancement of a suitable and lightweight IDS for WSNs in IoT is necessary [12, 13] such that it decreases computational costs and increases detection accuracies [14, 15].

Based on the analysis and detection technique there are two main IDSs methods including misuse and anomaly detection. Anomaly detection methods are the most popular intrusion detection techniques in which malicious behaviors are distinguished from normal behaviors. In anomaly detection methods, there is no need to database for saving malicious traffics and they are able to recognize new attacks unlike misuse-based methods [12, 16, 17]. These methods usually are developed by machine learning algorithms or classifiers such as support vector machine (SVM), Artificial neural network (ANN), k-nearest neighbor (KNN), and so on, which among them SVM is the most successful classifier in this area [15, 18, 19]. Nevertheless, pervasiveness IoT devices that connected to a Wi-Fi network result in generate a huge number of wireless traffic data, which some of them are worthless and redundant that not only increase the computational costs, but also decrease the performance of the IDS [9, 20].

A solution that can effectively solve the high dimensionality problem is feature selection (FS) mechanism. A FS mechanism through choosing the optimal features and eliminating the extra and useless features leads to dimensionality reduction [21, 22]. FS methods generally are divided to three categories, which are called filter, wrapper, and hybrid [22–24]. The filter techniques evaluate the chosen features based on the data characteristics while the wrapper and hybrid techniques exploit a classifier for assessing the chosen features unlike the filter [22–24]. The wrapper mechanisms have shown more successful accordingly are the most popular [22, 23, 25]. Investigating all possible feature subsets in large search space such as WSNs space makes FS to a NP-hard problem [26, 27], thus an effective global search technique is required to solve the FS problem optimally [26]. Nowadays metaheuristic optimization algorithms have shown satisfactory capabilities to handle FS problem [26, 28, 29]. These algorithms' efficiency is chiefly affected by their exploration and exploitation abilities [29–31].

Exploration is the process of looking for good solutions in the whole search space whereas exploitation is probing a limited region of the search space with the hope of improving the achieved solutions through exploration. Therefore, a metaheuristic algorithm to achieve high-quality solutions needs to make a balance between these two processes [29–31]. There are various metaheuristic algorithms that applied to FS that among them genetic algorithm (GA) as an old algorithm and grey wolf optimizer (GWO) as a new algorithm have shown excellent abilities in this domain [29, 32, 33]. Despite the many successfulness each of the two algorithms face weaknesses that can be resolved by hybridization them with each other [34, 35]. In this study is presented a lightweight anomaly detection model using SVM for WSNs in IoT (termed as LIDS) through a hybrid feature selection algorithm based on GA and GWO that named GABGWO. The main contributions of the proposed model include the following:

(1) Applying SVM classifier to distinguish between anomalous wireless traffic from normal wireless traffic with goal of development of the LIDS.
(2) GABGWO improves the efficiency of the SVM prediction model by means of recognition of the most relevant and informative wireless traffic intelligently.
(3) AWID as a real Wi-Fi traffic dataset is used for experimentation and validation purposes. Data preprocessing technique was carried out to accelerate the speed of searching by the GABGWO and to transform the AWID dataset into a compatible format supported by the SVM classifier and as a result obtain the accurate result of the proposed LIDS.
(4) The proficiency of the GABGWO hybrid algorithm in advancing LIDS is judged under two scenarios. The different performance measures used for evaluations consisting of the number of selected features (SF), accuracy (ACC), f-score (F1), recall (R), precision (P), false alarm rate (FAR), and computational times (CTs) that includes the time used by LIDS to detect intrusions (LIDS_Time) and the time used by GABGWO

* Corresponding author.
Email addresses: davahli@qom-iau.ac.ir (A. Davahli), shamsi@qut.ac.ir (M. Shamsi), abaee@shdu.ac.ir (G. Abaei)
https://dx.doi.org/10.22108/jcs.2020.119468.1033
ISSN: 2322-4460

(FS_Time).

The remainder of this paper is structured as follows: Section 2 presents the related works. In Section 3, a preliminary background of the GA and GWO is given. Section 4 explains the proposed model, which consists of data preprocessing, combining GA and GWO to FS, and anomaly classification major stages. The experiments and their setup and results are presented in Section 5 and Section 6, respectively. Finally, Section 7 concludes the paper with future works.

2 Related Works

The literature review shows that there are different researches in which used the various metaheuristic algorithms to advance and improve the performance of the classifier-based anomaly detection and other application fields. In this section, firstly the existing researches in other applications are overviewed. Secondly, the existing researches in anomaly detection application are summarized in Table 1.

2.1 Overview of Existing Researches for Other Applications

The authors in [27] proposed a new wrapper FS approach based on a new artificial bee colony (ABC) optimizer and KNN classifier that integrated with a non-dominated sorting process and genetic operators. They performed two different implementations that include binary ABC and continuous ABC. Their method was examined on 12 various datasets. The results in compared with other methods showed that their binary approach outperformed the other methods in terms of both ACC and CTs.

Two binary versions of WOA with KNN was proposed to do the FS for classification purposes [59]. In the first version, the authors aimed to study the efficiency of using the tournament and roulette wheel selection techniques instead of using a random operator for searching process. Then, they used crossover and mutation operators to improve the exploitation of the WOA. Their evaluations are performed on 20UCI datasets and then compared to ant lion optimizer (ALO), GA and particle swarm optimization (PSO), and five filter FS techniques. The evaluation results with ACC, SF and CTs metrics showed the efficiency of this approach.

The authors have developed two binary versions of GWO with KNN for FS in paper [60], and named the binary versions BGWO1 and BGWO2 respectively. In the BGWO1, while other agents moved toward the first three best agents also binaries, then a probabilistic crossover was performed between them (the three basic moved) to find the updated binary grey wolf position. In BGWO2, a sigmoidal function to binaries has been used to update the grey wolf position vector only and these values randomly threshold to find the updated binary grey wolf position. The performed evaluations over 18 various datasets from the UCI repository were indicative of a good ability of the proposed BGWOs in terms of ACC and SF measures.

Two different approaches for presenting different binary versions of ALO to FS problem were developed in [61]. The first approach only is based on ALO operators while the continuous stages were threshold via a threshold function in the second one. A set of evaluations based on 21UCI datasets and with SF,ER and CTs metrics, proved efficiency of the proposed method in compared to GA, PSO, and bat algorithms.

A new FS method through multi-objective GWO was proposed in [62]. In this method to find a subset of features with minor redundancy, at the first steps was used filter technique. Then at the later steps was employed wrapper technique with KNN machine learning algorithm. The performed assessments of this method over different 8 UCI datasets and against GA and PSO and other single-objective methods showed that the proposed multi-objective method achieved better ACC.

Another wrapper FS method based on GWO and KNN was presented in [63]. In this work, GWO tries to find the optimal features of the complex features space via the interaction of agents in the population. The result of the implementation of the proposed method in this work against GA and PSO over 8UCI datasets proved that the presented approach provides better performance in both SF and ACC.

In [64] the researches have been used firefly algorithm (FFA) for presenting an efficient FS method. Their method was called return-cost-based binary FFA (Rc-BBFA). Their mechanism was conducted in three steps. First, they measured a firefly's attractiveness from other ones by an indicator based on the return-cost. Second, in order to search the attractive one for every FFA, they proposed a Pareto dominance-based method. Third, to update the position of a FFA they developed a binary operator based on the return-cost attractiveness and the adaptive jump. They showed that their method with providing the better ACC, F1 and CTs outperformed other FS methods that are based on GA, PSO, and traditional FFA through a set of experiments over 10UCI datasets.

By using a multi-objective PSO algorithm and KNN, a FS method to choose the unreliable data was advanced in [65]. In this study, an efficient multi-objective FS algorithm is developed by providing two

Table 1. Summary of the Metaheuristic-Based FS Mechanisms for Anomaly Detection Using Classifiers Literature.

S. No.	Author & Year	Metaheuristic Algorithm	Machine Learning Algorithm	Traffic Type	Dataset	Performance Measure
1	Tao et al. 2018 [20]	GA	SVM	Wired	KDDcup99	R, ER, FAR, CTs
2	Raman et al. 2017 [36]	GA	SVM	Wired	KDDcup99, ISCXIDS2012	ACC, R, FAR, CTs
3	Senthilnayaki et al. 2015 [37]	GA	SVM	Wired	KDDcup99	ACC, SF
4	Ahmad et al. 2014 [38]	GA	SVM	Wired	KDDcup99	R, FAR, SF, CTs
5	Dastanpour et al. 2013 [39]	GA	SVM	Wired	KDDcup99	R, FAR, SF
6	Ferriyan et al. 2017 [40]	GA	Random Forest (RF), Naïve Bayes (NB), KNN, C4.5, Bayesian Net (BN)	Wired	NSL-KDD	R, CTs
7	Khammash and Krichen 2017 [41]	GA	Logistic Regression (LR)	Wired	KDDcup99, UNSW-NB15	ACC, R, FAR, SF
8	Desale et al. 2015 [42]	GA	NB, J48	Wired	NSL-KDD	ACC, CTs, SF
9	Senthilnayaki et al. 2013 [43]	GA	J48	Wired	KDDcup99	R, ER, CTs
10	Sindhu et al. 2012 [44]	GA	RF, NB, C4.5, Random Tree (RT), Decision Stump (DS), Representative Tree (REPT)	Wired	KDDcup99	R, P, F1, ER, SF
11	Alzubi et al. 2019 [45]	GWO	SVM	Wired	NSL-KDD	ACC, R, FAR, SF
12	Sathish et al. 2017 [46]	GWO	SVM	Wired	KDDcup99	ACC, FAR
13	Srivastava et al. 2019 [47]	GWO	SVM, KNN, ANN	Wired	KDDcup99	ACC, R, TNR
14	Roopa Devi and Suganthe 2017 [48]	GWO	SVM, NB	Wired	NSL-KDD	ACC, R, P, F1, SF, CTs
15	Seth and Chandra 2016 [49]	GWO	ANN	Wired	NSL-KDD	ACC, SF
16	Davahli et al. 2020 [50]	GA, GWO	SVM	Wireless	AWID	ACC, R, FAR, SF, P, F1, CTs
17	Roopa Devi et al. 2018 [51]	GWO, Cuckoo Search Optimization (CuSO)	SVM	Wired	NSL-KDD	ACC, R, TNR, P
18	Mazini et al. 2019 [52]	ABC	AdaBoost	Wired	NSL-KDD, ISCXIDS2012	R,ACC,FAR,SF,CTs
19	Qureshi et al. 2019b [53]	ABC	Random Neural Network (RNN)	Wired	NSL-KDD	ACC, R, TNR, FAR
20	Xue et al. 2018 [6]	Self-adaptive Differential Evolution (SaDE)	KNN	Wireless	KDDcup99	ACC, R
21	Li et al. 2018 [54]	Bat	RF	Wireless	KDDcup99	ACC,R,P,F1,FAR, CTs
22	Usha and Kavitha 2017 [9]	PSO	SVM	Wireless	AWID	ACC, R, P,F1, FAR, CTs
23	Bostani et al. 2017 [55]	Gravitational Search Algorithm (GSA)	SVM	Wired	NSL-KDD	ACC, R,FAR, SF, CTs
24	Kang and Kim 2016 [56]	Local Search Algorithm (LSA)	Multi-Layer Perceptron (MLP)	Wired	NSL-KDD	ACC, R, FAR, SF, CTs
25	Bamakan et al. 2016 [57]	PSO	SVM, Multiple Criteria Linear Programming (MCLP)	Wired	KDDcup99	ACC, R, FAR, SF
26	Eesa et al. 2015 [58]	Cuttle Fish Algorithm (CFA)	Decision Tree (DT)	Wired	KDDcup99	ACC, R, FAR

new operators for PSO. The first operator was designed to overwhelm the decline phenomenon of particles. Hybrid mutation as another operator was designed to enhance the suggested algorithm's search ability. Experimental and comparison results of the proposed method over 6UCI datasets proved that this algorithm is highly competitive in terms of ACC and SF with unreliable data.

Through combining a "chaotic" version of ALO with KNN was proposed another metaheuristic-based FS approach. This method tried to control the exploration and exploitation rate by means the parameter I. The approach was examined using 18UCI datasets with different number of features. The results of examination which were compared with the GA and PSO using ACC, F1 and SF quality metrics, proved that this method provides better performance [66].

The researchers in work [67] aimed to advance a PSO-KDE model in which PSO optimizer was used to simultaneously determine the kernel bandwidth and select the optimal features for kernel density estimation (KDE) classifier. They also used the classification performance and number of selected features to advance the fitness function for PSO-KDE. They evaluated the performance of their method by employing two datasets including wisconsin breast cancer dataset (WBCD) and wisconsin diagnosis breast cancer database (WDBC) using ACC, R and TNR. Evaluating results has demonstrated that the PSO-KDE method performs better than GA-KDE method in average, in diagnosis of breast cancer.

It is noteworthy that in addition to above-mentioned single metaheuristic algorithm-based FS, there are several researches in which a hybrid metaheuristic algorithm-based was presented to FS for general purposes such as [34, 35]. In paper [34], have presented two hybridization models to design different FS mechanisms through whale optimization algorithm (WOA) and simulated annealing (SA) algorithm with KNN. In their first method, SA is embedded in WOA, whereas it is applied to enhance the best solution found after each run of WOA in their second method. Their main purpose of applying SA was to improve the exploitation by probing the most promising regions that founded by WOA. The evaluations on 18 UCI datasets and in compared with other techniques confirmed the efficiency of the proposed method in terms of ACC, SF and CTs.

To solve FS problem a new hybrid algorithms based on binary bat and PSO algorithms and KNN classifier have been proposed in [35]. The hybrid algorithm that was called HBBEPSO, combined the bat optimizer with its capacity for helping explore the search space and the PSO with its ability to converge to the best global optimal in the features space. The performance of the HBBEPSO was investigated in compared with the bat, PSO and other algorithms which have been applied to FS. This method was evaluated over 20UCI datasets and in terms of SF and F1 evaluation metrics. The assessment results proved the capability of HBBEPSO algorithm to search the search space for optimal feature subsets.

2.2 Summarization of Existing Researches for Anomaly Detection

Each work that presents a metaheuristic-based FS method in anomaly detection field is categorized regarding the following characteristics: the authors name and publishing year, the applied metaheuristic algorithm, the utilized machine-learning algorithm, the type of network traffic, the evaluation network dataset and the validation metrics. It should be noted that the order of the studies is as follows: first, they are ordered based on the applied metaheuristic algorithm, second the utilized classifier and third the year of publication. As it can be seen in Table 1, it is tried to place researches which are recently presented and based on GA and GWO with SVM classifier at the top of the table.

3 Background

In this section, an overview of the main processes and characteristics of GA and GWO is given.

3.1 Genetic Algorithm

A genetic algorithm is one of the oldest methods for solving optimization problems. It was proposed by Holland in [68–70]. This algorithm tries to mimic animals and human genes behavior and the new generation production process. The algorithm has three main steps:

- Selection: In this step, it is tried to select some good and high-fitness solutions as parents. The details of selecting parents are proposed in different works in various ways. But for producing new solutions, we need to choose good solutions as parents in GA.
- Crossover: After choosing (usually two) parents in the selection, two new offsprings (new solutions) is produced in this step. A cross point (or two) is selected randomly, and two new solutions are produced by choosing the value of problem parameters from each of the parents on each side of the cross point. This step implemented by the probability of P_c.
- Mutation: To increase the chance of reaching new

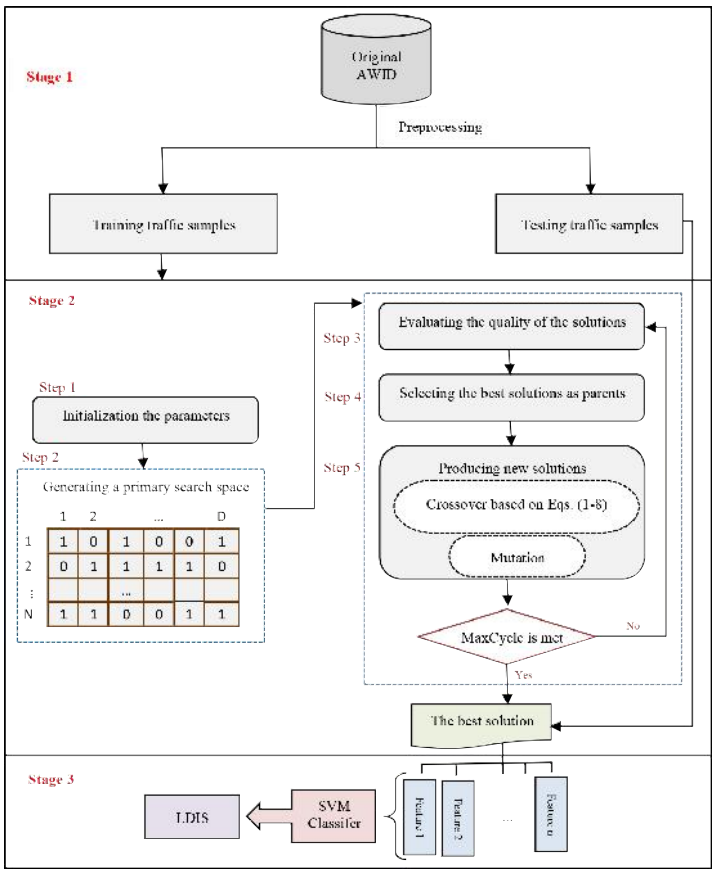


Figure 1. The Stages of the Anomaly Detection Model.

and unseen solutions, in this step we choose a parameter (or more) randomly, and change its value to other possible search space values. Again this step is carried out by probability of P_m.

3.2 Grey Wolf Optimizer

The GWO algorithm was proposed by Mirjalili recently [71], and during these few years, it has gained more attention by computer science researchers [72, 73]. The main idea of this algorithm is obtained from kinds of wolves and their behavior for praying. As an algorithmic method, it tries to show some mathematical behavior like wolves. There exist four kinds of wolves which are called alpha (α), beta (β), delta (δ), and omega (ω). In GWO, every one of candidate solutions (omega wolves) tries to be closer in a direction at a position that is produced by current positions of alpha, beta, and delta wolves. More mathematically, these steps are occurred for every solution to produce a new solution:

$$X[t+1] = X[T] - A \times D \tag{1}$$

X[t] denotes current position and X[t+1] denotes the next position of each wolf, D as a difference vector is calculated based on Eq. (2).

$$D = CX_p[t] - X[t] \tag{2}$$

A and C are coefficient vectors that are estimated based on Eq. (3) and Eq. (4) respectively.

$$A = 2ar_1 - a \tag{3}$$

$$C = 2r_2 \tag{4}$$

The value of r_1, r_2 vectors are generated between 0 and 1 randomly and a vector's value is decreased from 2 to 0 through Eq. (5) in each iteration.

$$a = 2 - t(\frac{2}{\text{MaxCycle}}) \tag{5}$$

t is the number of the current round and MaxCycle is the total number of rounds.

The following equations (6) to (8) are used for updating the position of omega wolves based on the positions of alpha, beta, and delta wolves:

$$X[t+1][i] = \frac{X_1[t][i] + X_2[t][i] + X_2[t][i]}{3} \tag{6}$$

t represents the current round, i denotes the index of variables and the X_1, X_2 and X_3 as ith omegas' positions at the ith round, are initialized by Eq. (7).

```
Initialize the parameters MaxCycle, D, iter, P_c, P_m, α, β and δ
Initialize N Solutions, X = {x_1, x_2, ... x_N}
Compute the fitness value of every solutions
For iter < MaxCycle
Begin
Evaluate
Select x_α, x_β, x_δ
For each solution
Begin
    If rand [0, 1] < P_c
    Begin
    Initialize an temporary array y_D = current solution
            If 0 < rand [0, D] < α
            For i = 0 to α
            Begin
                y_i = calculate x_i by using Eqs. (1) to (8)
            End
            Evaluate y
            If fitness (y) > fitness (current solution)
                Replace current solution with y

        Else if α < rand [0, D] < β
            For i = α to β
            Begin
                y_i = calculate x_i by using Eqs. (1) to (8)
            End
            Evaluate y
            If fitness (y) > fitness (current solution)
                Replace current solution with y

        Else if β < rand [0, D] < δ
            For i = β to δ
            Begin
                y_i = calculate x_i by using Eqs. (1) to (8)
            End
            Evaluate y
            If fitness (y) > fitness (current solution)
                Replace current solution with y
    End
End
Mutation the worst solution
If fitness (mutated solution) > fitness (worst solution)
    Replace current solution with y
End
```

Figure 2. Pseudo-Code of Hybrid GABGWO Algorithm.

$$\begin{aligned} X_1[i] &= X_\alpha[i] - A_1[i]D_\alpha[i], \\ X_2[i] &= X_\beta[i] - A_2[i]D_\beta[i], \\ X_3[i] &= X_\delta[i] - A_3[i]D_\delta[i] \end{aligned} \tag{7}$$

X_α, X_β and X_δ denote the position of alpha, beta, and delta wolves respectively, as before mentioned, the value of A_1, A_2 and A_3 are determined by Eq. (3), and the value of D_α, D_β and D_δ are determined by Eq. (8).

$$\begin{aligned} D_\alpha[i] &= C_1X_\alpha[i] - X[i], \\ D_\beta[i] &= C_2X_\alpha[i] - X[i], \\ D_\delta[i] &= C_3X_\delta[i] - X[i] \end{aligned} \tag{8}$$

The value of C_1, C_2, and C_3 are determined by Eq. (4).

4 The Suggested Model

To advance a lightweight anomaly detection model by considering the resource limitation problem in wireless networks, as Figure 1 illustrates three main stages including (1) data preprocessing stage, (2) wrapper feature selection stage, and (3) the anomaly classification stage are integrated.

According to the Figure 1 each stage itself has some steps which are described in detail as below:

4.1 The Data Preprocessing Stage

Generally, the preprocessing stage is conducted in five main steps: In the first step, the question marks are replaced with zero value. In the second step, the features that have nominal value e.g. source address or initialization vector (IV) with hexadecimal value are changed into an integer value. In the third step, the continuous values i.e. timestamps are normalized between zero and one, according to Eq. (9).

$$z_i = \frac{x_i - min(x)}{\max(x) - \min(x)} \tag{9}$$

Where z_i is considered as the normalized value, Xi refers to the corresponding feature value whose i can vary from 1 to 154, and min(x) and max(x) are the minimum and maximum values of the feature x, respectively. In the balancing process, to avoid outnumbering the attack instances by normal instances, the dataset with the ratio 1-to-1 between normal and attack instances is balanced. Finally, in the sampling process, 10000 instances as the training dataset are randomly sampled and features related to these instances consist of different values in traffic generated by various attacks; afterward, 5000 instances as testing dataset are sampled.

Where z_i is considered as the normalized value, X_i represents the corresponding feature value in which i can initialize from 1 to 154, and min(x) and max(x) are the minimum and maximum values of the feature x, respectively. In the fourth step, to avoid outnumbering the attack instances by normal instances, the dataset with the ratio 1-to-1 between normal and anomaly records is balanced.

Finally, because of the huge amount of the AWID dataset in the fifth step, 10000 instances as the training dataset are randomly sampled that consist of various intrusions, afterward, 5000 instances as testing dataset are sampled.

4.2 The Wrapper Feature Selection Stage

Because the nature of intrusion wireless traffic is nonlinear the LIDS faces the huge amount of network data that some amount of them are redundant and irrelevant features causing slow training and testing procedure, higher resource usage, and inaccurate detection. Hence, selecting the informative and optimal features for LIDS to accurate detection with low computational costs is crucial. Hence, an effective wrapper FS hybrid algorithm (GABGWO) with two novel operators namely crossover and mutation based on concepts of GA and mathematical equations of GWO is presented in this stage. GABGWO's pseudo-code is given in Figure 2. The steps related to combination

Table 2. Parameters Setting.

S. No	Parameters	Value
1	No. of Candidate Solutions (N)	8
2	MaxCycle	20
3	Number of Runs (M)	10
4	Candidate Solution Dimension (D)	Total number of features in dataset =154
5	Search Space Domain	0 and 1
6	GABGWO_Probability_Crossover (Pc)	0.8
7	GABGWO_Probability_Mutation (Pm)	0.03
8	α	90
9	β	134
10	δ	154
11	GA_Probability_Crossover (Pc)	0.8
12	GA_Probability_Mutation (Pm)	0.03
13	FWP-SVM-GA_Probability_Crossover (Pc)	0.75
14	FWP-SVM-GA_Probability_Mutation (Pm)	0.09
15	SVM_kernel	RBF
16	SVM_gamma	1/k which k is number of features
17	K, in K-fold_Validation_Cross	10

GA and GWO are presented in detail in the following steps:

Step 1- Initialization the parameters the parameter setting is one of the processes that affect the performance of the new hybrid GABGWO algorithm, like any optimization algorithm. In general, process of parameters setting of the random metaheuristic algorithms is performed by considering the algorithm's application or optimization problem. In this step, the initialization of all GABGWO's parameters is empirically carried out in a way that a trade-off between the number of selected features and accuracy is established. Table 2 shows the initialization of all GABGWO's parameters.

Step 2- Generating a primary search space This process creates a primary population of candidate solutions (feature subsets) by sets of '1' and '0' bits as seen in Figure 1. Each candidate solution will present a possible feature subset. The bit (gene) with value '1' represents the feature is selected, and '0' represents the feature is not selected. The N denotes the total number of candidate solutions, and D denotes the dimension of the candidate solutions that initialization of them is given in Table 2. Afterward, the following processes are applied over the solutions until a features subset deemed optimal is found or *MaxCycle* is satisfied. *MaxCycle* is the maximum number of cycles to find the best solutions among of population that is considered as termination criterion and initialized in Table 2.

Step 3- Evaluating the quality of the solutions GABGWO via its fitness function assesses the quality of each feature subset in this step. Since one of the main goals of the proposed hybrid FS algorithm is to increase the LIDS's accuracy, yield accuracy of the SVM that integrated with GABGWO is considered as the arithmetic value of GABGWO's fitness that calculated by Eq. (10).

Step 4- Selecting the best solutions as parents GABGWO chooses the three best feature subsets with based on their fitness value and names themx_α, x_β, x_δ, respectively. x_α, x_β, x_δ are considered as parents which will be incorporated in producing the new population in the next step. The main goal of this process is to update the other feature subsets according to the values of these best feature subsets by crossover operators which can enhance the GABGWO's exploitation ability.

Step 5- Producing new solutions To avoid locally optimal solutions GABGWO tries to update each feature subset of the current population through applying its two new operators, namely crossover and mutation over the best-chosen feature subsets as follows:

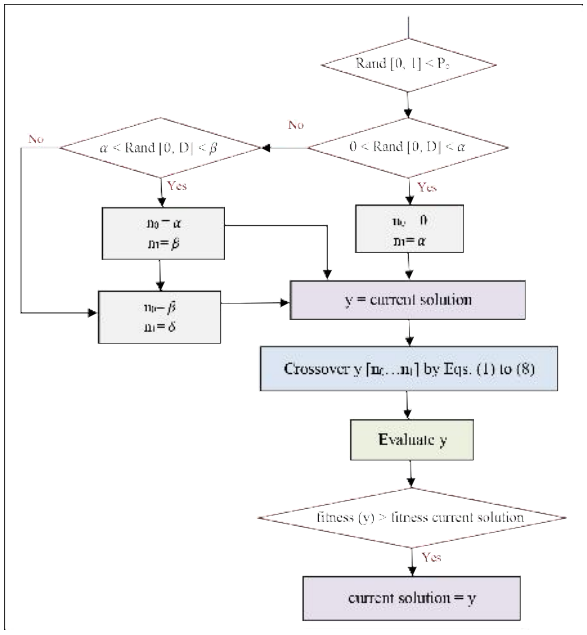


Figure 3. Crossover Operator of GABGWO.

- **Crossover operator** First a random value in 0,1 is generated and as Figure 3 illustrates the crossover operator is only performed if the generated random value is lesser than the control parameter P_c (the probability that GABGWO must do the crossover and is called GABGWO_Probability_Crossover that initialized in Table 2), then another integer random number is produced between [0, D], with regard to that the random number's value whether is $\in \{0 \dots \alpha\}$ or $\in \{\alpha{+}1 \dots \beta\}$ or $\in \{\beta{+}1 \dots \delta\}$, then the bits which are located in these ranges are updated using Eqs. (1) to (8). The α, β, and δ are integer numbers between 0 to D which are initialized in Table 2. These processes that are called crossover in our new GABGWO algorithm, try to update the other feature subsets in accordance with the values of the three best feature subsets via Eqs. (1) to (8). This not only increases the diversity of the GABGWO but also accelerates its convergence.
- **Mutation operator** This operator selects the worst solution resulted from the crossover and modifies its values in a way that '0' is changed to 1 or vice versa. The worst solution produced by crossover is selected and converted by mutation operator in a way that its '0' values are changed to 1 and its '1' values are changed to 0. After that, the fitness value of the mutated solution is calculated and if its fitness value is better than the worst solution, it is replaced with the worst solution. Note that this operator is done with the probability of P_m (the probability that GABGWO must do the mutation and is called GABGWO_Probability_Mutation that initialized in Table 2) and its value is initialized less than P_c.

4.3 Anomaly Classification Stage

Exploiting classifiers to recognize the anomaly traffic from normal ones is one of the most popular methods to advance anomaly detection [74, 75]. Since the SVM classifier has shown an excellent performance in the intrusion detection domain and due to its high ability to deal with huge traffic [19, 20, 74, 75], it is utilized to build the anomaly detection model in this stage. To enhance the performance of SVM and to develop a lightweight anomaly detection system, only the informative and optimal traffic features of the AWID wireless dataset chosen by GABGWO are given to SVM. To high accuracy obtain, the SVM performs k-fold cross-validation (k=10) over the selected traffics by GABGWO where in the wireless traffics are randomly split into 10 equal sections. For learning the anomaly detection system, nine sections are considered and the one remaining section is considered for anomaly identifying. These operations are repeated until all the sections are utilized for identifying. Indeed, these processes are done until all the traffic data are examined. Finally, the mean of 10 examination results is considered as an accurate and lightweight anomaly detection.

5 Experiments

This section describes the used evaluation intrusion dataset, experimental setup, parameter setting, and evaluation criteria.

5.1 The Intrusion Dataset

KDDcup99 and its derived versions are the most commonly used datasets utilized for evaluating the anomaly detection models. However, on the one hand, the KDDcup99 and its derivations are old and do not contain the new attacks, and on the other hand, because they are based on the DARPA 1998 TCP/IP data, they do not have the basic features of wireless networks [76–78]. AWID is a Wi-Fi network benchmark dataset that has recently been created [79] and can be used for the network anomaly detection evaluation, especially for Wi-Fi networks like IoT wireless networks [80, 81]. Thus, AWID is used for evaluating the LIDS. This dataset is freely available from http://icsdweb.aegean.gr/awid/. The CLS with 4 target classes and ATK with 16 target classes are two types of AWID and both types have 155 features. Notably, the 16 classes of ATK version are classified into the 4 classes in the CLS version that used in this work. CLS includes one normal class and three intrusion classes. It also contains 1,795,575 wireless traffic in-

Table 3. Confusion Matrix.

Real class \ Identified class	Anomaly	Normal
Anomaly	True Positive (TP)	False Negative (FN)
Normal	False Positive (FP)	True Negative (TN)

Table 4. The Average of the Obtained Results From Experiments Under Scenario 1.

	SF	ACC(%)	R(%)	P(%)	F1(%)	FAR(%)	LIDS_Time (S)
Without FS	154	99.16	99.46	96.49	97.96	0.63	14.176
FS by GABGWO	94	99.09	99.30	96.31	97.84	0.68	10.906

stances in which 1,633,190 samples are normal traffics, and 162,385 samples are various types of attacks. This benchmark intrusion dataset contains different types of data such as nominal, discrete, and continuous with various ranges. This results in difficulties for LIDS. Besides, unknown data values in AWID have been shown by '?' and this dataset is unbalanced. The normal samples bias SVM and subsequently, affect the performance of the anomaly detection method. In addition to these, the network traffic datasets have a huge number of instances that not only decrease the searching speed of the feature selection algorithm but also complicate the detection process [79–81]. Therefore, the data preprocessing should be conducted to overcome the limitations in advance using the AWID dataset. Besides, because of ubiquitous Wi-Fi devices, the AWID Wi-Fi intrusion dataset has a very large number of records, some of which are redundant and reduce the speed of the GABGWO and LIDS. Therefore, for the mentioned considerations, before using AWID, its necessary to preprocess it.

5.2 Experimental Setup

Experiments are performed by a system with Intel(R) Core(TM) i7-4720HQ processor @ 2.60 GHz, 8 GB RAM, and Windows 10 operating system. Python software is used for dataset preprocessing. The GABGWO and other FS algorithms are implemented by java programming language. SVM is called from the machine learning library of WEKA [82, 83] that attached to java.

5.3 Parameters Setting

The metaheuristic algorithms' parameters initialization is one of the procedures which significantly affects the efficiency of metaheuristic algorithms [30, 84]. This task is usually carried out by considering the algorithm applications or optimization problems [30, 39, 84]. In this work, the initialization of all parameters is empirically performed. Table 2 shows the initialization of all parameters used in this work.

5.4 Evaluation Measures

As seen in Table 1, the existing related works have used different measures for their evaluations. Among them, ACC, R, FAR, SF, and CTs (note that CTs consists of LIDS_Time that is the time consumed for anomaly detection by SVM and FS_Time that is the time consumed for choosing optimal traffics by GABGWO) are the most commonly used metrics. However, in this paper in addition to these metrics, we adopt some other evaluation metrics include P, F1. These metrics are computed according to four main criteria of the confusion matrix in Table 3 and Eqs. (10) to (14) [24, 75, 85].

- True positive: denotes the number of anomaly traffics that are identified truly as an anomaly.
- False Negative: denotes the number of anomaly traffics that are falsely identified as normal.
- False Positive: denotes the number of normal traffics that are falsely identified as an anomaly.
- True Negative: denotes the number of normal traffics that are identified truly as normal.

$$ACC = \frac{(TP + TN)}{(\mathrm{TP+TN+FP+FN})} \tag{10}$$

$$R = \frac{TP}{(\mathrm{TP+FN})} \tag{11}$$

$$P = \frac{TP}{(\mathrm{TP+FP})} \tag{12}$$

$$F1 = \frac{(2 \times TP)}{(2 \times TP) + \mathrm{FP+FN}} \tag{13}$$

$$FAR = \frac{FP}{(\mathrm{FP+TN})} \tag{14}$$

Table 5. The Average of the Obtained Results From Experiments Under Scenario 2.

Algorithm	SF	ACC (%)	R (%)	P (%)	F1 (%)	FAR (%)	LIDS_Time (S)
GA	76	98.87	99.25	95.39	97.28	0.81	11.124
GWO	64	97.22	99.01	93.13	95.97	1.28	12.161
FWP-SVM-GA	81	98.87	99.20	95.48	97.30	0.82	11.390
BGWO	104	98.61	96.35	96.99	96.67	0.75	12.091
GABGWO	94	99.09	99.30	96.31	97.84	0.68	10.906

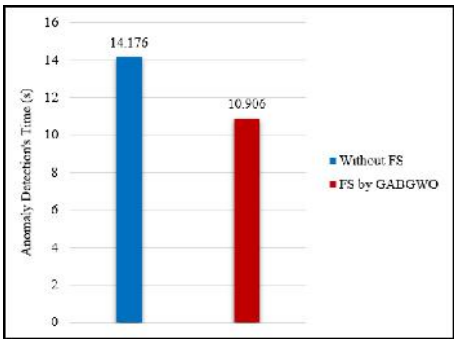


Figure 4. The Reduction Time for LIDS by GABGWO.

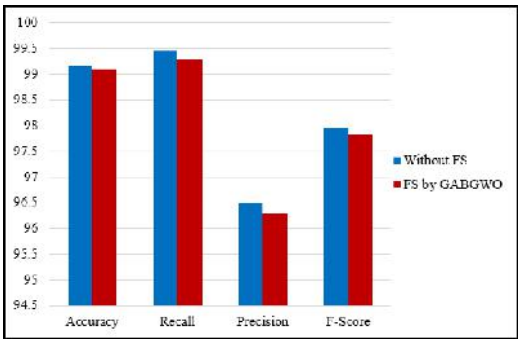


Figure 5. The performance comparison of LIDS without and with FS by GABGWO.

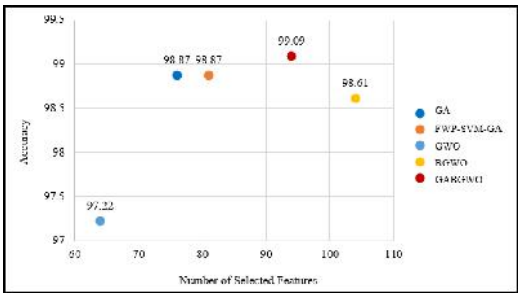


Figure 6. The Presented Accuracy Per Number of Chosen Features by LIDS Based on GABGWO and Other Algorithms.

6 Experimental Scenarios, Results, and Analysis

In this section, two scenarios are defined for evaluating the performance of the proposed anomaly detection method and also investigating the effectiveness of the GABGWO on it. It is important to note that, to achieve reliable results from random and non-deterministic GABGWO algorithm, it is performed for M (that is set in table2) times, then the average of M results is considered as the final result.

Scenario 1 The performance of the LIDS with all traffic features is compared to its performance with traffic features subset chosen from GABGWO and the average of the achieved results is given in Table 4.

The experiment results in Table 4 show the GABGWO can increase anomaly detection's speed with reducing the dimensionality of the huge network traffic, as seen in Figure 4. Besides, as Table 4 and Figure 5 illustrate the proposed method also satisfied the main goal of a wrapper FS method, namely, decreasing the number of features and maintaining the accuracies intact or at least a little worse. The obtained results in Table 4 show that the selected optimal features subset by GABGWO give accuracies near to accuracies obtained using whole features.

Scenario 2 The performance of the LIDS based on GABGWO is compared with its performance based on pure GA, GWO, and two other recent FS algorithms such as FWP-SVM-GA [20] and BGWO [60] with the same fitness function and parameter settings according to the Table 2. The average of the achieved results is summarized in Table 5.

As the experiment results show in Table 5, the GABGWO provides better performance than the original GA and GWO and other existing FS techniques with respect to various evaluation measures. GABGWO through choosing lesser optimal features than BGWO and choosing more informative features than GA, GWO, and FWP-SVM-GA, has presented high anomaly detection accuracy which these are clear in Figure 6 and 7.

Furthermore, as Figures 8 and 9 illustrate, choosing the more informative and related features by GABGWO causes more decrease in the FAR and computational time of the anomaly detection method than other techniques.

The GABGWO is also compared with GA, GWO, and FWP-SVM-GA metaheuristic FS algorithms in

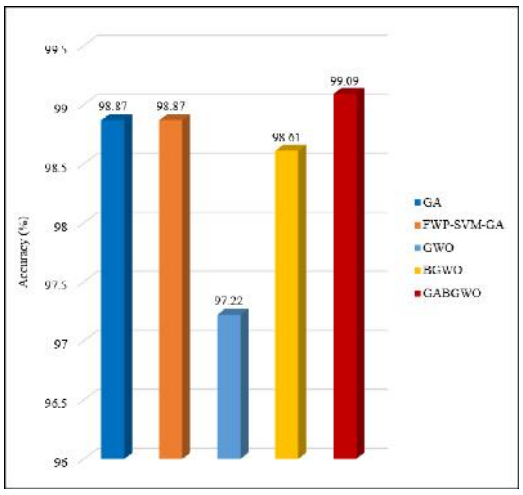


Figure 7. The Presented Accuracy by LIDS Based on GABGWO and Other Algorithms.

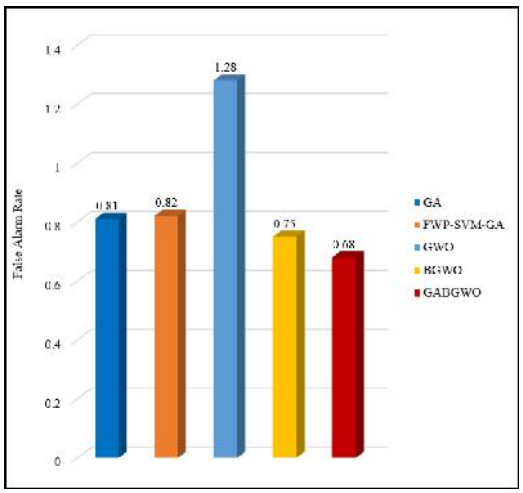


Figure 8. The Presented FAR by LIDS Based on GABGWO and Other Algorithms.

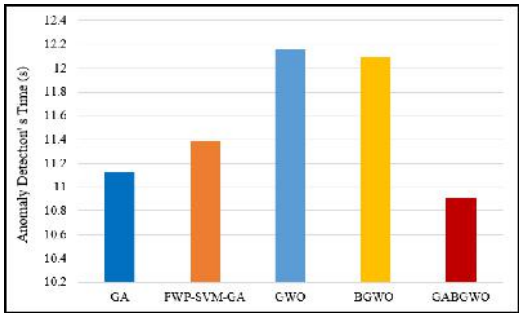


Figure 9. The Presented Anomaly Detection Time by LIDS Based on GABGWO and Other Algorithms.

terms of consumption time for producing and choosing the best solution for FS (FS_Time) and average comparison results are outlined in Table 6 and illustrated by Figure 10. The results prove that GABGWO provides less FS's time than BGWO and GWO but it still in comparison with GA and FWP-SVM-GW has more consumption time.

Table 6. The Average FS'S Time by GABGWO and Other FS Algorithms.

Algorithm	FS_Time (S)
GA	647.082
GWO	1157.397
FWP-SVM-GA	639.208
BGWO	3459.12
GABGWO	1147.156

7 Conclusions and Future Work

In this paper, we have presented an SVM-based lightweight anomaly detection model named LIDS, using combination concepts of GA and mathematical equations of GWO (GABGWO) for WSNs in IoT. Generally, the proposed model involves three main stages: to speed up the GABGWO and convert the complex traffics of the AWID Wi-Fi intrusion dataset into a readable format for SVM, a set of preprocessing operations are performed in stage 1. For selecting the optimal wireless traffic data to incorporate in anomaly detection, a wrapper FS technique based on the combination of GA and GWO is developed in five steps at stage 2. In the first step, all the parameters related to GABGWO, are initialized, in the second step, a primary search space is generated, quality of the solutions are evaluated in the third step, the best solutions are determined as parents in the fourth step, and finally in the fifth step, producing new solutions based on GABGWO 's crossover and mutation operators is performed. In stage 3, the optimal traffic features selected by GABGWO are given the SVM classifier to distinguish between anomaly and normal traffics. Experimentations are done under two different scenarios. In the first scenario, the proposed anomaly detection method is conducted with all features and in the second scenario, it conducted with features subset resulted by the GABGWO. The empirical results obtained from experiments proved that not only the GABGWO provided good performance for LIDS, but also it outperformed the pure GA and GWO and the existing FS algorithms for various evaluation metrics. In future work, the proposed hybrid algorithm will be extended with lower runtime to address different optimization problems.

References

[1] S. H. Jafier. Utilizing feature selection techniques in intrusion detection system for internet of things. In *Proceedings of the 2nd International Conference on Future Networks and Distributed Systems*, page 1–3, 2018.

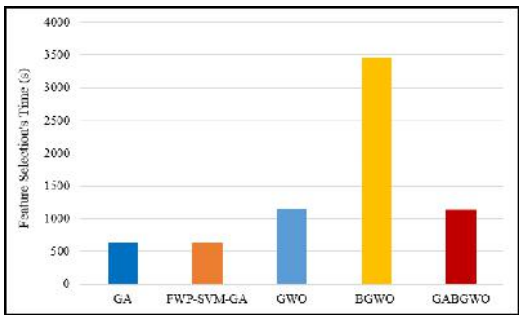


Figure 10. The Presented FS'S Time by GABGWO and Other Algorithms.

doi:10.1145/3231053.3234323.

[2] O. Flauzac, C. J. Gonzalez Santamaría, and F. Nolot. New security architecture for IoT network. *Procedia Computer Science*, 52:1028–1033, 2015. doi:10.1016/j.procs.2015.05.099.

[3] S. H. Jafier. Security issues and challenges for the IoT-based smart grid. In *Proceedings of the 2nd International Conference on Future Networks and Distributed Systems*, page 1–3, 2018. doi:10.1145/3231053.3234323.

[4] M. Sheikhan and H. Bostani. A hybrid intrusion detection architecture for internet of things. In *2016 8th International Symposium on Telecommunications (IST)*, pages 601–606. IEEE, 2017. ISBN 978-1-5090-3436-9. doi:10.1109/ISTEL.2016.7881893.

[5] A. Qureshi, H. L., J. Ahmad, and N. Mtetwa. A Heuristic Intrusion Detection System for Internet-of-Things (IoT). In *Intelligent Computing - Proceedings of the Computing Conference*, pages 86–98. Springer, Cham, 2019. ISBN 978-3-030-22870-5. doi:10.1007/978-3-030-22871-2_7.

[6] Y. Xue, W. Jia, X. Zhao, and W. Pang. An evolutionary computation based feature selection method for intrusion detection. *Security and Communication Networks*, 2018, 2018. doi:10.1155/2018/2492956.

[7] M. Alidoosti and A. Nowroozi. Cross layer-based intrusion detection based on network behavior for IoT. In *Cross layer-based intrusion detection based on network behavior for IoT*, pages 1–4. IEEE, 2018. ISBN 978-1-5386-1268-2. doi:10.1109/WAMICON.2018.8363921.

[8] A. A. Gendreau and M. Moorman. Survey of intrusion detection systems towards an end to end secure internet of things. In *2016 IEEE 4th international conference on future internet of things and cloud (FiCloud)*, pages 84–90. IEEE, 2016. ISBN 978-1-5090-4053-7. doi:10.1109/FiCloud.2016.20.

[9] M. Usha and P. Kavitha. Anomaly based intrusion detection for 802.11 networks with optimal features using SVM classifier. *Wireless Networks*, 23(8):2431–2446, 2017. doi:10.1007/s11276-016-1300-5.

[10] H. M. Aldosari. A proposed security layer for the Internet of things communication reference model. *Procedia Computer Science*, 65:95–98, 2015. doi:10.1016/j.procs.2015.09.084.

[11] F. Restuccia, S. D'Oro, and T. Melodia. Securing the internet of things in the age of machine learning and software-defined networking. *IEEE Internet of Things Journal*, 5(6):4829 – 4842, 2018. doi:10.1109/JIOT.2018.2846040.

[12] B. B. Zarpelão, R. S. Miani, C. T. Kawakani, and S. C. de Alvarenga. A survey of intrusion detection in Internet of Things. *Journal of Network and Computer Applications*, 84:25–37, 2017. doi:10.1016/j.jnca.2017.02.009.

[13] F. Restuccia, S. D'Oro, and T. Melodia. A survey on Internet of Things architectures. *Journal of King Saud University-Computer and Information Sciences*, 30(3):291–319, 2018. doi:10.1016/j.jksuci.2016.10.003.

[14] D. M. Mendez, I. Papapanagiotou, and B. Yang. Internet of things: Survey on security and privacy. *arXiv preprint arXiv:1707.01879*, pages 291–319, 2017. doi:10.1080/19393555.2018.1458258.

[15] L. Xiao, X. Wan, X. Lu, Y. Zhang, and D. Wu. IoT security techniques based on machine learning. *arXiv preprint arXiv:1707.01879*, 2018.

[16] H. Bostani and M. Sheikhan. Hybrid of anomaly-based and specification-based IDS for Internet of Things using unsupervised OPF based on MapReduce approach. *Computer Communications*, 98: 52–71, 2017. doi:10.1016/j.comcom.2016.12.001.

[17] A. A. Diro and N. Chilamkurti. Distributed attack detection scheme using deep learning approach for Internet of Things. *Future Generation Computer Systems*, 82:761–768, 2018. doi:10.1016/j.future.2017.08.043.

[18] S. Aljawarneh, M. Aldwairi, and M. B. Yassein. Anomaly-based intrusion detection system through feature selection analysis and building hybrid efficient model. *Journal of Computational Science*, 25:152–160, 2018. doi:10.1016/j.jocs.2017.03.006.

[19] D. Andročec and N. Vrček. Machine Learning for the Internet of Things Security: A Systematic Review. In *13th International Conference on Software Technologies*, 2018. doi:10.5220/0006841205630570.

[20] P. Tao, Z. Sun, and Z. Sun. An improved intrusion detection algorithm based on GA and SVM. *IEEE Access*, 6:13624 – 13631, 2018. ISSN 2169-3536. doi:10.1109/ACCESS.2018.2810198.

[21] K Anusha and E. Sathiyamoorthy. Comparative study for feature selection algorithms in intrusion detection system. *Automatic*

Control and Computer Sciences, 50:1–9, 2016. doi:10.3103/S0146411616010028.

[22] G. Chandrashekar and F. Sahin. A survey on feature selection methods. *Computers & Electrical Engineering*, 40(1):1, 2014. doi:10.1016/j.compeleceng.2013.11.024.

[23] T. Hamed, R. Dara, and S. C. Kremer. Network intrusion detection system based on recursive feature addition and bigram technique. *computers & security*, 73:137–155, 2018. doi:10.1016/j.cose.2017.10.011.

[24] R. Sheikhpour, M. Sarram Agha, S. Gharaghani, and M. A. Z. Chahooki. A survey on semi-supervised feature selection methods. *Pattern Recognition*, 64:141–158, 2017. doi:10.1016/j.patcog.2016.11.003.

[25] K. El-Khatib. Impact of feature reduction on the efficiency of wireless intrusion detection systems. *IEEE Transactions on Parallel and Distributed Systems*, 21(8):1143 – 1149, 2009. ISSN 1558-2183. doi:10.1109/TPDS.2009.142.

[26] B. Xue, M. Zhang, W. N. Browne, and X. Yao. A survey on evolutionary computation approaches to feature selection. *IEEE Transactions on Parallel and Distributed Systems*, 20(4):606 – 626, 2015. ISSN 1941-0026. doi:10.1109/TEVC.2015.2504420.

[27] E. Hancer, B. Xue, M. Zhang, D. Karaboga, and B. Akay. Pareto front feature selection based on artificial bee colony optimization. *Information Sciences*, 422:462–479, 2018. doi:10.1016/j.ins.2017.09.028.

[28] V. R. Balasaraswathi, M. Sugumaran, and Y. Hamid. Feature selection techniques for intrusion detection using non-bio-inspired and bio-inspired optimization algorithms. *Journal of Communications and Information Networks*, 2(4): 107–119, 2017. doi:10.1007/s41650-017-0033-7.

[29] H. Faris, I. Aljarah, M. A. Al-Betar, and S. Mirjalili. Grey wolf optimizer: a review of recent variants and applications. *Neural computing and applications*, 30(2):413–435, 2018. doi:10.1007/s00521-017-3272-5.

[30] M. Črepinšek, S. Liu, and M. Mernik. Exploration and exploitation in evolutionary algorithms: A survey. *ACM computing surveys (CSUR)*, 45(3): 1–33, 2013. doi:10.1145/2480741.2480752.

[31] N. Singh and S.Singh. Hybrid algorithm of particle swarm optimization and grey wolf optimizer for improving convergence performance. *Journal of Applied Mathematics*, 2017, 2017. doi:10.1155/2017/2030489.

[32] W. Siedlecki and J. Sklansky. A note on genetic algorithms for large-scale feature selection. In *Handbook of pattern recognition and computer vision*, pages 88–107. World Scientific, 1993. doi:10.1142/9789814343138_0005.

[33] C. Tsai, W. Eberle, and C. Chu. Genetic algorithms in feature and instance selection. *Knowledge-Based Systems*, 39:240–247, 2013. doi:10.1016/j.knosys.2012.11.005.

[34] M. M. Mafarja and S. Mirjalili. Hybrid Whale Optimization Algorithm with simulated annealing for feature selection. *Neurocomputing*, 260:302–312, 2017. doi:10.1016/j.neucom.2017.04.053.

[35] M. A. Tawhid and K. B. Dsouza. Hybrid Binary Bat Enhanced Particle Swarm Optimization Algorithm for solving feature selection problems. *Applied Computing and Informatics*, 2018. doi:10.1016/j.aci.2018.04.001.

[36] M. G. Raman, N. Somu, K. Kirthivasan, R. Liscano, and V. S. Sriram. An efficient intrusion detection system based on hypergraph-Genetic algorithm for parameter optimization and feature selection in support vector machine. *Knowledge-Based Systems*, 134:1–12, 2017. ISSN 671–682. doi:10.1016/j.knosys.2017.07.005.

[37] B. Senthilnayaki, K. Venkatalakshmi, and A. Kannan. Intrusion detection using optimal genetic feature selection and SVM based classifier. In *2015 3rd International Conference on Signal Processing, Communication and Networking (ICSCN)*, pages 1–4. IEEE, 2015. ISBN 978-1-4673-6823-0. doi:10.1109/ICSCN.2015.7219890.

[38] I. Ahmad, M. Hussain, A. Alghamdi, and A. Alelaiwi. Enhancing SVM performance in intrusion detection using optimal feature subset selection based on genetic principal components. *Neural computing and applications*, 24(7-8):1671–1682, 2014. doi:10.1007/s00521-013-1370-6.

[39] A. Dastanpour and R. A. R. Mahmood. Feature selection based on genetic algorithm and SupportVector machine for intrusion detection system. In *The Second International Conference on Informatics Engineering & Information Science (ICIEIS2013)*, pages 169–181, 2013. doi:10.13140/2.1.4289.4721.

[40] A. Ferriyan, A. H. Thamrin, K. Takeda, and J. Murai. Feature selection using genetic algorithm to improve classification in network intrusion detection system. In *2017 International Electronics Symposium on Knowledge Creation and Intelligent Computing (IES-KCIC)*, pages 46–49. IEEE, 2017. ISBN 978-1-5386-0716-9. doi:10.1109/KCIC.2017.8228458.

[41] C. Khammassi and S. Krichen. A GA-LR wrapper approach for feature selection in network intrusion detection. *computers & security*, 70:255–277, 2017. doi:10.1016/j.cose.2017.06.005.

[42] K. S. Desale and R. Ade. Genetic algorithm based feature selection approach for effective intrusion detection system. In *2015 International Confer-*

ence on Computer Communication and Informatics (ICCCI), pages 1–6. IEEE, 2015. ISBN 978-1-4799-6804-6. doi:10.1109/ICCCI.2015.7218109.

[43] B. Senthilnayaki, K. Venkatalakshmi, and A. Kannan. An intelligent intrusion detection system using genetic based feature selection and Modified J48 decision tree classifier. In *2013 fifth international conference on advanced computing (ICoAC)*, pages 1–7. IEEE, 2013. ISBN 978-1-4799-3448-5. doi:10.1109/ICoAC.2013.6921918.

[44] S. S. S. Sindhu, S. Geetha, and A. Kannan. Decision tree based light weight intrusion detection using a wrapper approach. *Expert Systems with applications*, 39(1):129–141, 2012. doi:10.1016/j.eswa.2011.06.013.

[45] Q. M. Alzubi, M. Anbar, Z. N. Alqattan, M. A. Al-Betar, and R. Abdullah. Intrusion detection system based on a modified binary grey wolf optimization. *Neural Computing and Applications*, 32: 6125–6137, 2020. doi:10.1007/s00521-019-04103-1.

[46] V. Sathish, P. Khader, and S. Abdul. Improved Detecting Host Based Intrusions Based On Hybrid SVM Using Grey Wolf Optimizer. *International Journal Of Security and Its Applications*, 11 (9):59–72, 2017. doi:10.14257/ijsia.2017.11.9.05.

[47] D. Srivastava, R. Singh, and V. Singh. An Intelligent Gray Wolf Optimizer: A Nature Inspired Technique in Intrusion Detection System (IDS). *Journal of Advancements in Robotics*, 6(1):18–24, 2019. ISSN 2455-1872.

[48] E. Devi and R. Suganthe. Enhanced transductive support vector machine classification with grey wolf optimizer cuckoo search optimization for intrusion detection system. *Concurrency and Computation: Practice and Experience*, 32(4), 2018. doi:10.1002/cpe.4999.

[49] J. Seth Kumar and S. Chandra. Intrusion detection based on key feature selection using binary GWO. In *2016 3rd international conference on computing for sustainable global development (INDIACom)*, pages 3735–3740. IEEE, 2017. ISBN 978-1-4673-9417-8.

[50] A. Davahli, M. Shamsi, and G. Abaei. Hybridizing genetic algorithm and grey wolf optimizer to advance an intelligent and lightweight intrusion detection system for IoT wireless networks. *Journal of Ambient Intelligence and Humanized Computing*, 2020. doi:10.1007/s12652-020-01919-x.

[51] E. Devi and R. Suganthe. Feature selection in intrusion detection grey wolf optimizer. *Asian Journal of Research in Social Sciences and Humanities*, 7(3):671–682, 2017. ISSN 671–682. doi:10.5958/2249-7315.2017.00197.6.

[52] M. Mazini, B. Shirazi, and I. Mahdavi. Anomaly network-based intrusion detection system using a reliable hybrid artificial bee colony and AdaBoost algorithms. *Knowledge-Based Systems*, 31(4):541–553, 2019. doi:10.1016/j.jksuci.2018.03.011.

[53] A. Qureshi, H. Larijani, N. Mtetwa, A. Javed, and J. Ahmad. RNN-ABC: A New Swarm Optimization Based Technique for Anomaly Detection. *Computers*, 8(3), 2019. doi:10.3390/computers8030059.

[54] J. Li, Z. Zhao, R. Li, and H. Zhang. AI-based Two-Stage Intrusion Detection for Software Defined IoT Networks. *IEEE Internet of Things Journal*, 6(2):2093 – 2102, 2018. ISSN 2455-1872. doi:10.1109/JIOT.2018.2883344.

[55] H. Bostani and M. Sheikhan. Hybrid of binary gravitational search algorithm and mutual information for feature selection in intrusion detection systems. *Soft computing*, 21(9):2307–2324, 2017. doi:10.1007/s00500-015-1942-8.

[56] S. Kang and K. J. Kim. A feature selection approach to find optimal feature subsets for the network intrusion detection system. *Cluster Computing*, 19(1):325–333, 2016. doi:10.1007/s10586-015-0527-8.

[57] S. M. H. Bamakan, H. Wang, T. Yingjie, and Y.Shi. An effective intrusion detection framework based on MCLP/SVM optimized by time-varying chaos particle swarm optimization. *Neurocomputing*, 199:90–102, 2016. doi:10.1016/j.neucom.2016.03.031.

[58] A. S. Eesa, Z. Orman, and A. M. A. Brifcani. A novel feature-selection approach based on the cuttlefish optimization algorithm for intrusion detection systems. *Expert Systems with Applications*, 42(5):2670–2679, 2015. doi:10.1016/j.eswa.2014.11.009.

[59] M. Mafarja and S. Mirjalili. Whale optimization approaches for wrapper feature selection. *Applied Soft Computing*, 62:441–453, 2018. doi:10.1016/j.asoc.2017.11.006.

[60] E. Emary, H. M. Zawbaa, and A. E. Hassanien. Binary grey wolf optimization approaches for feature selection. *Neurocomputing*, 172:371–381, 2016. doi:10.1016/j.neucom.2015.06.083.

[61] E. Emary, H. M. Zawbaa, and A. E. Hassanien. Binary ant lion approaches for feature selection. *Neurocomputing*, 213:54–65, 2016. doi:10.1016/j.neucom.2016.03.101.

[62] E. Emary, W. Yamany, A. E. Hassanien, and V. Snasel. Multi-objective gray-wolf optimization for attribute reduction. *Procedia Computer Science*, 65:623–632, 2015. doi:10.1016/j.procs.2015.09.006.

[63] E. Emary, H. M. Zawbaa, C. Grosan, and A. E. Hassenian. Feature subset selection approach by gray-wolf optimization. In *Afro-European conference for industrial advancement*, pages 1–

13. Springer, 2015. ISBN 978-3-319-13571-7. doi:10.1007/978-3-319-13572-4_1.

[64] Y. Zhang, X. Song, and D. Gong. A return-cost-based binary firefly algorithm for feature selection. *Information Sciences*, 418-419:561–574, 2017. doi:10.1016/j.ins.2017.08.047.

[65] Z. Yong, G. Dun-wei, and Z. Wan-qiu. Feature selection of unreliable data using an improved multi-objective PSO algorithm. *Neurocomputing*, 171:1281–1290, 2016. doi:10.1016/j.neucom.2015.07.057.

[66] H. M. Zawbaa, E. Emary, and C. Grosan. Feature selection via chaotic antlion optimization. *PloS one*, 11(3), 2016. doi:10.1371/journal.pone.0150652.

[67] R. Sheikhpour, M. A. Sarram, and R. Sheikhpour. Particle swarm optimization for bandwidth determination and feature selection of kernel density estimation based classifiers in diagnosis of breast cancer. *Applied Soft Computing*, 40:113–131, 2016. doi:10.1016/j.asoc.2015.10.005.

[68] J. Holland. Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence. *Control and artificial intelligence*, 1975.

[69] J. H. Holland. *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence*. MIT press, 1992.

[70] J. H. Holland. Genetic Algorithms, Scientific American. *Scientific american*, 267(1):66–73, 1992. doi:https://www.jstor.org/stable/24939139.

[71] S. Mirjalili, S. M. Mirjalili, and A. Lewis. Grey Wolf Optimizer. *Advances in engineering software*, 69:46–61, 2014. doi:10.1016/j.advengsoft.2013.12.007.

[72] A. Kishor and P. K. Singh. Empirical study of grey wolf optimizer. In *Proceedings of fifth international conference on soft computing for problem solving*, pages 1037–1049. Springer, Singapore, 2016. ISBN 978-981-10-0447-6. doi:10.1007/978-981-10-0448-3_87.

[73] M. A. Al-Betar, M. A. Awadallah, H. Faris, I. Aljarah, and A. I. Hammouri. Natural selection methods for grey wolf optimizer. *Expert Systems with Applications*, 113:481–498, 2018. doi:10.1016/j.eswa.2018.07.022.

[74] E. A. Shams and A. Rizaner. A novel support vector machine based intrusion detection system for mobile ad hoc networks. *Wireless Networks*, 24(5):1821–1829, 2018. doi:10.1007/s11276-016-1439-0.

[75] M. Al-Garadi, A. Mohamed, A. Al-Ali, X. Du, and M. Guizani. A survey of machine and deep learning methods for internet of things (IoT) security. *arXiv preprint arXiv:1807.11023*, 2018. doi:10.1109/comst.2020.2988293.

[76] P. Aggarwal and S. K. Sharma. Analysis of KDD dataset attributes-class wise for intrusion detection. *Procedia Computer Science*, 57:842–851, 2015. doi:10.1016/j.procs.2015.07.490.

[77] L. Dhanabal and S. Shantharajah. A study on NSL-KDD dataset for intrusion detection system based on classification algorithms. *International Journal of Advanced Research in Computer and Communication Engineering*, 4(6):446–452, 2015. doi:10.17148/IJARCCE.2015.4696. 446.

[78] M. Alidoosti and A. Nowroozi. A detailed analysis of the KDD CUP 99 data set. In *2009 IEEE symposium on computational intelligence for security and defense applications*, pages 1–6. IEEE, 2009. ISBN 978-1-4244-3763-4. doi:10.1109/CISDA.2009.5356528.

[79] C. Kolias, G. Kambourakis, A. Stavrou, and S. Gritzalis. Intrusion detection in 802.11 networks: empirical evaluation of threats and a public dataset. *IEEE Communications Surveys & Tutorials*, 18(1):184 – 208, 2015. doi:10.1109/COMST.2015.2402161.

[80] M. E. Aminanto, H. C. Tanuwidjaja, P. D. Yoo, and K. Kim. Wi-Fi intrusion detection using weighted-feature selection for neural networks classifier. In *2017 International Workshop on Big Data and Information Security (IWBIS)*, pages 99–104. IEEE, 2017. ISBN 978-1-5386-2038-0. doi:10.1109/IWBIS.2017.8275109.

[81] S. H. Jafier. Detecting impersonation attack in WiFi networks using deep learning approach. In *International Workshop on Information Security Applications*, pages 136–147. Springer, 2017. ISBN 978-3-319-56548-4. doi:10.1007/978-3-319-56549-1_12.

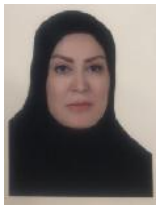
[82] I. Witten, E. Frank, and M. Hall. *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, 2016.

[83] M. Alidoosti and A. Nowroozi. Weka: Practical machine learning tools and techniques with java implementations. In *Proc ICONIP/ANZIIS/ANNES99 Future Directions for Intelligent Systems and Information Sciences*, pages 192–196. Morgan Kaufmann, 1999. ISBN 978-1-5386-7582-3.

[84] A. Eiben and S. Smit. Parameter tuning for configuring and analyzing evolutionary algorithms. *Swarm and Evolutionary Computation*, 1(1):19–31, 2011. doi:10.1016/j.swevo.2011.02.001.

[85] Y. Xin, L. Kong, Z. Liu, Y. Chen, Y. Li, H. Zhu, M. Gao, H. Hou, and C. Wang. Machine learning and deep learning methods for cybersecurity. *IEEE Access*, 6:35365 – 35381, 2018. doi:10.1109/ACCESS.2018.2836950.

Azam Davahli received her B.S. degree in Software Engineering and M.S. degree in Computer Networks from Islamic Azad University, Qazvin, Iran, in 2006, 2009 respectively. Currently, she is a Ph.D. student in Software Engineering at Islamic Azad University Qom, Iran. Her research interests are the security of wireless and internet of things networks, cryptography, feature selection, optimization algorithms, and soft computing.



Mahboubeh Shamsi received her B.S. degree in Mathematics from Isfahan University, Isfahan, Iran, and her M.S. degree in Software Engineering from Islamic Azad University, Najafabad, Iran, and her Ph.D. degree in Software Engineering from University Technology Malaysia, Malaysia, in 2003, 2006 and 2011, respectively. Currently, she is an assistant professor in the faculty of Electrical and Computer Engineering at the Qom University of Technology, Qom, Iran. Her research interests include image processing, soft computing, cloud computing, and central and distributed database.

Golnoush Abaei received her M.S. in Software Engineering from the University of Mysore, Mysore, Karnataka State, India, and her Ph.D. degree in Artificial Intelligence from University Technology Malaysia, Malaysia, in 2008 and 2015, respectively. Currently, she is an assistant professor in the faculty of Electrical, Computer, and Biomedical Engineering and Head of Research in Shahabdanesh Institute of Higher Education in Iran. Her research interests are artificial intelligence, soft computing, software testing, and software defect prediction.