Received: July 30, 2023. Revised: August 27, 2023.

INASS

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

An Efficient Hybrid Filter-Wrapper Feature Selection Approach for Network Intrusion Detection System

Samer Saeed Issa[1] Sinan Q. Salih[2]* Yasir Dawood Salman[3] Faris Hasan Taha[4]

[1]*Computer science department, Al-Rafidain University College, Baghdad, Iraq*
[2]*Technical College of Engineering, Al-Bayan University, Baghdad, Iraq*
[3]*Department of Technical Computer Engineering, Dijlah University College, Baghdad, Iraq*
[4]*Department of Medical Equipment Technology Engineering, College of Engineering Technology, Al-Kitab University, Kirkuk, Iraq*
* Corresponding author's Email: sinan.salih@albayan.edu.iq

Abstract: The detection rate of network intrusion detection systems mainly depends on relevant features; however, the selection of attributes or features is considered an issue in NP-hard problems. It is an important step in machine learning and pattern recognition. The major aim of feature selection is to determine the feature subset from the current/existing features that will enhance the learning performance of the algorithms, in terms of accuracy and learning time. This paper proposes a new hybrid filter-wrapper feature selection method that can be used in classification problems. The information gain ratio algorithm (GR) represents the filter feature selection approach, and the black hole algorithm (BHA) represents the wrapper feature selection approach. The comparative analysis of network intrusion detection methods focuses on accuracy and false positive rate. GBA shines with exceptional results: achieving 96.96% accuracy and a mere 0.89% false positive rate. This success can be traced to GBA's improved initialization via the GR technique, which effectively removes irrelevant features. By assigning these features almost zero weights, GBA hones its ability to accurately spot intrusions while drastically reducing false alarms. These standout outcomes underline GBA's superiority over other methods, showcasing its potential as a reliable solution for bolstering network security.

Keywords: Information security, Intrusion detection systems, Optimization, Feature selection, Black hole algorithm.

1. Introduction

Intrusion detection systems (IDS) play a critical role in securing information and communication systems. They are designed to detect and identify network traffic that poses a threat. IDS employ various soft computing models, such as artificial neural networks, Bayesian networks, genetic algorithms, fuzzy logic, and decision trees, to effectively identify anomalies and misuse. Feature selection is a crucial aspect of IDS, as it involves selecting relevant attributes and eliminating irrelevant ones from datasets, enhancing the performance of data learning models.

Signature-based IDS, a common classification of IDS, is capable of recognizing patterns in traffic or application data that may indicate a potential attack. It maintains a database of attack signatures and regularly updates it to ensure efficient detection. On the other hand, anomaly-based IDS compares all activities against predefined patterns to identify any abnormal behavior [1-3].

The main purpose of an IDS is to detect network attacks and promptly alert system administrators. An effective IDS should be capable of efficiently identifying malicious attacks and implementing appropriate countermeasures. This research aims to improve and enhance the accuracy of IDS systems, as current systems have limitations in their detection capabilities. The IDS employs a range of soft computing models, including artificial neural networks, Bayesian networks, fuzzy logic, J84,

decision trees, and genetic algorithms [4-6]. The techniques used in misuse and anomaly detection systems can be categorized into three main types: knowledge-based detection, statistical-based detection, and machine learning detection [2, 7].

Numerous machine learning techniques have been employed to enhance the attack detection threshold and accuracy of IDS systems. These techniques have been instrumental in developing effective classification and clustering models that can distinguish between attacks and normal network behaviour. Accurately identifying intrusions within the vast amount of network traffic has always presented a challenging task for IDS systems [8]. The training data may contain some irrelevant features that do not aid in the detection process. These unimportant features are usually redundant and can add noise to the classifier's design. As a result, it's important to select data with useful characteristics that will help classifier performance improve [9-11].

While acquiring enough training data can be challenging, the size of the required training samples can be reduced by performing feature selection. This process helps enhance the overall performance of classification algorithms. Feature selection, also known as attribute reduction, is a widely studied and significant topic across various domains, including machine learning, signal processing, data mining, and pattern recognition[12-14].

Attributes reduction is the process of selecting significant attributes and eliminating irrelevant ones from a dataset, resulting in a more efficient data learning model. The pruned dataset maintains an accurate representation of the original data features essential for describing the data. [12, 15-17]. The selection of features is a challenging NP-hard problem, and developing an efficient algorithm using the minimum attribute reduction method is a significant task [18, 19].

Recently, swarm-based and evolutionary methods such as ant colony optimization (ACO) [12, 20, 21], genetic algorithm (GA) [16, 22, 23], and artificial bee colony (ABC) [24, 25] have been explored in this context. In addition, particle swarm optimization (PSO) [9, 26] and harmony search algorithm (HSA) have been used to handle the problems of features selection [27, 28].

The study conducted by [29] introduced a meta-heuristic optimization method called the "black hole" algorithm, inspired by the gravitational pull of black holes on neighbouring stars. The development of the BHA algorithm was founded on the interaction between the black hole and its neighbouring stars. The primary contributions of this research are as follows:

- A hybrid filter-wrapper algorithm, information gain ratio - black hole algorithm (GBA), was introduced. Information gain ratio determined feature importance, while BHA optimized star positions for feature selection, using Naïve Bayesian classifier as the objective function.
- The algorithm was tested on benchmark problems, particularly NSL-KDD dataset, aiming to enhance accuracy and detection rate of network intrusion detection systems by selecting the optimal feature subset.

This paper is organized as follows. Section 2 explains the problem of feature selection, while section 3 explains the proposed algorithm in details. The results and discussion are presented in section 4. Finally, section 5 presents the conclusion of the study.

2. Preliminaries

2.1 The feature selection problem

Feature selection is a crucial step when working with datasets that contain a large number of features. The primary goal of feature selection is to reduce the features set by eliminating redundant features and retaining informative ones to improve the efficiency of classifiers. One significant advantage of feature selection is the reduction in data volume required for the learning process, resulting in faster computation, lower memory usage, and improved accuracy and speed of classification.

The problem of feature selection is known to be NP-hard [18, 19]. Assuming we have a dataset D with #F features, the total number of features in the dataset is DS = D × #F. The concept of feature selection involves selecting #f features from the entire feature subset (#f < #F), aiming to maximize an objective function such as classification accuracy. This optimization problem involves two major decisions: the value of #f and the optimal subset of features within the subset. Given a set of features #F, the subset feature selection problem aims to identify a subset L ⊆ #F that satisfies |L| = #f and maximizes (or minimizes) the objective function:

$$O(L) = \max O(X) \quad X \subseteq \#F, |X| = \#f \quad (1)$$

Efficient objective functions play a crucial role in subset feature selection, but finding a universal function that suits all data mining problems is challenging. The objective function typically focuses on the accuracy of classifiers.

Feature selection methods can be classified into filter, wrapper, embedded, and hybrid models. Filter

methods address feature selection by conducting statistical analysis of datasets without incorporating learning algorithms. These methods are typically fast in feature selection, such as information gain [30-32], information gain ratio[31], gini index[32], and fisher score [33]. However, the filter approach treats each feature independently, potentially disregarding dependencies among features that could impact the performance of the classification model when compared to others [12].

Wrapper-based models evaluate subset features using learning algorithms as the objective function for prediction and performance assessment [34]. There are two main types of wrapper approach: Sequential feature selection (SFS) algorithms and heuristic search algorithms. SFS algorithms can be further categorized into backward selection and forward selection algorithms. Backward selection algorithms commence with an empty set of features and progressively include more features until reaching the maximum objective function. Conversely, in forward selection, the algorithms begin with a full set of features and progressively remove some features to maximize the objective function. Backward selection algorithms are computationally intensive as they gradually eliminate features, causing longer execution times, particularly for large datasets. In contrast, forward selection algorithms can make early suboptimal choices, potentially missing valuable features or selecting less beneficial ones as they progressively add features. Balancing computational efficiency and optimal feature selection is crucial when using these methods.

In contrast, heuristic search algorithms are used to optimize objective functions by evaluating different subsets. These algorithms generate multiple subsets of data either by exploring a search space or generating problem-specific solutions. The main challenge of wrapper-based models is the high computational cost associated with obtaining subset features. Typically, a learning algorithm is trained and tested on individual subsets to evaluate classifier accuracy, with significant computational time spent on training predictors for high-dimensional datasets [12, 19].

Embedded models integrate feature selection into training, particularly in binary decision trees. Hybrid methods blend filter and wrapper approaches, using filters to reduce dimensionality before generating diverse subsets[18]. Hybrid models focus on integrating filter and wrapper-based approaches to achieve high-performing learning algorithms while minimizing computational time compared to filter-based methods.

One example of a hybrid approach involves the combination of a genetic algorithm and mutual information for identifying relevant subset features in classification tasks [35]. Instead of solely optimizing the classification error rate, this method optimizes the mutual information between predictive labels and true class labels in a trained classifier. Real-world datasets were used to validate this optimized approach, and the results demonstrated that the hybrid method outperformed filter-based methods in terms of accuracy. Thus, it was concluded that the hybrid method is more efficient than the wrapper method. This approach can suffer from prolonged convergence times due to their inherent randomness, making them computationally demanding, especially for large datasets. Additionally, Hu et al. explored the use of filter and wrapper methods for discovering biomarkers in cancer classification using microarray gene expression data [36].

The Fisher's ratio, in a combination of methods, was utilized as the filtering technique. Rigorous testing on real datasets demonstrated that the hybrid approach achieved superior computational efficiency compared to the wrapper method. Moreover, the results indicated a significant advantage of the hybrid approach over the simple filter method. Long et al. introduced a novel and self-adaptive firefly algorithm known as DbFAFS. [37]. This method aimed to address the limitations of the classical firefly algorithm in exploring new search spaces, particularly in local areas, Additionally, Ahmed et al. presented BAMI, a hybrid algorithm based on the Bat algorithm that combined elements from Naïve Bayes and mutual information [38]. The results indicated that BAMI was more efficient than the conventional Bat algorithm with Naïve Bayes (BANV). As a result, it was concluded that BAMI had significantly reduced computation time compared to BANV. Both FA and BA require careful parameter tuning, which can be time-consuming and complex, affecting their efficiency and ease of implementation.

Due to it ease of implementation, and lack of controlling parameters, Black hole algorithm is utilized in this study as a wrapper feature selection approach.

3. Black hole algorithm (BHA)

As mentioned earlier, the BHA algorithm is primarily based on the concept of a region in space characterized by a high concentration of mass, resulting in a strong gravitational force that prevents anything from escaping its pull. This region, known as the event horizon, leads to the permanent loss of any object that enters it. The BHA consists of two main components: the migration of stars that have

crossed the event horizon and their subsequent re-initialization. The algorithm operates as following: a set of N stars (where N = denotes the number of stars), are randomly positioned within the search space. The star with the best evaluation function is designated as the black hole, denoted as x_{BH}, which remains stationary unless a star with a superior solution is discovered. The number N represents the total number of candidate stars actively searching for the optimal solution. The movement of each star towards the black hole in each generation can be determined using the following equation:

$$x_i(t+1) = x_i(t) + rand \times \left(x_{BH} - x_i(t)\right) \quad i = 1.2.\cdots N, \quad (2)$$

The equation uses the variable "$rand$" to represent a randomly generated number ranging from 0 to 1. In BHA, any star that is located within a distance less than the event horizon to the black hole will vanish. The event horizon is defined by a radius (R) that can be calculated as follows:

$$R = \frac{f_{BH}}{\sum_{i=1}^{N} f_i} \quad (3)$$

In the BHA algorithm, the fitness values of BH and the individual stars (represented by f_i and f_{BH}, respectively) are used to evaluate their performance. The variable N denotes the total number of stars or individual solutions in the algorithm. If the distance between an individual solution and the black hole is less than a specified radius (R), the individual solution collapses, and a new individual solution is randomly generated and distributed in the solution space. One advantage of the BHA is its parameter-less nature, which simplifies its implementation. Unlike some other heuristics, the BHA has the ability to converge to the global optimum in all runs and is not prone to being trapped in local optima[29, 39-42]. In this study, the BHA was selected as the feature selection method for enhancing the detection rate of IDS due to its simplicity, ease of implementation, and absence of specific parameters.

4. The proposed algorithm (GBA)

The primary objective of developing a hybrid model for feature selection is to strike a better balance between the computational efficiency of filter models and the performance accuracy of wrapper models. In traditional wrapper models like BHA, all stars are randomly initialized with selected features at random positions. In the proposed algorithm, all stars in the swarm are initialized with favourable positions, ensuring that the population starts the search process from a promising starting point. Moreover, the incorporation of the selection of good features aims to strategically guide the search effort and facilitate the convergence of the population towards the best-known solution. Fig. 1 depicts the main flowchart of the proposed modification to the standard BHA. By combining the strengths of filter and wrapper approaches, our GBA algorithm significantly enhances the accuracy and detection rate of network intrusion detection systems. This strategic integration of feature assessment, initialization, and optimization offers a theoretical foundation for the algorithm's effectiveness and its outperformance of the existing techniques, as we will demonstrate through rigorous evaluation on benchmark datasets.

The key steps in the proposed algorithm are as follows:

I. Initialization:

This step initiates with the calculation of the gain ratio of each feature in the data set using the following Equation:

$$GR\ (f) = \frac{IG(f)}{IV(f)} \quad (4)$$

where IG is the gain value of a feature f which can be calculated using Eq. (5); IV represents the intrinsic value, which is the generated potential information from the partitioning of the training set, corresponding to the partition's outcomes on attributes, IV can be calculated using Eq. (6).

$$IG(f) = H(S) - \sum_i \frac{Si}{S} H(Si) \quad (5)$$

$$IV(f) = -\sum_i \frac{|D_i|}{D} \times \log_2 \frac{|D_i|}{D} \quad (6)$$

The GR output for each feature is a real number, but for the purpose of the proposed method, binary digits are used to represent features. The binary representation uses "0" for non-selected features and "1" for selected features. To convert the gain ratio into binary format within the range [0,1], the following equation was utilized:

$$X_i = \begin{cases} 1, Sigmoid\ (Gr_i) > U(0,1) \\ 0, Otherwise \end{cases} \quad (7)$$

Where X_i is the position of a star, the sigmoid value for (Gr_i) is $1/\ (1 + e^{-Gr_i})$, and U is the uniform distribution.

II. Fitness function

The main aim of the proposed algorithm is to minimize the classification error rate on the

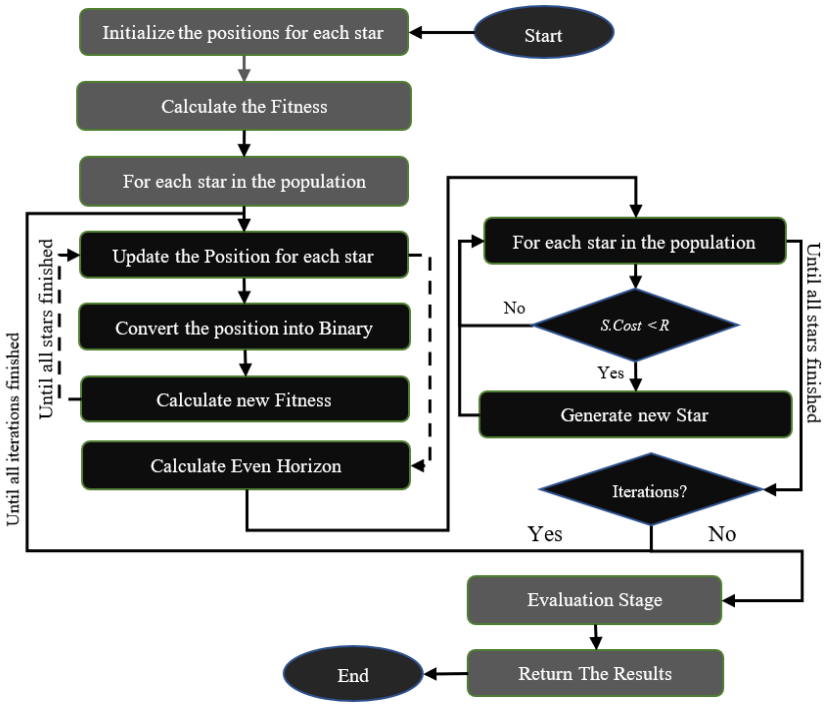

Figure. 1 The main flowchart of GBA for feature selection

validation set while maximizing the number of non-selected features, which are considered irrelevant. This objective is expressed in Eq. (8).

To compute the fitness function, a classifier is employed, and in this research, the Naïve Bayesian classifier was used to assess the accuracy.

$$Err = \sigma * \frac{[\#F]}{[\#All\ F]} + (1 - \sigma) * \frac{Err[\#F]}{Err[\#All\ F]} \quad (8)$$

where $\#F$ is the selected features; Err is the classifier error rate, specifically the 5- cross-validation error rate obtained after training the Naïve Bayesian classifier; and σ is a constant value limited to the range [0,1] controlling the significance of classification performance concerning the number of selected features.

III. Execute BHA steps:

- Read the inputs: Dataset, number of stars, number of iterations.
- Initialize the size of each star based on the output of gain ration method.
- Generate random stars (i.e., Solutions) randomly in the search space.
- Encode the solutions in binary via Eq. (6).
- Evaluate each star using the fitness function.
- Set the best star in terms of the fitness value as black hole (BH).
- Update the position of each solution via the following equation:

$$S_i.X_j^{new} = S_i.X_j^{old} + Rand(0,1) \times (BH - S_i.X_j^{old}) \quad (9)$$

Where BH represents the Black Hole or the best solution in the current iteration, and $S_i.X_j^{new}$ and $S_i.X_j^{old}$ represent the new and the old position of the start S_i respectively.

The updated position, should be re-converted into binary form, because the binary values or the sequence of 0s and 1s have changed. The continuous values should convert using Eq. (4). Then, the star should be re-evaluated using the fitness function.

- Calculate and check the event horizon (R)

$$R = \frac{BH.Cost}{\sum_{i=1}^{N} S_i.Cost} \quad (10)$$

In this step, the new positions for all stars are evaluated whether they have crossed the event horizon or not. The event horizon (R) is calculated in each iteration based on the cost of the black hole using the following equation:

The cost of each star in the population is compared with the value of R, the star with cost lower than R is eliminated and regenerated using Initialization.

- The stop condition in the proposed GBA algorithm is the number of iterations, which is a fixed number. The algorithm stops when the algorithm reached that number, otherwise the algorithm executes the movement and regenerating steps again. To be

more specific, if the number of loops still lower than the number of iterations, then go back.

- Calculate the final results using the evaluation metrics, and print the final results.

5. Results and discussion

A novel hybrid filter-wrapper feature selection algorithm for selecting optimal subset features was proposed in this study. With this algorithm, it is aimed that the minimum number of selected features which provides the highest rate of classification accuracy will be selected.

In this paper, black hole algorithm has been used as a wrapper feature selection and hybridized with information gain ratio method, which represents a filter feature selection method, the proposed algorithm called GBA. In order to test GBA, five datasets are used in the experiments and comparison results. The datasets have various number of instances (rows) and features as representative a various number of issues, as shown in Table 1.

GBA has been tested and compared with well-known algorithms, which are particle swarm optimization (PSO) [43], genetic algorithm (GA) [44], Standard Firefly algorithm [45] and Bat algorithm[38, 46]. These algorithms used the Naïve Bayesian classifier for calculating the fitness function same as the GBA. Furthermore, the parameters settings for the studied algorithms are presented in Table 2.

The results are divided into three parts, first part shows the final number of selected features. While second part displays the classification accuracies among two different classifiers for the selected subset features. The last part shows the average number of iterations, which illustrates the speed and the performance of the proposed algorithm.

Table 3 provides the results of the comparison between GBA and the other algorithms, in terms of the number of selected features from the original datasets. The best results obtained from the proposed algorithm or the comparative algorithms have been highlighted in bold. It is obvious that GBA obtained results better than PSO, GA and the standard FA. The obtained results showed that the proposed GBA and BA are with same performance. To test the difference in performance among all the algorithms, the Wilcoxon test was performed and the results of the study are presented in Table 4. The results confirmed a significant difference in the performance of the studied algorithms (GBA, PSO, GA and FA) except for BA which had a similar performance with the proposed algorithm.

The experiment above illustrated the resulted features, which is the first part of the test. The second part evaluates the classification accuracy obtained by GBA and other algorithms. The experiment has been done 10 times by using two different well-known classifiers, JRip and J48, with 5-fold cross validation. The average accuracies obtained by these classifiers

Table 1. The datasets properties

Datasets	#Instances	#Features
Exactly	1000	13
Exactly2	1000	13
HeartEW	294	13
M-of-N	1000	13
Vote	300	16

Table 2. Parameters settings

Parameter		Value
General	Swarm Size	25
	Iterations	250
	Fitness function constant σ	0.999
	No. of Runs	20
FA	β_0	1.0
	γ	1.0
	α	0.2
	δ	0.96
PSO	ω	0.1
	c	0.1
GA	Migration Fraction	0.2
	Crossover Fraction	0.8
BA	Pulse Rate (r)	0.9
	Min Frequency (f_{min})	0
	Max Frequency (f_{max})	2
	Decrease Sound Loudness (a)	0.9
	Weighting Value (δ)	0.9
	Weighting Value(Φ)	0.1

Table 3. Average of selected features

Dataset	GFA	PSO	GA	FA	BA
Exactly	1	6.9	7	6	1
Exactly2	1	2.9	7	5	1
HeartEW	4	6.25	6	5	4
M-of-N	6	7.6	8	7	6
Vote	1	3.5	3	2	1

Table 4. Wilcoxon test results

Dataset	vs. PSO	vs. GA	vs. FA	vs. BA
Exactly	**.000**	**.000**	**.000**	1
Exactly2	**.000**	**.000**	**.000**	1
HeartEW	**.000**	**.000**	.013	1
M-of-N	**.000**	.013	.013	1
Vote	**.000**	**.000**	.013	1

Table 5. Accuracy using J48 classifier

Dataset	GBA	PSO	GA	FA	BA
Exactly	68.8 ±**0**	68.8 ±**0**	68.8 ±**0**	68.8 ±**0**	68.8 ±**0**
Exactly2	75.8 ±**0**	75.8 ±**0**	75.8 ±**0**	75.8 ± **0**	75.8 ±**0**
HeartEW	79.93 ±**0**	79.08 ±0.28	79.25 ±0	79.43 ±0.22	79.93 ±0
M-of-N	100.0±**0**	100.0 ±**0**	100.0 ±**0**	100.0 ±**0**	100.0 ±**0**
Vote	95.0 ±**0**	94.36 ±0	94.0 ±0	94.52 ±0	95.0 ±**0**

Table 6. Accuracy using JRip classifier

Dataset	GBA	PSO	GA	FA	BA
Exactly	**68.8** ±**0**	68.01 ±0.22	68.0 ±0	**68.8** ±**0**	**68.8** ±**0**
Exactly2	**75.8** ±**0**	**75.8** ±**0**	**75.8** ±**0**	**75.8** ± **0**	**75.8** ±**0**
HeartEW	80.61 ±0	**82.47** ±**5.33**	81.97 ±0	80.33 ±0	80.61 ±0
M-of-N	98.9 ±0	98.92 ±0.23	**99.1** ±**0**	98.8 ±0	98.9 ±0
Vote	**95.0** ±**0**	94.42 ±0.35	93.66 ±0	94.61 ±0.18	**95.0** ±**0**

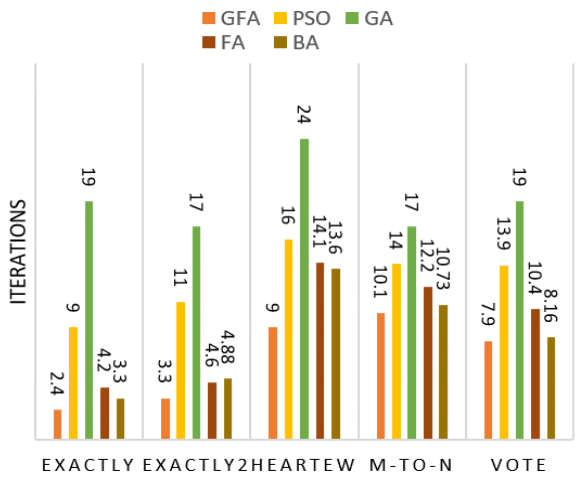

Figure. 2 Comparison between different feature selection methods

are shown in Tables 5 and 6 respectively.

Table 2 presents the performance of the evaluated algorithms in selecting feature subsets. It can be observed that the proposed GBA and BA achieved the smallest number of features across the datasets. Notably, GBA demonstrated significant differences in performance compared to the benchmarking algorithms (PSO, GA, and FFA), except for BA which showed similar performance to GBA (Table 3). It is worth noting that GBA even reduced the number of features to a single feature in three datasets (Table 3).

When evaluating the feature subsets, considering the interaction between classification accuracy and the number of selected features by GBA in comparison to the benchmarking algorithms, three sets of results can be obtained:

1. In some cases, GBA achieved the same classification accuracy as the benchmarking algorithms while using a reduced number of features. This is demonstrated in datasets Exactly, Exactly2, and M-of-N with similar classification accuracies in the J48 classifier (Table 5). Similar results were observed for the Exactly2 dataset in the JRip classifier (Table 6). In contrast, the other benchmarking algorithms (excluding BA) selected more features, indicating the presence of redundant features. This was particularly evident in the Exactly2 dataset, where different numbers of selected features yielded the same level of inaccuracy.
2. In other cases, GBA reduced the number of features and simultaneously increased the classification accuracy. This is evident in the HearEW and Vote datasets using the J48 classifier (Table 5) where GBA and BA outperformed the other algorithms. Similarly, in the Exactly and Vote datasets using the JRip classifier (Table 6), GBA and BA achieved superior performance. It should be noted that FA yielded the same classification rates as GBA and BA in the Exactly dataset. The differences in the number of selected features can be attributed to the presence of noisy features that affected the accuracy of the classification process, as observed in the Vote dataset.
3. In some instances, selecting a smaller feature subset led to slightly lower classification accuracy. This can be seen in the HeartEW and M-of-N datasets using the JRip classifier (Table 6), where PSO and GA exhibited better classification rates compared to GBA and BA. However, GBA and BA still achieved the same classification rates, outperforming FA.

Finally, the results in Table 4, obtained through the Wilcoxon test, indicate that GBA is statistically superior or equal to all other selected algorithms.

6. GBA for IDS

An IDS, as previously defined, is a security tool utilized for the efficient protection of information and

communication systems. Its main purpose is to detect and identify potentially harmful network traffic. Similar to firewalls and antivirus software, an IDS also has access control capabilities. The classification of IDS types, as depicted in Fig. 5 of [47], relies on their detection techniques, particularly differentiating between signature-based and anomaly-based detection systems. Signature-based IDS can identify familiar patterns of malicious traffic or application data by comparing them to a database of attack signatures. On the other hand, anomaly-based IDS identifies deviations from established behaviour by analysing all activities.

For an IDS system to be effective, it should possess certain key characteristics. Firstly, it needs to exhibit accuracy in detecting attacks, minimizing false positive and false negative alarms. Secondly, the system should be extensible, allowing for the updating of various components such as signal analyzers and wireless channels. This extensibility also enables the deployment of multiple monitors in different geographical locations, enhancing the coverage of the wireless network. Lastly, adaptability is crucial to reduce the cost and time required for updating the wireless IDS (WIDS) [48].

In the development of IDS that utilizes machine learning techniques, a crucial aspect is designing appropriate features to distinguish normal behaviours from system or network attacks [49]. The lack of public datasets for objective evaluation has hindered the systematic assessment of the proposed features' effectiveness in developing intrusion detection systems. To address this issue, MIT Lincoln Laboratory [50] provided the 1999 KDDCUP dataset, while Tavallaee et al. [51] offered a modified version called the NSL_KDD dataset. These datasets have been widely used in studies to objectively evaluate the performance of proposed IDS systems.

In the realm of automatically detecting such attacks, IDSs primarily rely on packet monitoring to identify abnormal behaviours. Machine learning techniques are commonly employed to recognize these abnormal traffic patterns. Two popular machine learning approaches for attack detection are classification-based and clustering-based methods. However, certain methods may not be effective when dealing with large volumes of data. Moreover, traffic data often contains numerous features, many of which are irrelevant or redundant. To address this, feature selection algorithms are utilized to eliminate these unnecessary features. In this section, we apply the proposed algorithm to an IDS dataset and evaluate its performance using GBA.

The performance of the achieved feature subsets by the proposed GBA is assessed using various

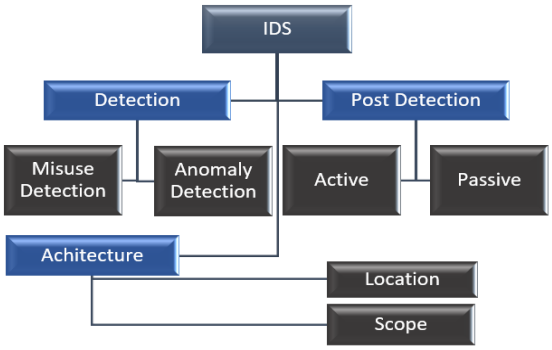

Figure. 3 The classification of IDS

metrics, including accuracy, detection rate, false alarm rate, and mean. These metrics have been extensively utilized in previous studies to evaluate algorithmic performance [52]. The metrics are commonly defined as follows:

$$Acc = \frac{TP+TN}{TP+TN+FP+FN} \quad (12)$$

$$DR = \frac{TP}{TP+FN} \quad (13)$$

$$FAR = \frac{FP}{TP+FP} \quad (14)$$

For the evaluation purposes, all the attack types were converted into a single class called ‘Attack’, while the rest were called ‘Normal’. The GBA was used for the best subset features selection for the binary classification problem of IDS. The data was subdivided into two parts, 70% for training and 30% for testing. The performance of the standard BHA and GBA was evaluated through 20 runs, employing varying numbers of stars and iterations. The objective was to compare the effectiveness of GBA and BHA in identifying the optimal subset of features with improved accuracy and detection rates. Four population sizes, namely 10, 20, 30, and 40, were utilized in this study, while the number of iterations was fixed at 100. The results of these investigations are presented in Table 7.

From the Table 7, it is obvious that GBA had a better accuracy compared to BA; in other words, GBA attained an accuracy of 96 % using 10 swarms while BHA achieved the same accuracy using higher number of swarms (*Stars* = 40). As a result, GBA was able to solve the problem at a faster rate than BA, meaning that GBA required a less computational complexity than BHA. Figs. (3-6) illustrate the comparison between BA and GBA using different swarm sizes and the same number of iterations.

In summary, the GBA algorithm outperforms the BA algorithm in terms of accuracy, detection rate, and false alarm rate across different cases and

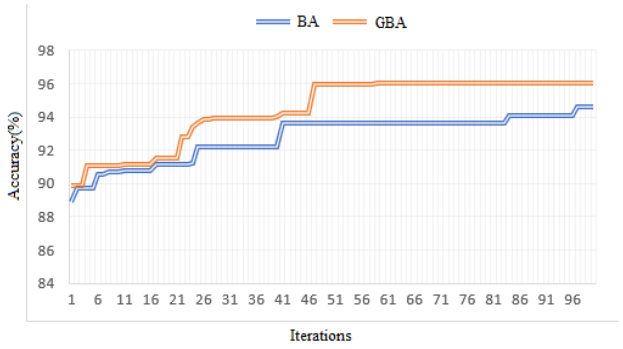

Figure. 4 BA & GBA with stars = 10

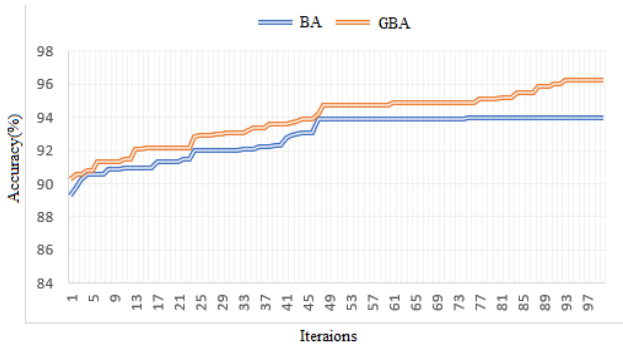

Figure. 5 BA & GBA with stars = 20

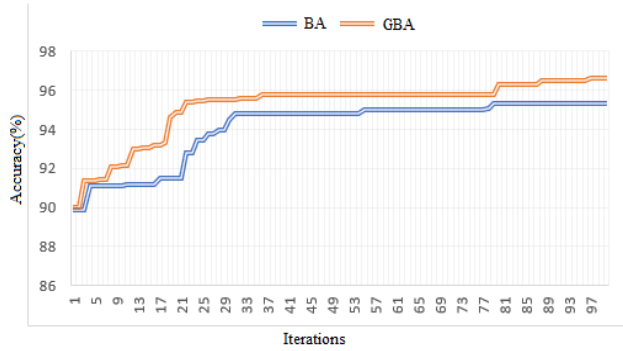

Figure. 6 BA & GBA with stars = 30

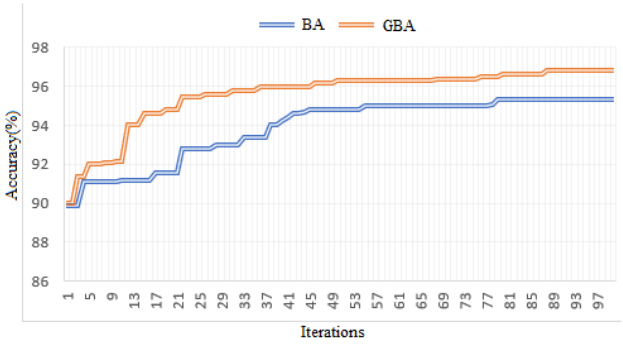

Figure. 7 BA & GBA with stars = 40

iterations. The GBA algorithm demonstrates greater robustness and reliability in identifying network intrusions while minimizing false alarms. These results highlight the effectiveness of the hybrid filter-wrapper feature selection method (GBA) in enhancing the performance of network intrusion detection systems. It is essential to consider these findings when choosing an appropriate algorithm for intrusion detection applications, as it can significantly impact the overall security and reliability of a network. In next subsection, GBA is going to be compared against several state of arts algorithms.

7. Comparison with state-of-arts

In this subsection, we present a comprehensive comparison between the GBA and several other state-of-the-art algorithms for network intrusion detection. The evaluation is based on key performance metrics, including accuracy, detection rate, and false alarm rate. By analyzing the results, we aim to identify the strengths and weaknesses of each algorithm and highlight the advantages of GBA in enhancing intrusion detection systems. Table 8 presents this comparison. The presented study is compared against the other related works based on the same dataset, which NSL-KDD. These studies were utilized different feature selections methods, and different classification algorithms.

The presented table offers a comparative analysis of various models and algorithms for network intrusion detection, focusing on accuracy (A%) and false positive rate (FPR%). Among the examined approaches, GBA stands out with exceptional performance, achieving an impressive accuracy of 96.96% and an exceptionally low false positive rate of 0.89%. One of the key factors contributing to GBA's superiority lies in its enhancement of the initialization step. By applying the GR technique, GBA effectively removes the most unrelated features from the original dataset. This is achieved by calculating the weight for each feature, resulting in their weights being almost zero. Consequently, GBA benefits from a highly refined and relevant feature subset, leading to its outstanding performance in accurately detecting network intrusions while minimizing false alarms. The notable influence of this enhanced initialization distinguishes GBA from other contemporary models and algorithms,

Table 8. Results comparison

Ref	Model	Acc%	FPR%
[53]	Majority Voting + GR , IG , λ^2	85.23	12.8
[54]	MFFSEM + RF	84.33	24.82
[55]	LightGBM	89.79	9.13
[56]	Autoencoder	84.21	N/A
[57]	Stacking	92.17	2.52
[58]	Ensemble + PSO	90.39	1.59
[59]	RPSO	85	N/A
	GBA	96.96	0.89

Table 7. Results of BA and GBA

Itr	Stars	Alg.	Case	SF	RF	ACC (%)	DR (%)	FAR (%)	Mean
100	10	BA	Best	15	26	95.16	96.13	4.81	94.22
			Worst	15	26	94.23	94.14	5.22	
		GBA	Best	**10**	**31**	**96.01**	**98.2**	**3.64**	95.8
			Worst	13	28	95.43	97.71	4.02	
	20	BA	Best	16	25	95.88	96.21	4.26	95.26
			Worst	17	24	94.500	94.4	4.9	
		GBA	Best	**11**	**30**	**96.266**	**98.6**	**2.8**	95.99
			Worst	14	27	95.79	97.9	3.58	
	30	BA	Best	15	26	95.9	96.66	3	95.58
			Worst	14	27	94.90	95.8	3.6	
		GBA	Best	**13**	**28**	**96.63**	**98.94**	**2.1**	96.4
			Worst	15	26	95.9	97.99	3.1	
	40	BA	Best	15	26	96.16	96.2	2.4	95.63
			Worst	16	25	94.90	95.64	3.02	
		GBA	Best	**10**	**31**	**96.96**	**99.1**	**0.89**	96.8
			Worst	12	29	96.02	98	1.79	

positioning it as a promising and dependable solution for network intrusion detection, thereby augmenting the overall security of network systems.

8. Conclusion

In this paper, we present GBA, a hybrid filter-wrapper approach. GBA combines the information gain ratio model with the black hole algorithm and utilizes the Naive Bayes classifier. The goal of GBA is to leverage the efficiency of the filter approach while achieving higher accuracy similar to the wrapper approach. We identify the most relevant features using the information gain ratio, which are then used to replace the randomly selected features during the search initialization in GBA. The main contribution is the strategic integration of the information gain ratio to enhance search initialization in GBA, leading to improved feature selection efficiency and accuracy.

To evaluate the performance of GBA, we compare it with other algorithms such as PSO, GA, FA, and BA using five selected datasets. The results show that GBA outperforms the benchmarking algorithms in terms of accuracy, except for BA, which achieves similar performance to GBA. Statistical tests reveal that GBA achieves significantly lower computation time across all tested datasets. Moreover, the introduced novel feature selection technique, GBA, enhanced intrusion detection system accuracy. GBA's success, achieving 96.96% accuracy and 0.89% false positive rate, is attributed to GR-based initialization, outperforming standard BHA and demonstrating potential for network security enhancement.

Conflicts of interest

The authors declare no conflict of interest.

Author contributions

Conceptualization, SSI and FHT; methodology, SQS and YDS; software, SQS; validation, SSI, FHT, and SQS; formal analysis and investigation, YDS; data curation, SSI and SQS; writing—original draft preparation, SQS and YDS; writing—review and editing, SSI and FHT.

Acknowledgments

This work was supported by Al-Bayan University, College of Technical Engineering.

References

[1] V. Chandola, A. Banerjee, and V. Kumar, “Anomaly detection for discrete sequences: A survey”, *IEEE Trans. Knowl. Data Eng.*, Vol. 24, No. 5, pp. 823-839, 2012, doi: 10.1109/TKDE.2010.235.

[2] S. Agrawal and J. Agrawal, “Survey on anomaly detection using data mining techniques”, *Procedia Computer Science*, pp. 708-713, 2015, doi: 10.1016/j.procs.2015.08.220.

[3] V. Balamurugan and R. Saravanan, “Enhanced intrusion detection and prevention system on cloud environment using hybrid classification

and OTS generation", *Cluster Comput.*, pp. 1-13, 2017, doi: 10.1007/s10586-017-1187-7.

[4] M. Sheikhan and N. Mohammadi, "Neural-based electricity load forecasting using hybrid of GA and ACO for feature selection", *Neural Comput. Appl.*, Vol. 21, No. 8, pp. 1961-1970, 2012, doi: 10.1007/s00521-011-0599-1.

[5] S. Aljawarneh, M. B. Yassein, and M. Aljundi, "An enhanced J48 classification algorithm for the anomaly intrusion detection systems", *Cluster Comput.*, pp. 1-17, 2017, 10.1007/s10586-017-1109-8.

[6] A. J. Malik and F. A. Khan, "A hybrid technique using binary particle swarm optimization and decision tree pruning for network intrusion detection", *Cluster Comput.*, pp. 1-14, 2017, doi: 10.1007/s10586-017-0971-8.

[7] M. Sheikhan, Z. Jadidi, and A. Farrokhi, "Intrusion detection using reduced-size RNN based on feature grouping", *Neural Comput. Appl.*, Vol. 21, No. 6, pp. 1185-1190, 2012, 10.1007/s00521-010-0487-0.

[8] H. Bostani and M. Sheikhan, "Hybrid of binary gravitational search algorithm and mutual information for feature selection in intrusion detection systems", *Soft Comput.*, Vol. 21, No. 9, pp. 1-18, 2015, doi: 10.1007/s00500-015-1942-8.

[9] B. Xue, M. Zhang, and W. N. Browne, "Particle swarm optimisation for feature selection in classification: Novel initialisation and updating mechanisms", *Appl. Soft Comput. J.*, Vol. 18, pp. 261-276, 2014, doi: 10.1016/j.asoc.2013.09.018.

[10] Y. D. Salman and N. L. Hashim, "An Improved Method of Obtaining Basic Path Testing For Test Case Based On UML State Chart", In: *Proc. of International Symposium on Research in Innovation and Sustainability 2014*, pp. 1607-1610, 2014.

[11] Y. D. S. N. L. Hashim, "An Improved Algorithm in Test Case Generation from UML Activity Diagram Using Activity Path", In: *Proc of the 3rd International Conference on Computing and Informatics*, pp. 226-231, 2011.

[12] P. Moradi and M. Rostami, "Integration of graph clustering with ant colony optimization for feature selection", *Knowledge-Based Syst.*, Vol. 84, pp. 144–161, 2015, doi: 10.1016/j.knosys.2015.04.007.

[13] H. Tao, S. M. Awadh, S. Q. Salih, S. S. Shafik, and Z. M. Yaseen, "Integration of extreme gradient boosting feature selection approach with machine learning models: application of weather relative humidity prediction", *Neural Comput. Appl.*, 2022, doi: 10.1007/s00521-021-06362-3.

[14] F. Cui, S. Q. Salih, B. Choubin, S. K. Bhagat, P. Samui, and Z. M. Yaseen, "Newly explored machine learning model for river flow time series forecasting at Mary River, Australia", *Environ. Monit. Assess.*, 2020, doi: 10.1007/s10661-020-08724-1.

[15] N. C. Long, P. Meesad, and H. Unger, "A highly accurate firefly based algorithm for heart disease prediction", *Expert Syst. Appl.*, Vol. 42, No. 21, pp. 8221-8231, 2015, doi: 10.1016/j.eswa.2015.06.024.

[16] C. H. Cheng, T. L. Chen, and L. Y. Wei, "A hybrid model based on rough sets theory and genetic algorithms for stock price forecasting", *Inf. Sci. (Ny).*, Vol. 180, No. 9, pp. 1610-1629, 2010, doi: 10.1016/j.ins.2010.01.014.

[17] A. A. Qasim and A. H. Sallomi, "Design and Analysis of Phased Array System by MATLAB Toolbox", *Al-Kitab J. Pure Sci.*, 2023, doi: 10.32441/kjps.04.01.p5.

[18] I. Guyon and A. Elisseeff, "An Introduction to Variable and Feature Selection", *J. Mach. Learn. Res.*, Vol. 3, No. 3, pp. 1157-1182, 2003, doi: 10.1016/j.aca.2011.07.027.

[19] J. Tang, S. Alelyani, and H. Liu, "Feature Selection for Classification: A Review", *Data Classif. Algorithms Appl.*, pp. 37-64, 2014, doi: 10.1.1.409.5195.

[20] A. A. Ani, "Ant Colony Optimization for Feature Subset Selection", *International Journal of Computer and Information Engineering*, Vol. 1, No. 4, pp. 999-1002, 2007.

[21] S. M. Vieira, J. M. C. Sousa, and T. A. Runkler, "Two cooperative ant colonies for feature selection using fuzzy models", *Expert Syst. Appl.*, Vol. 37, No. 4, pp. 2714-2723, 2010, doi: 10.1016/j.eswa.2009.08.026.

[22] M. M. Kabir, M. Shahjahan, and K. Murase, "A new local search based hybrid genetic algorithm for feature selection", *Neurocomputing*, Vol. 74, No. 17, pp. 2914-2928, 2011, doi: 10.1016/j.neucom.2011.03.034.

[23] C. H. Lin, H. Y. Chen, and Y. S. Wu, "Study of image retrieval and classification based on adaptive features using genetic algorithm feature selection", *Expert Syst. Appl.*, Vol. 41, No. 15, pp. 6611-6621, 2014, doi: 10.1016/j.eswa.2014.04.033.

[24] M. Schiezaro and H. Pedrini, "Data feature selection based on Artificial Bee Colony algorithm", *J. Image Video Process.*, Vol. 1, No. 47, pp. 1-8, 2013.

[25] V. Agrawal and S. Chandra, "Feature Selection using Artificial Bee Colony Algorithm for

Medical Image Classification", In: *Proc. of 2015 Eighth Int. Conf. Contemp. Comput.*, Vol. 1, pp. 2-7, 2015.

[26] B. Xue, M. Zhang, S. Member, and W. N. Browne, "Particle Swarm Optimization for Feature Selection in Classification : A Multi-Objective Approach", *IEEE transactions on cybernetics*, Vol. 43, No. 6, pp. 1656-1671, 2012.

[27] C. C. O. Ramos, A. N. Souza, G. Chiachia, A. X. Falco, and J. P. Papa, "A novel algorithm for feature selection using Harmony Search and its application for non-technical losses detection", *Comput. Electr. Eng.*, Vol. 37, No. 6, pp. 886-894, 2011, doi: 10.1016/j.compeleceng.2011.09.013.

[28] H. H. Inbarani, M. Bagyamathi, and A. T. Azar, "A novel hybrid feature selection method based on rough set and improved harmony search", *Neural Comput. Appl.*, Vol. 26, No. 8, pp. 1859-1880, 2015, doi: 10.1007/s00521-015-1840-0.

[29] A. Hatamlou, "Black hole: A new heuristic optimization approach for data clustering", *Inf. Sci. (Ny).*, Vol. 222, pp. 175-184, 2013, doi: 10.1016/j.ins.2012.08.023.

[30] J. Biesiada and W. Duch, "Feature Selection for High-Dimensional Data: A Kolmogorov-Smirnov Correlation-Based Filter", *Comput. Recognit. Syst.*, Vol. 30, pp. 95-103, 2005, doi: 10.1007/3-540-32390-2_9.

[31] E. Harris, "Information Gain Versus Gain Ratio: A Study of Split Method Biases", *AI&M*, 2002.

[32] L. E. Raileanu and K. Stoffel, "Theoretical comparison between the Gini Index and Information Gain criteria", *Ann. Math. Artif. Intell.*, Vol. 41, No. 1, pp. 77-93, 2004, doi: 10.1023/B:AMAI.0000018580.96245.c6.

[33] Q. Gu, Z. Li, and J. Han, "Generalized Fisher Score for Feature Selection", *CoRR*, Vol. abs/1202.3, No. August, pp. 327-330, 2012, [Online]. Available: http://arxiv.org/abs/1202.3725%5Cnhttp://researchbank.rmit.edu.au/view/rmit:160103

[34] V. Kumar, "Feature Selection: A literature Review", *Smart Comput. Rev.*, Vol. 4, No. 3, 2014, doi: 10.6029/smartcr.2014.03.007.

[35] D. Tomar and S. Agarwal, "Hybrid feature selection based weighted least squares twin support vector machine approach for diagnosing breast cancer, hepatitis, and diabetes", *Adv. Artif. Neural Syst.*, Vol. 2015, 2015.

[36] Z. Hu, Y. Bao, T. Xiong, and R. Chiong, "Hybrid filter–wrapper feature selection for short-term load forecasting", *Eng. Appl. Artif. Intell.*, Vol. 40, pp. 17-27, 2015, doi: 10.1016/j.engappai.2014.12.014.

[37] L. Zhang, L. Shan, and J. Wang, "Optimal feature selection using distance-based discrete firefly algorithm with mutual information criterion", *Neural Comput. Appl.*, pp. 1-14, 2016, doi: 10.1007/s00521-016-2204-0.

[38] A. M. Taha, S. D. Chen, and A. Mustapha, "Bat Algorithm Based Hybrid Filter-Wrapper Approach", *Adv. Oper. Res.*, Vol. 2015, No. 3, p. 1, 2015, doi: 10.1155/2015/961494.

[39] A. P. Piotrowski, J. J. Napiorkowski, and P. M. Rowinski, "How novel is the 'novel' black hole optimization approach?", *Inf. Sci. (Ny).*, 2014, doi: 10.1016/j.ins.2014.01.026.

[40] S. Kumar, D. Datta, and S. K. Singh, "Black Hole Algorithm and Its Applications", in *Studies in Computational Intelligence*, pp. 147-170, 2015, doi: 10.1007/978-3-319-11017-2_7.

[41] S. Q. Salih, "A New Training Method Based on Black Hole Algorithm for Convolutional Neural Network", *J. Sourthwest Jiaotong Univ.*, Vol. 54, No. 3, pp. 1-10, 2019, doi: 10.1002/9783527678679.dg01121.

[42] S. Q. Salih, A. L. Khalaf, N. S. Mohsin, and S. F. Jabbar, "An optimized deep learning model for optical character recognition applications", *Int. J. Electr. Comput. Eng.*, 2023, doi: 10.11591/ijece.v13i3.pp3010-3018.

[43] K. H. Chen, L. F. Chen, and C. T. Su, "A new particle swarm feature selection method for classification", *J. Intell. Inf. Syst.*, No. 510, pp. 1-24, 2014, doi: 10.1007/s10844-013-0295-y.

[44] J. Yang and V. Honavar, "Feature subset selection using a genetic algorithm", *IEEE Intell. Syst. their Appl.*, Vol. 13, No. 2, pp. 44-49, 1998, doi: 10.1109/5254.671091.

[45] E. Emary, H. M. Zawbaa, K. K. A. Ghany, A. E. Hassanien, and B. Parv, "Firefly Optimization Algorithm for Feature Selection", In: *Proc. 7th Balk. Conf. Informatics Conf. - BCI '15*, pp. 1-7, 2015, doi: 10.1145/2801081.2801091.

[46] A. M. Taha, A. Mustapha, and S. Chen, "Naive Bayes-Guided Bat Algorithm for Feature Selection", *Sci. World J.*, Vol. 2013, pp. 1-9, 2013, doi: 10.1155/2013/325973.

[47] A. S. K. Pathan, *The State of the Art in Intrusion Prevention and Detection*. CRC Press, Taylor & Francis Group, 2014. [Online]. Available: https://books.google.com/books?id=MJrNBQAAQBAJ

[48] S. Fayssal, S. Hariri, and Y. A. Nashif, "Anomaly-Based Behavior Analysis of Wireless Network Security", *2007 Fourth Annu. Int. Conf. Mob. Ubiquitous Syst. Netw. Serv.*, pp. 1-8, 2007, doi: 10.1109/MOBIQ.2007.4451054.

[49] S. W. Lin, K. C. Ying, C. Y. Lee, and Z. J. Lee,

"An intelligent algorithm with feature selection and decision rules applied to anomaly intrusion detection", *Appl. Soft Comput. J.*, Vol. 12, No. 10, pp. 32850-3290, 2012, doi: 10.1016/j.asoc.2012.05.004.

[50] R. Lippmann, J. W. Haines, D. J. Fried, J. Korba, and K. Das, "1999 DARPA off-line intrusion detection evaluation", *Comput. Networks*, Vol. 34, No. 4, pp. 579-595, 2000, doi: 10.1016/S1389-1286(00)00139-0.

[51] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, "A detailed analysis of the KDD CUP 99 data set", In: *Proc. of IEEE Symposium on Computational Intelligence for Security and Defense Applications, CISDA 2009*, 2009. doi: 10.1109/CISDA.2009.5356528.

[52] S. H. Kang and K. J. Kim, "A feature selection approach to find optimal feature subsets for the network intrusion detection system", *Cluster Comput.*, Vol. 19, No. 1, pp. 325-333, 2016, doi: 10.1007/s10586-015-0527-8.

[53] S. Krishnaveni, S. Sivamohan, S. Sridhar, and S. Prabhakaran, "Network intrusion detection based on ensemble classification and feature selection method for cloud computing", *Concurr. Comput. Pract. Exp.*, 2022, doi: 10.1002/cpe.6838.

[54] H. Zhang, J. L. Li, X. M. Liu, and C. Dong, "Multi-dimensional feature fusion and stacking ensemble mechanism for network intrusion detection", *Futur. Gener. Comput. Syst.*, Vol. 122, pp. 130-143, Sep. 2021, doi: 10.1016/j.future.2021.03.024.

[55] J. Liu, Y. Gao, and F. Hu, "A fast network intrusion detection system using adaptive synthetic oversampling and LightGBM", *Comput. Secur.*, 2021, doi: 10.1016/j.cose.2021.102289.

[56] C. Ieracitano, A. Adeel, F. C. Morabito, and A. Hussain, "A novel statistical analysis and autoencoder driven intelligent intrusion detection approach", *Neurocomputing*, 2020, doi: 10.1016/j.neucom.2019.11.016.

[57] B. A. Tama, L. Nkenyereye, S. M. R. Islam, and K. S. Kwak, "An enhanced anomaly detection in web traffic using a stack of classifier ensemble", *IEEE Access*, 2020, doi: 10.1109/ACCESS.2020.2969428.

[58] M. H. L. Louk and B. A. Tama, "PSO-Driven Feature Selection and Hybrid Ensemble for Network Anomaly Detection", *Big Data Cogn. Comput.*, Vol. 6, No. 4, p. 137, Nov. 2022, doi: 10.3390/bdcc6040137.

[59] "Feature Selection of The Anomaly Network Intrusion Detection Based on Restoration Particle Swarm Optimization", *Int. J. Intell. Eng. Syst.*, Vol. 15, No. 5, pp. 592-600, 2022, doi: 10.22266/ijies2022.1031.51.

Notation list

N	Symbol	Meaning
1	x_i, x_{BH}	The position of the star
2	f_i	the fitness value
3	IG	Information Gain
4	GR	Gain Ratio
5	IV	Intrinsic Value
6	$H(S)$	Entropy Function
7	Err	Error Rate
8	$\#F$	Number of Selected Features
9	TP, TN	True Positive and Negative
10	FP, FN	False Positive and Negative
13	ACC	Classification Accuracy
14	DR	Detection Rate
15	FAR	False Alarm Rate