BULGARIAN ACADEMY OF SCIENCES

CYBERNETICS AND INFORMATION TECHNOLOGIES • Volume **22**, No 4
Sofia • 2022 Print ISSN: 1311-9702; Online ISSN: 1314-4081
DOI: 10.2478/cait-2022-0040

Hybrid Feature Selection Method for Intrusion Detection Systems Based on an Improved Intelligent Water Drop Algorithm

Esra'a Alhenawi[1], *Hadeel Alazzam*[2], *Rizik Al-Sayyed*[3], *Orieb AbuAlghanam*[4], *Omar Adwan*[1,4]

[1]*Department of Software Engineering, Al-Ahliyya Amman University, Amman, Jordan*
[2]*Department of Intelligence Systems, Al-Balqa Applied University, Al-Salt, Jordan*
[3]*King Abdullah II School for Information Technology, The University of Jordan, Amman, Jordan*
[4]*Department of Computer Science, The University of Jordan, Amman, Jordan*
E-mails: e.alhenawi@ammanu.edu.jo hadeel.alazzam@bau.edu.jo O.AbuAlganam@ju.edu.jo r.alsayyed@ju.edu.jo adwanoy@ammanu.edu.jo adwanoy@ju.edu.jo

Abstract: *A critical task and a competitive research area is to secure networks against attacks. One of the most popular security solutions is Intrusion Detection Systems* (*IDS*). *Machine learning has been recently used by researchers to develop high performance IDS. One of the main challenges in developing intelligent IDS is Feature Selection* (*FS*). *In this manuscript, a hybrid FS for the IDS network is proposed based on an ensemble filter, and an improved Intelligent Water Drop* (*IWD*) *wrapper. The Improved version from IWD algorithm uses local search algorithm as an extra operator to increase the exploiting capability of the basic IWD algorithm. Experimental results on three benchmark datasets "UNSW-NB15", "NLS-KDD", and "KDDCUPP99" demonstrate the effectiveness of the proposed model for IDS versus some of the most recent IDS algorithms existing in the literature depending on "F-score", "accuracy", "FPR", "TPR" and "the number of selected features" metrics.*

Keywords: *Cybersecurity, Decision tree, Ensemble filter, Feature selection, Intelligent Water Drop, Intrusion Detection System.*

1. Introduction

Since society is becoming more technologically reliant than ever before, cybersecurity is one of the key research topics nowadays [1]. Cybersecurity represents the process of defending networks, computers, servers, mobiles, devices, electronic systems, and data from malicious attacks, and this defending can be achieved using many security solutions such as cryptography [2, 3], firewalls, authentication techniques [4], prevention systems, or intrusion detection systems (henceforth IDS) [5]. IDS is a software application designed for detecting malicious content through monitoring network traffic activities to report suspicious network activity [6].

IDS can be classified into four categories depending on the detection method, where detection systems may depend on signature, anomaly, specification, or hybrid detection methods [7]. In the first category, detecting an abnormal behaviour is achieved by using well-known patterns (signatures) for the previous threats in the database [8]. When it comes to well-known, popular threats, this category provides better performance and very strong outcomes, but it is unable to identify the new unseen attacks or the zero-day attacks [7].

The second category is also known as outlier detection, and it is based on detecting ordinary patterns or abnormal types of data. In other words, an anomaly is a data point or odd observed data that is too far apart from other data points in a dataset. This type can detect previously undiscovered attack incidents, but the percentage of activities that have been wrongly defined as attacks is typically high [7, 9].

In specification-based detection categories, a human expert will be depended on to construct the desired template as manual specifications for evaluating the valid behaviour of a device. Using this type of system, benign behaviours that have not been previously observed are not flagged as intrusions [7].

Hybrid detection systems exploit the advantages of different intrusion detection methods in order to implement a strong framework for detecting intrusions. This type commonly uses a combination of a signature-based detection system and an anomaly-based model for improving accuracy with a low level of false positive rates for signature-based methods [7].

FS methods aim to select the optimal subset of features in order to improve the machine-learning model's performance in terms of improving the model performance and accelerating the training speed by decreasing the number of features. FS methods can use one of the machine learning categories, i.e., supervised, semi-supervised, or unsupervised [10, 11]. Moreover, FS methods can be classified into five categories based on the selection strategy: "filter, wrapper, embedded, hybrid, and ensemble" FS methods.

In the filter category, the selected features will be chosen based on the highest ranks at a specific threshold, while in the wrapper, the selected set of features will be taken based on the best-achieved classification results [11]. In the embedded methods, the selected set of features will be automatically chosen as a part of the classification process. Ensemble methods aim to solve instability and perturbation issues in many individual FS methods, it has two types: homogeneous, and heterogeneous [12]. Hybrid FS methods select a subset of features by combining more than one FS method from different selection strategies for exploiting their advantages simultaneously [13].

In this paper, a hybrid FS method for IDS is developed. This method combines an ensemble filter with a wrapper based on an IWD algorithm with two improvements related to the next feature selection strategy and increasing the exploitation capability of IWD by adding some of LS algorithms.

This paper proposes the following main contributions:

- It develops a hybrid FS method based on an ensemble filter and an improved IWD as a wrapper for an IDS.
- It improves the exploitation capability of the basic IWD algorithm by adding three local search algorithms (Tabu Search (TS), Novel Local Search Algorithm (NLSA), and Hill Climbing (HC).
- It improves the FS process update using correlation coefficient (cc) filter as a Heuristic UnDesirability (HUD) for next node selection to eliminate the redundant features.

The remainder of this paper is organized as follows: Section 2 presents a brief review of the latest works that have been conducted on using feature selection for IDS. Section 3 displays the proposed method for IDS. Section 4 illustrates the experimental setups. The experimental results and discussion are presented in Section 5. Finally, Section 6 draws the conclusion.

2. Related works

FS methods mark an important stage before building the machine-learning model. Evolutionary algorithms and swarm intelligence algorithms, such as Genetic Algorithms and ant colonies, have been successfully employed to solve the problem of feature selection [14]. These methods are based on understanding the biological behaviour of animals/insects and using this understanding to solve optimization problems. Moreover, the metaheuristic approach is widely used as a feature selection algorithm, as in [15], that uses an ant colony optimization, while in [16], three algorithms based on particle swarm optimization have been proposed based on fuzzy rough fitness functions. In [17], an ensemble FS has been proposed based on the genetic algorithm.

FS has two main objectives. The first is enhancing the classifier's performance and obtaining a higher accuracy rate, and the second is reducing a subset of selected features. Therefore, FS should be considered a multi-objective problem.

Various nature-inspired metaheuristic search algorithms have been implemented in the literature for feature reduction, which aims to get better data visualization and performance results. Genetic Algorithms (GAs) are the most commonly techniques used in the features selection problem, in which the GA-based approach has been applied with various classifiers for feature selection and building classification models, such as the GA-SVM models [18-20] and the GA-XGBoost models [21, 22].

A new class of models that is inspired by nature is known as Swarm Intelligence (SI). It has emerged from different natural swarm behaviours such as Ant Colony Optimization (ACO), Particle Swarm Optimization (PSO), and Grasshopper Optimisation Algorithm (GOA). This class has been proposed and applied in the feature selection research fields and domains. For instance, in medical fields, researchers in [18, 20, 22] apply GA-based models, while [23, 24] have adopted the PSO-based approach. In addition, ACO is used in text categorization and image annotation [25] and in the Road Sign Detection and Recognition (RSDR) system [26].

The research conducted on the adaptation and application of metaheuristics to FS problems is still ongoing, and in the literature, several new approaches have been proposed. The Intelligent Water Drop (IWD) algorithm is an example of a recent stochastic swarm-based approach that helps solve FS optimization problems in various domains such as human motion detection and motor fault detection [27].

Another research domain where several researchers have proposed various hybrid models for classification to reduce the dimension of features subset and achieve high performance is IDS. This paper investigates the applicability of various nature-inspired metaheuristic search algorithms, especially (IWD) which is a feature selection method for optimizing IDS. Table 1 below summarizes all these efforts.

For instance, [28] uses the IWD algorithm to select the feature subset along with SVM as a classifier to evaluate the selected features to improve their IDS. They perform their experiments using the KDD CUP'99 dataset, and the obtained results have been compared to earlier designs tested on the same dataset such as [19] and [29] which apply GA with the SVM classifier. The results show that IWD is more efficient in reducing feature dimensions and obtaining higher results, in which it has been able to reduce the number of features from 41 to 9, and the detection rate is 99.41%. [29], however, reduced features to 10.

Recently, [30] apply the PSO Algorithm along with the Random Forest (RF) classifier to select and evaluate the selective features of the NSL-KDD dataset, which is a modified version of the KDDCUPP99 dataset. Their experiments have achieved distinguished results using 10 features.

Table 1. Summary of hybrid models in literature for optimizing the subset of features in intrusion detection systems

Reference	Hybrid model	Performance metrics	Dataset	Number of features
[19]	GA with SVM	Accuracy, FP rate, TP rate, number of selected features	KDDCUP^{99}and UNSW-NB15	-
[29]	GA with SVM	TPR, FPR, Accuracy, Precision, Recall, F-measure, and ROC	KDDCUP^{99}	10
[28]	IWD with SVM	Detection Rate (DR), False Alarm Rate (FAR), accuracy, and precision	KDDCUP^{99}	9
[30]	PSO with RF	Accuracy, precision, FP rate, and detection rate	NSL-KDD	10
[10]	PIO with DT	Accuracy, detection rate, false alarms, and F-score	KDDCUP^{99}, NSL-KDD, and UNSW-NB15	7 for KDDCUP^{99}, 5 for both NSL-KDD, and UNSW-NB15
[31]	MFO with DT	Accuracy, sensitivity, detection rate, and F-score	CIC2017	4

However, [10] develop a hybrid model of Pigeon Inspired Optimizer (PIO) and Decision Tree (DT) to produce an IDS feature selection algorithm. They use three common IDS datasets and four evaluation metrics including accuracy, F-score, detection rate, and false alarms for evaluation. The obtained results show that their algorithm has achieved an accuracy of 0.96 and a detection rate of 0.98 in KDDCUPP99. On the other hand, in the NSL-KDD dataset, they have gained an accuracy of 0.0.88 and a detection rate of 0.86. In addition, the UNSW-NB15 dataset has an accuracy of 91% and a detection rate of 89%.

Furthermore, [31] utilize Moth Flame Optimization (MFO) for the feature selection process and the DT classifier to evaluate the selected subset of features. The algorithm has been evaluated using the CIC2017 Dataset. Results show that the proposed model has achieved a 100% detection rate using four features only.

3. The proposed hybrid FS method for IDS

This section presents a detailed description of the proposed hybrid FS method as shown in Fig. 1. The proposed method consists of two main phases. The first phase is called an ensemble-filter-based phase, and it is illustrated in Subsection 3.1. The second phase presents an improved IWD-based wrapper, and it is illustrated in Subsection 3.2.

3.1. First phase of the proposed method: An Ensemble Filter

In the first phase, an ensemble filter is applied based on two well-known ranking-based filters, named ReliefF and Fisher score. This phase aims to enhance the performance of the model by reducing number of features that will be passed to the wrapper phase by eliminating the irrelevant features from the original set of features (S).

Initially, the "Relief F" filter is applied over all original sets of features (S) as an input to produce a specific number of features based on the top-ranked features named (Re). Then, the "Fisher score" filter has been applied to S in order to produce another set of features that is called Fi. After that, the set of features that have been produced by each filter is aggregated to be gathered using the "Union" process to produce the final set of features that is called F, and it will be passed to the next phase (wrapper phase) as illustrated in Algorithm 1 and Fig. 1.

Algorithm 1. Applying the proposed hybrid feature selection method for IDS pseudo code

```
Input: Microarray dataset of S features
Output: Best feature subset TB and its performance measures Q(TB)
Step 1. Fi ← Fisher score over S features
Step 2. Re ← Relief F over S features
Step 3. F ← (Fi U Re)
Step 4. Initializing static and dynamic parameters for IWD
Step 5. while ($iter_{count}$ < $iter_{max}$) do
Step 6.     Spread the predefined IWDs randomly over the filtered features (F)
Step 7.   For each IWD do    \\ where, each IWD starts from a specific feature
Step 8.    while (number of features in IWD features list = N) do
Step 9.       Compute the probability for each (F – 1) unlisted features to be selected as a next feature using Equation (3)
Step 10.       Select the feature with the highest probability of selection as a next feature in IWD features list
Step 11.         Update IWD features list by adding the selected feature
Step 12.         Update velocity and soil value after each transition based on Equation (4)   Equation (6), and Equation (7)
Step 13.    end while
Step 14.    Compute a fitness function (F-Score) using the features in IWD features list.
Step 15. End for
Step 16. Compute IB based on Equation (9)
Step 17. Update the soil through the selected best path based on Equation (10)
Step 18. LS Solution ← [TS(IB) or NLSA(IB) or HC(IB)]
Step 19.        If (LS solution better than IB)
Step 20.    IB ← LS Solution
Step 21.    If (IB better than TB)
Step 22.    TB ← IB based on Equation (11).
Step 23. end while
Step 24. Return best feature set TB and Q(TB)
```

RelifF is one of the most common ranking-based filters developed by Kira and Rendell as an extension of the Relief algorithm [32]. This filter computes the weight of each feature as an average of all k-nearest neighbors from the same class and the different classes for each random instance that is selected each time [33]. This filter works by calculating weight for each feature as an average of all neighbors from the same class and neighbors from different classes [34].

Fisher Score is a ranking-based filter that selects the set of features that provides more distinction between classes by selecting the top-ranked features that guarantee to maximize the distance between data points in different classes and, at the same time, minimize the distance between data points from the same class [35]. Fisher score is a special case from the similarity-based FS methods and can be calculated from a Laplacian Score. Fisher score is calculated for each feature f_j depending on the next equation, then the top set of features that have a larger fisher score value will be selected [36]:

$$\text{Fisher score}(f_j) = \frac{1}{\text{Laplacian\_score}(f_j)}. \tag{1}$$

3.2. Second phase of the proposed method: An improved IWD wrapper

In this paper, two improvements are applied to the original IWD algorithm in order to increase the exploitation capability of the IWD algorithm, decrease the risks of the local optima problem, and guide the IWD search to eliminate the redundant features.

First improvement is achieved by adding one of three different LS algorithms in each iteration from IWD, while the second improvement is made by using the correlation coefficient filter as a Heuristic UnDesirability (HUD) for eliminating the selection of redundant features.

3.2.1. First Improvement in the proposed IWD based wrapper phase

In the proposed FS method, three LS algorithms are used for increasing the exploitation capability of the IWD algorithm as a first improvement in the original IWD. LS algorithms aim to search for a better solution around the best solution reached from each iteration of the original IWD algorithm. This section briefly displays these LS algorithms:

1. Tabu Search (TS) is developed by [37] in 1998. It starts with a specific solution and then updating it based on neighborhood evaluations such as hill climbing [38]. However, TS avoids cycling movements and entrapment in local optima as search guides by the Tabu list, which works as an adaptive memory that stores all visited points [39].

2. Novel Local Search Algorithm (NLSA) is proposed by [40]. It works by selecting and flipping a specific number (*n*) of features randomly from a specific solution for a specific number of iterations in order to improve the current solution.

3. Hill Climbing (HC) is a greedy local search algorithm that starts with a specific solution and continuously searches in its neighborhood for the purpose of finding the best solution to a specific problem. This algorithm terminates when it reaches the point that no neighbors have a better value [41].

3.2.2. Second Improvement in the proposed IWD based wrapper phase

This improvement aims to eliminate the selection of redundant features by applying a fast correlation coefficient filter as a HUD value that is used for selecting the next feature in each IWD solution list. HUD is proportionate to an amount of soil, which IWD loads through its movement over a path from a specific feature $f1$ to the next feature $f2,$ and $\Delta\text{soil}(f1, f2)$ based on Equation (7). On the other hand, IWD's current soil and soil $(f1, f2)$ depend on $\Delta\text{soil}(f1, f2)$ value. HUD has a high value when $f1$ and $f2$ features are highly correlated, i.e., $f1$ and $f2$ features are most likely to be redundant. As a result, the probability for selecting $f2$ as the next feature in the IWD list will be decreased, where $\Delta\text{soil}(f1, f2)$ will be small based on Equation (7), and consequently, $\text{soil}(f1, f2)$ will be small. HUD (*f*1, *f*2) has an inverse relationship with the probability of choosing *f*2 as a next feature as shown in the equation

$$\text{HUD}(f1, f2) = \alpha \frac{1}{P(f1, f2)}. \tag{2}$$

The proposed hybrid FS method for IDS has eleven steps:

Step 1. In the beginning, an ensemble filter takes the original dataset with S features as an input. It then returns as a filtered set of features F that contains the most relevant set of features. This forms the input of the wrapper phase.

Step 2. The wrapper stage starts by defining the static and dynamic parameters of the IWD algorithm, which has been originally developed by Shah-Hossein [42].

The static parameters include:

- The maximum number of experiment iterations (iter_{max}) and tunes for getting better results.
- The current number of iterations ($\text{iter}_{\text{count}}$).
- The maximum number of intelligent water drops equals the number of filtered features that are generated from the filtering phase in the proposed hybrid feature selection method F.
- The initial soil carried by each IWD and its initial velocity. These values tune experimentally for getting better results and will be initially equal to all IWDs.
- Static parameters for soil and velocity updating $a_{\text{s}}, b_{\text{s}}, c_{\text{s}}, a_{\text{v}}, b_{\text{v}}$ and c_{v}, respectively. All these parameters are equal to one.

- Global and local soil updating parameters ρ^{IWD} and ρ^{n}, respectively.
- Quality of the total best solutions $Q(\text{TB})$
- The initial amount of soil $\text{init}^{\text{soil}}$. This value is a user-selected value that tunes experimentally to get the best results. Soil between any two features such as *f*1 and *f*2 equal $\text{init}^{\text{soil}}$, as shown in the next equation [42].

(3) $$\text{Soil}(f1, f2) = \text{init}^{\text{soil}}.$$

The dynamic parameters include:

- The velocity of each IWD, which equals the initial velocity $\text{vel}^{\text{IWD}} = \text{Init}^{\text{Vel}}$, and then it is updated at each transition of IWD.
- The soil carried by each IWD, which is updated by moving the drop starting from an initial value that has been tuned experimentally $\text{IWD} - \text{Soil} = \text{soil}^{\text{IWD}}$.
- The subset of the selected features for each drop, which is initially empty $\text{SubSet}(\text{IWD}_i)$.

Step 3. The predefined water drops spread randomly over input features, where each drop starts from a specific feature as the first selected feature and continues its search as an independent agent in order to find the most informative N features.

Step 4. Each drop provides the next feature $f2$ with the highest probability to selection based on Equation (4) [42] to its $\text{SubSet}(\text{IWD}_i)$, starting from a specific feature $f1$. The drop's movement continues until selecting N features.

(4) $$p_{f1}^{\text{IWD}}\,(f2) = \frac{f\left(\text{Soil}(f1, f2)\right)}{\sum_{fi \notin \text{SubSet}(\text{IWD}_i)}\left(f\left(\text{Soil}(f1, f_i)\right)\right)},$$

where $i \leq F - 1$, such that,

(5) $$f\left(\text{Soil}(f1, f2)\right) = \frac{1}{\varepsilon s + g\left(\text{Soil}(f1,\ f2)\right)},$$

$$g\left(\text{Soil}(f1, f2)\right) =$$

$$= \begin{cases} \text{Soil}(f1, f2) \ \text{ if } \ \text{minsoil}_{f_i \notin \text{Subset(IWD)}}(f1, f2) \leq 0, \\ \text{Soil}(f1, f2) - \text{minsoil}_{f_i \notin v_c(\text{IWD})}(f1, f2) \qquad \text{else.} \end{cases}$$

Step 5. After each movement of the drop, its velocity is updated based on Equation (6) [42]. In addition, soil may be carried by this water drop, and the soil's value between the current feature and the next selected feature is updated using Equation (10) and Equation (9) [42], respectively. All previous values depend on the current velocity of the water drop and HUD, as can be noticed from Equation (7) [42]. HUD in this paper represents by the correlation coefficient filter value for *f*2 as shown in Equation (8).

(6) $$\text{vel}^{\text{IWD}}\,(t+1) = \text{vel}^{\text{IWD}}\,(t) + \frac{a_v}{b_v + c_v.\text{Soil}^2(f1, f2)},$$

where, $\text{vel}^{\text{IWD}}\,(t+1)$ is the current IWD's velocity after updating based on the current movement,

(7) $$\Delta\text{soil}(f1, f2) = \frac{a_s}{b_s + c_s.\text{time}^2\left(f1, f2;\, \text{vel}^{\text{IWD}}\,(t+1)\right)},$$

where

$$\text{time}^2\left(f1, f2;\ \text{vel}^{\text{IWD}}\,(t+1)\right) = \frac{\text{HUD}(f2)}{\text{vel}^{\text{IWD}}\,(t+1)},$$

where

(8) $$\text{HUD}(f2) = \text{CC}(f2),$$

(9) $$\mathrm{Soil}(f1,f2) = (1-P_n).\mathrm{Soil}(f1,f2) - P_n\,.\Delta\mathrm{Soil}(f1,f2),$$

(10) $$\mathrm{Soil}^{\mathrm{IWD}} = \mathrm{Soil}^{\mathrm{IWD}}.\Delta\mathrm{Soil}(f1,f2).$$

Step 6. At the end of each iteration, each drop has a specific subset of features with (N) features, and they are evaluated based on the used fitness functions (F-score), and later, the solution that provides the best F-score value is considered the current Iteration's Best solution IB. See the next equation [42]:

(11) $$\mathrm{IB} = \mathrm{arg}^{\max}_{\forall\mathrm{Subset}(\mathrm{IWD}_i)}\, F - \mathrm{Score}\ (\mathrm{Subset}\ (\mathrm{IWD}_i)).$$

Step 7. The soil found in the path that consists of the set of features that provides the best F-score value using the next equation should be Updated [42]:

(12) $$\mathrm{Soil}(f\ 1,f\ 2) = (1 + P_{\mathrm{IWD}}).\mathrm{Soil}(f1,f2) - \\ -P_{\mathrm{IWD}}.\frac{1}{N-1}.\mathrm{Soil}^{\mathrm{IWD}}_{\mathrm{IB}}\ \forall\ (f1,f2) \in \mathrm{IB},$$

where, N is the number of features in IB.

Step 8. One of the local search algorithms (TS, NLSA, or HC) that has been added at the end of each iteration from the original IWD algorithm should be applied for increasing the IWD exploitation capability over the current iteration best solution IB. Then, if the solution of the used Local Search algorithm (LS solution) provides a better F-score value than IB, IB will be replaced by the LS solution as shown in the equation

(13) $$\mathrm{IB} = \begin{cases}\text{LS solution if } Q(\text{LS solution}) > Q(\text{ IB}),\\ \text{IB} \qquad\qquad\qquad\qquad\qquad \text{otherwise.}\end{cases}$$

Step 9. After the end of all iterations, the Final Solution TB, which stores the most optimal subset of features that are found so far, is updated after each iteration with a better solution if it is found or still unchanged based on the equation [42]

(14) $$TB = \begin{cases}\text{TB} \quad \text{if} \quad Q(\text{TB}) \geq Q(\text{IB}),\\ \text{IB} \qquad\qquad\qquad \text{otherwise.}\end{cases}$$

Step 10. Steps from Step 3 to Step 8 should be repeated until reaching the predefined ($\mathrm{iter}_{\max}$).

Step 11. The best subset of features that are found so far TB, and performance measures representing by "F-score", "Accuracy", "TPR", and "FPR" metrics, using the selected set of features $Q(\mathrm{TB})$ should be returned after the end of all iterations.

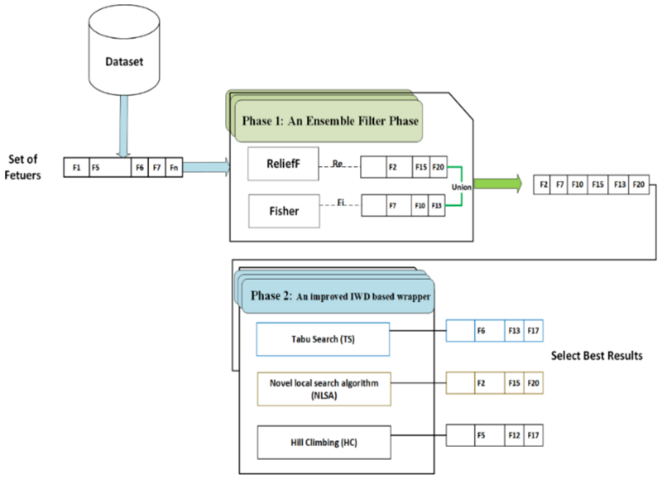

Fig. 1. The Proposed Hybrid FS Algorithm

4. Experimental setup

In this paper, the performance of the proposed hybrid FS method for IDS is evaluated by three common benchmark datasets: KDDCUPP99[43], NSL-KDD [44], and UNSW-NB15 [45] with DT for classification. The proposed method has been compared to some of the FS algorithms in the literature for IDS such as Pigeon Inspired Optimizer (PIO), Genetic Algorithm (GA), and Particle Swarm Optimization (PSO) in terms of accuracy, TPR, FPR, F-score, and the number of selected features as an evaluation metrics. The obtained results represent an average of performing the same experiment thirty times.

4.1. Datasets

This subsection describes the datasets used in this paper.

KDDCUP^{99} represents an enhancement of the DARPA dataset used for implementing IDS. It consists of "4,898,431", and "311,431" records in training and testing sets, respectively. Here, four types of attacks can be simulated including "Denial of Service", "unauthorized access to local superuser root", "unauthorized access from a remote machine", and "surveillance and other probing attacks" [46]. The KDDCUPP^{99} has 41 features from three categories, mainly: content features, traffic features, and basic features. These features are presented and illustrated clearly in [10].

UNSW-NB15 is used to simulate real and contemporary attack models. It contains 49 features and simulates nine types of attacks including "DOS", "ShellCode", "Worms", "Fuzzers", "Backdoors", and "Exploits" [45].

NSL-KDD is an improved version of KDDCUP^{99} that does not have redundant connections in both training and testing sets. It has forty one features and 125,973 connections in the train set versus 22,544 connections in the test set [47].

4.2. Data Pre-processing

This can be achieved by completing three main phases:

- **Data Cleaning.** Data pre-processing starts by removing the missing values and duplicating records in order to guarantee classification fairness.
- **Data Translation.** In this phase, the class column inputs are translated to binary values, where normal records are labelled (0) and attack records are labelled (1). In addition, all non-numeric data are translated to numeric values.
- **Data Normalization.** All used datasets are normalized using the next equation [48] for scaling all data values into a range of [0, 1] in order to eliminate the classifier bias problem for achieving a better classification performance [49]:

$$X_{\text{normalized}} = \frac{X - X_{\min}}{X_{\max} - X_{\min}}. \tag{15}$$

4.3. Classifier

Decision Tree (DT) classifies a population like tree branch segments. DT constructs an inverted tree with internal, root, and leaf nodes. It can handle and interpret the interaction between features effectively in large, complicated datasets with simple structures compared to other classifiers. DT works by dividing the dataset into training and testing sets. It uses the training set to build a model and the testing set to test the trained model's performance [50]. In this work, DT is used as a classifier to train the model based on the selected subset of features obtained using the proposed FS method to be able to classify the normal versus attacks classes in the testing set.

4.4. Evaluation metrics

This section presents the evaluation metrics that are used for evaluating the proposed hybrid FS method.

Step 1. Accuracy. It accounts the instances classified correctly as shown in the next equation [51]:

$$\text{Accuracy} = \frac{\text{TP+TN}}{\text{TP+TN+FP+FN}}. \tag{16}$$

Step 2. F-score. It measures the effects of both precision and recall [52], as shown in the equation

$$\text{F-score} = \frac{2*\text{TP}}{2*\text{TP+FP+FN}}. \tag{17}$$

Step 3. True Positive Rate (TPR). It measures how often the classifier correctly detects the attacks' instances from all actual attacks instances [51], as shown in equation

$$\text{Detection Rate (TPR)} = \frac{\text{TP}}{\text{TP+FN}}. \tag{18}$$

Step 4. False Positive Rate (FPR) or False Alarm Ratio (FAR). It measures how often the classifier wrongly predicts the normal instances as attack instances from all actual normal instances [53], as shown in the equation

$$\text{False Alarms (FPR)} = \frac{\text{FP}}{\text{TN+FP}}. \tag{19}$$

All mentioned metrics depend on four parameters:

- True Positive (TP) represents a number of records correctly predicted as "attack".
- True Negative (TN) represents a number of records correctly predicted as "normal".
- False Positive (FP) represents a number of records wrongly predicted as "attack".
- False Negative (FN) represents a number of records wrongly predicted as "normal".

5. Experimental results and discussion

This section is devoted to displaying the results of applying the proposed FS algorithm versus some FS algorithms from the literature used for IDS such as PIO, PSO, and GA. Three benchmark IDS datasets have been used for evaluation, including UNSW-NB15, NLS-KDD, and KDDCUP^{99}. For each dataset, the results are achieved from the proposed algorithm, and some of the most recent algorithms from the literature are presented in three ways:

– a table listing the performance indicators' numerical values for each algorithm;

– a table that presents the set of features selected from each algorithm;

– and a plot for simplifying the readability of previous results for readers.

All results represent the average of 30 runs for each algorithm. The following subsections display the results reached using each dataset.

5.1. UNSW-NB15 results

Table 2 illustrates the results that are obtained using the set of features that are selected by ten examined algorithms presented in Table 3 for training and testing the classifier over the "UNSW-NB15" dataset, depending on accuracy, TPR, and F-score metrics.

Table 2. UNSW-NB15 dataset results using the DT classifier depending on Accuracy, TPR, FPR, and F-score metrics

Model.	Method	Accuracy±STDV	TPR ±STDV	FPR ±STDV	F-score ±STDV
The proposed model without LS	PHFS-IWD	0.879 ± 0.0002	0.934 ± 0.0002	0.29 ± 0.0002	0.821 ± 0.0002
The proposed model with TS	PHFS-IWDTS	0. 895 $\pm 1.13\times10^{-16}$	0.919 ± 0.000	0.157 ± 0.000	0.834 $\pm 2.3\times10^{-16}$
The proposed model with NLSA	PHFS-IWDNLSA	0.925 ± 0.0001	0.988 ± 0.00008	0.027 ± 0.0001	0.871 ± 0.0002
The proposed model with HC	PHFS-IWDHC	0.892 ± 0.0004	0.969 ± 0.0001	0.0135 ± 0.000	0.847 ± 0.0005
From [54]	GA-RF	0.921 ± 0.000	NA	0.016 ± 0.0000	NA
From [55]	IGRF-RFE	0.842 ± 0.000	0.0842	NA	0.829 ± 0.000
From [10]	PIO	0.913 ± 0.0003	0.897 ± 0.0002	0.052 ± 0.0004	0.904 ± 0.0002
From [56]	Rule-Based	0.652 ± 0.000	0.903 ± 0.000	0.02 ± 0.000	0.681 ± 0.000
From [57]	Wrapper-Based-DT	0.864 ± 0.000	0.97 ± 0.000	0.028 ± 0.000	NA

Table 3. UNSW-NB15 dataset results depending on the number of features, and features sets

Model	Method	Number of features	Features set
The proposed model without LS	PHFS-IWD	3	[4, 27, 8]
The proposed model with TS	PHFS-IWDTS	3	[6, 4, 10]
The proposed model with NLSA	PHFS-IWDNLSA	4	[29, 11, 9, 30]
The proposed model with HC	PHFS-IWDHC	4	[4, 10, 11, 6]
From [54]	GA-RF	9	[27, 3, 41, 35, 36, 10, 31, 2, 18]
From [55]	IGRF-RFE	23	[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 12, 14, 15, 16, 18, 23, 24, 25, 26, 27, 31, 35]
From [10]	PIO	14	[3, 8, 9, 11, 12, 23, 26, 27, 28, 31, 38, 39, 40, 43]
From [56]	Rule-Based	13	[5, 8, 9, 10, 13, 14, 32, 41, 42, 43,45, 46,47]
From [57]	Wrapper-Based-DT	19	[3, 4, 6, 8, 9, 12, 15, 16, 21, 22, 30, 31, 32, 35, 36, 37, 40, 41, 42]

The proposed PHFS-IWDNLSA achieved the highest accuracy of 0.925, F-score of 0.871, and TPR of 0.988 against all other algorithms. In addition, the results show that PHFS-IWDNLSA provides better results represented by accuracy, TPR, and F-score values than the other proposed algorithms. All proposed methods achieve a competitive performance with the least set of features, not exceeding 4 features, as shown in Table 3. PHFS-IWDNLSA, and PHFS-IWDHC provide better FPR than all other methods examined in this subsection.

5.2. NLS-KDD results

Table 4 demonstrates that PHFS-IWDNLSA outperforms all other algorithms based on accuracy, TPR, FPR, and F-score. It provides the second best value in terms of F-score after PIO with a fewer number of features as shown in Table 5. The proposed method, without any local search algorithms (PHFS-IWD), achieved less accuracy and F-score values, compared to the proposed methods with local search algorithms (PHFS-IWDTS, PHFS-IWDNLSA, and PHFS-IWDHC).

Table 4. NLS-KDD dataset results using the DT classifier depending on Accuracy, TPR, FPR, and F-score metrics

Model	Method	Accuracy±STDV	TPR ±STDV	FPR ±STDV	F-score±STDV
The proposed model without LS	PHFS-IWD	$0.811 \pm 3.4\times10^{-16}$	$0.943 \pm 4.7\times10^{-5}$	0.29 ± 0.0000	$0.811 \pm 8.5\times10^{-6}$
The proposed model with TS	PHFS-IWDTS	$0.\ 867 \pm 2.8\times10^{-5}$	$0.925 \pm 6.2\times10^{-5}$	0.19 ± 0.0000	$0.855 \pm 3.4\times10^{-5}$
The proposed model with NLSA	PHFS-IWDNLSA	0.871 ± 0.0006	0.926 ± 0.0001	0.18 ± 0.0000	0.859 ± 0.0006
The proposed model with HC	PHFS-IWDHC	$0.831 \pm 8.4\times10^{-5}$	$0.959 \pm 7.8\times10^{-6}$	0.27 ± 0.0000	$0.829 \pm 6.9\times10^{-5}$
From [58]	RL-NIDS	0.813 ± 0.000	0.9643 ± 0.000	0.248 ± 0.000	NA
From [59]	MFFSEM	0.843 ± 0.000	0.964 ± 0.000	0.248 ± 0.000	NA
From [10]	PIO	0.869 ± 0.006	0.817 ± 0.012	0.064 ± 0.0008	0.864 ± 0.006
From [60]	Two-Stage Classifier Ensemble	0.858 ± 0.000	0.88 ± 0.012	NA	NA

In addition, it is clear that all proposed methods outperform other methods from the literature in terms of TPR. The proposed method provides better results using NLSA, in comparison with TS. However, using TS achieves better results than using HC based on both accuracy and F-score. Overall, the proposed methods provide competitive results depending on accuracy, FPR, F-score, and TPR, compared to some of the most recent methods from the literature using the least number of features, as shown in Table 5.

Table 5. NLS-KDD dataset results using the DT classifier depending on the number of features, and features sets

Model	Method	Number of features	Features set
The proposed model without LS	PHFS-IWD	4	[28, 0, 1, 5]
The proposed model with TS	PHFS-IWDTS	3	[2, 6, 27]
The proposed model with NLSA	PHFS-IWDNLSA	4	[2, 6, 10, 27]
The proposed model with HC	PHFS-IWDHC	4	[27, 10, 5, 1]
From [10]	PIO	18	[1, 3, 4, 5, 6, 8, 10, 11, 12, 13, 14, 15, 17, 18, 27, 32, 36, 39, 41]
From [60]	PSO	37	[2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 17,18, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 31, 32, 33, 34, 35, 36,37, 38, 39, 40, 41]
From [61]	IG	8	[5, 3, 6, 4, 30, 29, 33, 34]

5.3. KDDCUP^{99} results

PHFS-IWDNLSA is considered the best-performing algorithm in terms of accuracy, F-score, and FPR, with scores that equal 95.3%, 96.9 %, and 0.0026, respectively, using only 7 out of 41 features as shown in Table 6 and Table 7 below. Other proposed methods (PHFS-IWDTS, PHFS-IWDHC, and PHFS-IWD) follow PHFS-IWDNLSA and PIO in terms of accuracy. On the contrary, these methods use the least set of features, not exceeding 5 out of 41 features.

It is obvious that all proposed methods provide better performance in terms of FPR, and TPR compared with all other state-of-the-art examined methods, using the least set of features as illustrated in Table7. PHFS-IWDNLSA, PHFS-IWDTS and PHFS-IWDHC outperform the proposed method without any local search algorithms (PHFS-IWD) in terms of FPR, TPR, and F-score.

Table 6. KDDCUP99 dataset results using the DT classifier depending on Accuracy, TPR, FPR, and F-score metrics

Model	Method	Accuracy±STDV	TPR ±STDV	FPR ±STDV	F-score ±STDV
The proposed model without LS	PHFS-IWD	0.934 ± 0.0005	0.998 ± 0.0001	0.075 ± 4.0×10^{-5}	0.953 ± 0.0004
The proposed model with TS	PHFS-IWDTS	0. 942 ± 5.6×10^{-16}	0.984 ± 0.0002	0.016 ± 0.002	0.961 ± 0.0004
The proposed model with NLSA	PHFS-IWDNLSA	0.953 ± 0.002	0.997 ± 0.008	0.0026 ± 0.005	0.969 ± 0.008
The proposed model with HC	PHFS-IWDHC	0.941 ± 2.3×10^{-16}	0.993 ± 0.005	0.007 ± 0.002	0.962 ± 0.000
From [59]	MFFSEM	0.925 ± 0.000	0.925 ± 0.000	0.02 ± 0.000	NA
From [62]	HAM	0.872 ± 0.000	0.909 ± 0.000	0.17 ± 0.000	NA
From [10]	PIO	0.947 ± 0.001	0.974 ± 0.001	0.097 ± 0.001	NA
From [63]	FGLCC	0.926 ± 0.00	0.913 ± 0.00	0.022 ± 0.000	NA

Table 7. KDDCUP^{99} dataset results using the DT classifier depending on the number of features, and features sets

Model	Method	Number of features	Features set
The proposed model without LS	PHFS-IWD	4	[2, 3, 4, 35]
The proposed model with TS	PHFS-IWDTS	4	[2, 13, 1, 10]
The proposed model with NLSA	PHFS-IWDNLSA	7	[3, 4, 6, 13, 23, 29, 34]
The proposed model with HC	PHFS-IWDHC	5	[2, 3, 6, 23, 36]
From [63]	FGLCC	16	[4, 6, 10, 13, 22, 23, 24, 27, 29, 30, 32, 35, 36, 39, 40, 41]
From [10]	PIO	10	[3, 4, 6, 11, 13, 18, 23, 36, 37, 39]
From [63]	Cuttlefish	10	[4, 10, 13, 22, 23, 24, 29, 35,36, 41]

6. Conclusion

This paper proposes a new IDS based on a hybrid feature selection method. This method consists of two stages. In the first stage, in order to select the most relevant features, an ensemble filter is applied, and in the second stage, an improved IWD algorithm is deployed as a wrapper. Three common IDS benchmark datasets are used with the DT classifier to evaluate the proposed hybrid FS methods versus some of the most recent works in the literature in terms of accuracy, F-score, TPR, FPR, and the number of the selected features. From results, it is noted that the proposed methods has achieved superior performance in terms of accuracy, TPR, FPR, and F-score metrics for all three datasets versus some recent state-of-the-art methods. In addition, these methods reduce the number of the selected features in all datasets. PHFS-IWDNLSA outperforms the rest of the proposed methods. It reduces the number of features to 4 out of 41, 4 out of 49, and 7 out of 41 in UNSW-NB15, NLS-KDD, and KDDCUP^{99} datasets, respectively.

***Declaration of Competing Interest*:** The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

***Data Availability Statement*:** The KDDCUP99 data are available in UCI repository [**https://archive.ics.uci.edu/ml/datasets/kdd+cup+1999+data**], while the NSL-KDD dataset are available from Canadian Institute for Cybersecurity [**https://www.unb.ca/cic/datasets/nsl.html**], and finally the UNSW-NB15 dataset are available in Kaggle repository [**https://www.kaggle.com/datasets/mrwellsdavid/unsw-nb15**].

References

1. Alazzam, H., O. AbuAlghanam, Q. M. Al-zoubi, A. Alsmady, E. Alhenawi. A New Network Digital Forensics Approach for Internet of Things Environment Based on Binary Owl Optimizer. – Cybernetics and Information Technologies, Vol. **22**, 2022, No 3, pp. 146-160.
2. AbuAlghanam, O., M. Qatawneh, W. Almobaideen, M. Saadeh. A New Hierarchical Architecture and Protocol for Key Distribution in the Context of Iot-Based Smart Cities. – Journal of Information Security and Applications, Vol. **67**, pp. 103-173.
3. Abualghanam, O., M. Qatawneh, W. Almobaideen. A Survey of Key Distribution in the Context of Internet of Things. – Journal of Theoretical and Applied Information Technology, Vol. **97**, No 22, pp. 3217-3241.
4. AbuAlghanam, O., L. Albdour, L., O. Adwan. Multimodal Biometric Fusion Online Handwritten Signature Verification Using Neural Network and Support Vector Machine. – Transactions, Vol. **7**, No 8.
5. Alazzam, H., A. Sharieh, K. E. Sabri. A Lightweight Intelligent Network Intrusion Detection System Using Ocsvm and Pigeon Inspired Optimizer. – Applied Intelligence, pp. 1-18.
6. Scarfone, K., P. Mell. Guide to Intrusion Detection and Prevention Systems (IDPS). NIST. Special Publication. Vol. **800**. 2007. 94 p.
7. Torabi, M., N. I. Udzir, M. T. Abdullah, R. A. Yaakob. Review on Feature Selection and Ensemble Techniques for Intrusion Detection System. – Network, Vol. **1**, No 2.
8. Raghunandan, K. R., A. Ganesh, S. Surendra, K. Bhavya. Key Generation Using Generalized Pell's Equation in Public Key Cryptography Based on the Prime Fake Modulus Principle to Image Encryption and Its Security Analysis. – Cybernetics and Information Technologies, Vol. **20**, 2020, No 3, pp. 86-101.
9. Ejaz, S., U. Noor, Z. Rashid. Visualizing Interesting Patterns in Cyber Threat Intelligence Using Machine Learning Techniques. – Cybernetics and Information Technologies, Vol. **22**, 2022, No 2, pp. 96-113.
10. Alazzam, H., A. Sharieh, K. E. Sabri. A Feature Selection Algorithm for Intrusion Detection System Based on Pigeon Inspired Optimizer. – Expert Systems with Applications, Vol. **148**, pp. 113249.
11. AbuAlghanam, O., O. Adwan, M. A. Al Shariah, M. Qatawneh. Enhancing the Speed of the Learning Vector Quantization (LVQ) Algorithm by Adding Partial Distance Computation. – Cybernetics and Information Technologies, Vol. **22**, 2022, No 2, pp. 36-49.

12. Dietterich, T. G. Ensemble Methods in Machine Learning. – In: Proc. of International Workshop on Multiple Classifier Systems, Springer, pp. 1-15.
13. Alhenawi, E., R. Al-Sayyed, A. Hudaib, S. Mirjalili. Feature Selection Methods on Gene Expression Microarray Data for Cancer Classification: A Systematic Review. – Computers in Biology and Medicine, Vol. **140**, pp. 105051.
14. Kant, S., D. Agarwal, P. K. Shukla. A Survey on Fuzzy Systems Optimization Using Evolutionary Algorithms and Swarm Intelligence. – Computer Vision and Robotics, pp. 421-444.
15. Dharini, S., S. Jain. A Novel Metaheuristic Optimal Feature Selection Framework for Object Detection with Improved Detection Accuracy Based on Pulse-Coupled Neural Network. – Soft Computing, pp. 1-13.
16. Huda, R. K., H. Banka. Efficient Feature Selection Methods Using PSO with Fuzzy Rough Set as Fitness Function. – Soft Computing, Vol. **26**, No 5, pp. 2501-2521.
17. Wang, H., C. He, Z. Li. A New Ensemble Feature Selection Approach Based on Genetic Algorithm. – Soft Computing, Vol. **24**, No 20, pp. 15811-15820.
18. Tan, F., X. Fu, Y. Zhang, A. G. Bourgeois. A Genetic Algorithm-Based Method for Feature Subset Selection. – Soft Computing, Vol. **12**, No 2, pp. 111-120.
19. Gharaee, H., H. Hosseinvand. A New Feature Selection IDS Based on Genetic Algorithm and SVM. – In: Proc. of 8th International Symposium on Telecommunications (IST'16), IEEE, 2016, pp. 139-44.
20. Kumar, G. R., G. Ramachandra, K. Nagamani. An Efficient Feature Selection System to Integrating SVM with Genetic Algorithm for Large Medical Datasets. – International Journal, Vol. **4**, No 2, pp. 272-277.
21. Ghatasheh, N., I. Altaharwa, K. Aldebei. Modified Genetic Algorithm for Feature Selection and Hyper Parameter Optimization: Case of XGBoost in Spam Prediction – IEEE Access.
22. Deng, X., M. Li, S. Deng, L. Wang. Hybrid Gene Selection Approach Using Xgboost and Multi-Objective Genetic Algorithm for Cancer Classification. – Medical & Biological Engineering & Computing, Vol. **60**, pp. 663-681.
23. Sahu, B., D. Mishra. A Novel Feature Selection Algorithm Using Particle Swarm Optimization for Cancer Microarray Data. – Procedia Engineering, Vol. **38**, 2012, pp. 27-31.
24. Vijayashree, J., H. P. Sultana. A Machine Learning Framework for Feature Selection in Heart Disease Classification Using Improved Particle Swarm Optimization with Support Vector Machine Classifier. – Programming and Computer Software, Vol. **44**, 2018, No 6, pp. 388-397.
25. Paniri, M., M. B. Dowlatshahi, H. Nezamabadi-pour. MLACO: A Multi-Label Feature Selection Algorithm Based on Ant Colony Optimization. – Knowledge-Based Systems, Vol. **192**, 2020, pp. 105-285.
https://www.sciencedirect.com/science/article/pii/ S0950705119305805
26. Jayaprakash, A., C. KeziSelvaVijila. Feature Selection Using Ant Colony Optimization (ACO) and Road Sign Detection and Recognition (RSDR) System. – Cognitive Systems Research, Vol. **58**, 2019, pp. 123-133.
27. Alijla, B. O., C. P. Lim, L. P. Wong, A. T. Khader, M. A. Al-Betar. An Ensemble of Intelligent Water Drop Algorithm for Feature Selection Optimization Problem. – Applied Soft Computing, Vol. **65**, pp. 531-541.
28. Acharya, N., S. Singh. An IWD-Based Feature Selection Method for Intrusion Detection System. – Soft Computing, Vol. **22**, pp. 4407-4416.
29. Aslahi-Shahri, B., R. Rahmani, M. Chizari, A. Maralani, M. Eslami, M. J. Golkar, A. Ebrahimi. A Hybrid Method Consisting of GA and SVM for Intrusion Detection System. – Neural Computing and Applications, Vol. **27**, pp. 1669-1676.
30. Kunhare, N., R. Tiwari, J. Dhar. Particle Swarm Optimization and Feature Selection for Intrusion Detection System. – Sadhan, Vol. **45**, No 1, pp. 1-14.
31. Al Tawil, A., K. E. Sabri. A Feature Selection Algorithm for Intrusion Detection System Based on Moth Flame Optimization. – In: Proc. of International IEEE Conference on Information Technology (ICIT'21), 2021, pp. 377-381.

32. Kira, K., L. A. Rendell. The Feature Selection Problem: Traditional Methods and a New Algorithm. – AAAI, Vol. **2**, pp. 129-134.

33. Zhang, Y., X. Ren, J. Zhang. Intrusion Detection Method Based on Information Gain and ReliefF Feature Selection. – In: Proc. of International Joint Conference on Neural Networks (IJCNN'19), 2019, pp. 1-5.

34. Shreem, S. S., S. Abdullah, M. Z. A. Nazri, M. Alzaqebah. Hybridizing ReliefF, MRMR Filters and GA Wrapper Approaches for Gene Selection. – J Theor Appl Inf Technol, Vol. **46**, No 2, pp. 1034-1039.

35. Gu, Q., Z. Li, J. Han. Generalized Fisher Score for Feature Selection. – arXiv preprint arXiv:12023725.

36. He, X., D. Cai, P. Niyogi. Laplacian Score for Feature Selection. – Advances in Neural Information Processing Systems, Vol. **18**.

37. Glover, F., M. Laguna. Tabu Search – Handbook of Combinatorial Optimization. – Springer, 1998. pp. 2093-2229.

38. Ahmadian, A., A. Elkamel, A. Mazouz. An Improved Hybrid Particle Swarm Optimization and Tabu Search Algorithm for Expansion Planning of Large Dimension Electric Distribution Network. – Energies, Vol. **12**, No 16, pp. 30-52.

39. Alazzam, H., E. Alhenawi, R. Al-Sayyed. A Hybrid Job Scheduling Algorithm Based on Tabu and Harmony Search Algorithms. – The Journal of Supercomputing, Vol. **75**, pp. 7994-8011.

40. Tubishat, M., N. Idris, L. Shuib, M. A. Abushariah, S. Mirjalili. Improved Salp Swarm Algorithm Based on Opposition Based Learning and Novel Local Search Algorithm for Feature Selection. – Expert Systems with Applications, Vol. **145**, pp. 113-122.

41. Shehab, M., A. T. Khader, M. A. Al-Betar, L. M. Abualigah. Hybridizing Cuckoo Search Algorithm with Hill Climbing for Numerical Optimization Problems. – In: Proc. of 8th International Conference on Information Technology (ICIT'17), 2017, pp. 36-43.

42. Shah-Hosseini, H. The Intelligent Water Drops Algorithm: A Nature Inspired Swarm-Based Optimization Algorithm. – International Journal of Bio-Inspired Computation, Vol. **1**, No 1-2, pp. 71-79.

43. Lippmann, R. P., I. Graf, D. Wyschogrod, S. E. Webster, D. J. Weber, S. Gorton. The 1998 DARPA/AFRL Off-Line Intrusion Detection Evaluation. – In: Proc. of First International Workshop on Recent Advances in Intrusion Detection (RAID).

44. Ahmed, M., A. Mahmood, N. Abdun, J. Hu. A Survey of Network Anomaly Detection Techniques – Journal of Network and Computer Applications, Vol. 60, pp. 19-31.

45. Moustafa, N., J. Slay. UNSW-NB15: A Comprehensive Data Set for Network Intrusion Detection Systems (UNSW-NB15 Network Data Set). – In: Proc. of Military Communications and Information Systems Conference (MilCIS'15), 2015, pp. 1-6.

46. Tavallaee, M., E. Bagheri, W. Lu, 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, 2009, pp. 1-6.

47. Ingre, B., A. Yadav. Performance Analysis of NSL-KDD Dataset Using ANN. – In: Proc. of International Conference on Signal Processing and Communication Engineering Systems, 2015, pp. 92-96.

48. Ribaric, S., I. Fratric. Experimental Evaluation of Matching-Score Normalization Techniques on Different Multimodal Biometric Systems. – In: Proc. of MELECON 2006-2006 IEEE Mediterranean Electrotechnical Conference, pp. 498-501.

49. Sahu, S. K., S. Sarangi, S. K. Jena. A Detail Analysis on Intrusion Detection Datasets. – In: Proc. of IEEE International Advance Computing Conference (IACC'14), 2014, pp. 1348-1353.

50. Song, Y. Y., L. Ying. Decision Tree Methods: Applications for Classification and Prediction. – Shanghai Archives of Psychiatry, Vol. **27**, No 2, pp. 130.

51. Rawashdeh, H., S. Awawdeh, F. Shannag, E. Henawi, H. Faris, N. Obeid. Intelligent System Based on Data Mining Techniques for Prediction of Preterm Birth for Women with Cervical Cerclage. – Computational Biology and Chemistry, Vol. **85**, pp. 107233.

52. Shukla, A. K, P. Singh, M. Vardhan. A Hybrid Gene Selection Method for Microarray Recognition. – Biocybernetics and Biomedical Engineering, Vol. **38**, No 4, pp. 975-91.

53. Powers, D. M. Evaluation: from Precision, Recall and F-Measure to ROC, Informedness, Markedness and Correlation. – arXiv preprint arXiv:201016061.

54. Liu, Z., Y. Shi. A Hybrid IDS Using GA-Based Feature Selection Method and Random Forest. – Int. J. Mach. Learn. Comput., Vol. **12**, No 2, pp. 43-50.

55. Yin, Y., J. Jang-Jaccard, W. Xu, A. Singh, J. Zhu, F. Sabrina. IGRF-RFE: A Hybrid Feature Selection Method for MLP-Based Network Intrusion Detection on UNSW-NB15 Dataset. – arXiv preprint arXiv:220316365.

56. Kumar, V., D. Sinha, A. K. Das, S. C. Pandey, R. T. Goswami. An Integrated Rule Based Intrusion Detection System: Analysis on UNSW-NB15 Data Set and the Real Time Online Dataset. – Cluster Computing, Vol. **23**, No 2, pp. 1397-418.

57. Umar, M. A., C. Zhanfang, Y. Liu. Network Intrusion Detection Using Wrapper-Based Decision Tree for Feature Selection. – In: Proc. of International Conference on Internet Computing for Science and Engineering, 2020, pp. 5-13.

58. Wang, W., S. Jian, Y. Tan, Q. Wu, C. Huang. Representation Learning-Based Network Intrusion Detection System by Capturing Explicit and Implicit Feature Interactions. – Computers & Security, Vol. **112**, pp. 102537.

59. Zhang, H., J. L. Li, X. M. Liu, C. Dong. Multi-Dimensional Feature Fusion and Stacking Ensemble Mechanism for Network Intrusion Detection. – Future Generation Computer Systems, Vol. **122**, pp. 130-143.

60. Tama, B. A., M. Comuzzi, K. H. Rhee. TSE-IDS: A Two-Stage Classifier Ensemble for Intelligent Anomaly-Based Intrusion Detection System. – IEEE Access, Vol. **7**, pp. 94497-94507.

61. Aljawarneh, S., M. Aldwairi, M. B. Yassein. Anomaly-Based Intrusion Detection System through Feature Selection Analysis and Building Hybrid Efficient Model. – Journal of Computational Science, Vol. **25**, pp. 152-160.

62. Ghanem, W. A., A. Jantan. Training a Neural Network for Cyberattack Classification Applications Using Hybridization of an Artificial Bee Colony and Monarch Butterfly Optimization. – Neural Processing Letters, Vol. **51**, No 1, pp. 905-946.

63. Mohammadi, S., H. Mirvaziri, M. Ghazizadeh-Ahsaee, H. Karimipour. Cyber Intrusion Detection by Combined Feature Selection Algorithm. – Journal of Information Security and Applications, Vol. **44**, pp. 80-88.

Received: 03.10.2022; Second Version: 10.10.2022; Accepted: 18.10.2022 (fast track)