Karbala International Journal of Modern Science

Volume 7 | Issue 4 Article 15

Proposed Hybrid CorrelationFeatureSelectionForestPanalizedAttribute Approach to advance IDSs

Doaa Nteesha Mhawi
Computer Science Department, University of Technology, Technical Institute for Administration, Middle Technical University, Baghdad, Iraq, dododuaaenteesha@mtu.edu.iq

Prof. Soukaena H. Hashem
Computer Science Department, University of Technology, Baghdad, Iraq

Recommended Citation

Mhawi, Doaa Nteesha and Hashem, Prof. Soukaena H. (2021) "Proposed Hybrid CorrelationFeatureSelectionForestPanalizedAttribute Approach to advance IDSs," *Karbala International Journal of Modern Science*: Vol. 7 : Iss. 4 , Article 15.
Available at: https://doi.org/10.33640/2405-609X.3166

Proposed Hybrid CorrelationFeatureSelectionForestPanalizedAttribute Approach to advance IDSs

Abstract

NetworkIntrusionDetectionSystem(NIDS), widely used network infrastructure. Although many datamining has been used to increase the effectiveness of IDSs, current ID still struggle to perform well. therfore; proposed a new NIDS focused on feature_selection. The proposed CorrelationFeatureSelection_ForestPanalizedAttributes(CFS_FPA) used for dimensionality_reduction and selects the optimal_subset. based on two steps: first check each feature with a target(class) and choose only features that most effective by applying CFS filter using a statistical_method, then applied FPA to select only features will enhance ID and reduce_dimensionality. proposal tested with the NSLKDD experimental results of accuracy 0.997% and 0.004 FAR, wherein UNSWNB15_dataset accuracy and FAR are 0.995%, 0.008 consequently.

Keywords

CorrelationFeatureSelection CFS, EnsembleMethod, ForestPanalizedAttribute FPA, CyberSecurity CS, IntrusionDetectionSystems IDSs

Creative Commons License

1. Introduction

Detecting zero-day incursions is a difficult task that may risk a company's survival. Massive amounts of new vulnerabilities are discovered daily, and the consequences of these invasions are becoming increasingly grave [1,2]. Computer attacks are becoming more complex, thus posing difficulties such as incorrect detection of an intrusion [3]. Intrusion detection systems (IDSs) provide warnings when they identify unusual behaviors or known threats. Any action that causes harm to an information system can be classified as an intrusion [4]. IDSs detect malicious activity in computer systems that use software or hardware. IDSs are used to monitor a computer system for unusual behavior that a regular packet filter might miss. Given that IDSs monitor network packets for hazardous activity signals, high cyber resiliencies against damaging activities and unauthorized access to a computer system are essential. IDSs employ two ways to detect intrusions: Signature Intrusion Detection Systems (SIDS) and Anomaly Intrusion Detection Systems (AIDS).

SIDS, also known as Knowledge Detection or Misuse Detection, is a system that generates a signature identification for known malware to identify it in the future [5]. The trace can be marked as malware if the same signature is found again. SIDS typically has high detection accuracy, particularly for previously detected intrusions. As a result of SIDS's success in updating the signature database, three issues arise.

First, the polymorphic features of malware make it simpler to fool signature-based systems. As this technique in the IDS database does not correspond to any signature, the similarity test fails, enabling an attacker to access the computer system. Second, the larger the signature database, the longer it takes to evaluate and interpret all of information. Finally, given that the signature is not saved in the database, SIDS has trouble detecting zero-day malware [6−8].

The limitations in SIDS have been rectified by using AIDS techniques, which are currently being used to identify malicious machine assaults. This method is based on the assumption that a harmful activity has a different profile than typical user behavior [9−12]. AIDS generates a statistical model of usual user behavior, and any deviation from this model is recognized. The AIDS design definition profiles and represents the typical and predicted standard behavior profile by tracking behaviors and categorizing abnormal events according to how much they deviated from the norm. AIDS analyzes data such as how many emails a user receives, how many unsuccessful login attempts a user has made, and how much CPU a host consumes in a specific interval to find trends. Anomaly detection techniques offer a high degree of generalizability and the potential to discover new threats, but they may have significant false alarm rates owing to the shifting cyber-attack climate. Considering that they vary from standard practices, alien user preferences are known as intrusions. The planning phase and the research phase are the two stages of AIDS. The normal profile is trained using data that show normal behavior during the training process, and the model is then assessed using data that were not used during the training phase. Depending on the methods used to learn about AIDS, it may be divided into many subcategories, such as, mathematics, knowledge-based, and machine learning [13−15].

The ability to detect zero-day attacks is a key advantage of AIDS, as it does not rely on signature databases to do so. When the examined conduct deviates from the standard, AIDS sends out a warning signal [16]. AIDS also has several advantages. First, it can detect internal malicious activity. When an attacker begins transacting on a compromised account, which may be mistaken for regular user behavior, an alarm is triggered. Second, given that the framework is built from personalized profiles, figuring out what a regular user does without triggering alarms is exceedingly difficult for a cyber criminal [4]. Traditional IDSs have many flaws, including the inability to discriminate between new vulnerabilities, updates that were needed, and produced high FAR and low accuracy. AIDS has flaws and a high rate of false alarms [17−19]. To overcome these shortcomings, a novel IDS model integrating SIDS and AIDS is presented as a way to enhance accuracy and to lower FAR. SIDS was able to identify well-known incursions, whereas AIDS was able to identify novel ones.

Furthermore, various attack types and network traffic attributes offer Machine Learning with still another hurdle, that it expands the problem's search area and raises computational and temporal complexity [20]. Feature selection is a good solution for IDSs that identify highly important features and removes useless ones with minimal performance degradation [21,22]. Wrapper, filter, and embedded methods are the three major models for feature selection dependent feature selection or information gain ratio (IGR). The IGR is the ratio of

https://doi.org/10.33640/2405-609X.3166

information gain to intrinsic information in a standard filter method. Although it minimizes the bias against multi-valued characteristics and eliminates the disadvantage of knowledge acquisition, in other cases it may favor features with fewer values. Correlation-based feature selection, unlike the knowledge gain ratio, enhances the relevance of input and output features while decreasing duplication. This method selects one function at a time according to how closely it correlates with the outputs, allowing for greater attribute flexibility and tuple reduction [12,23–28]. In this paper, we provide a standard method for dimensionality reduction and redundancy removal, as well as a natural-inspired FS methodology for extracting a subset of the original features. FS via ensemble learning increases the IDS's stability and accuracy while demanding minimal computational and time resources.

On an extended test consisting of two datasets, the NSL-KDD and UNSW BN15 datasets, the idea is compared against existing approaches. The proposed method outperforms comparable algorithms by using measures of accuracy, F-measure, and DR while maintaining normal FAR levels, according to experimental data. The rest of the paper is organized as follows. In Second 2, similar works are reviewed. In Section 3, the proposed hybrid CFS-FPA in feature selection is defined in detail. In Section 4, the experiment results are presented. Section 5 provides a review and discussion of the previous articles.

2. Related works

Various IDSs for detecting anomalous activities are found in the literature. However, the majority of these IDSs create a large number of false positives and have low detection accuracy. Many hybrid IDSs have been proposed to mitigate the shortcomings of SIDS and AIDS.

- [3], offered a multi-class support vector machine-based intrusion detection algorithm on the basis of chi-square feature selection (SVM). A parameter-tweaking strategy is used to optimize the over-fitting constant “C” and the gamma parameter of the Radial Basis Function kernel. These are the only two parameters that allow the SVM model to function. The primary goal of this application is to develop a multi-class SVM, which has never been used for IDS before, to reduce training and testing time while improving individual network attack classification accuracy. Our suggested methodology improves detection rates while minimizing false alarm rates, according to the NSL-KDD dataset, which is a recent version of the KDDCup 1999 dataset. Using in time-critical circumstances, an evaluation of the computing time necessary for training and testing is also carried out.
- [9], Investigations NSLKDD dataset, a variant with the well KDD Cup 99 data set, were used to assess the suggested hybrid intrusion detection approach. The experimental findings show that the suggested technique outperforms traditional approaches in terms of detection rate for new and known assaults while having a low FAR. Furthermore, the suggested solution considerably decreases the training and testing processes' high time complexity. The anomaly detection model's training and testing times are only 50% and 60% of the time required for traditional models, respectively, in experiments.
- [21], This work uses many current feature selection approaches to develop a robust classifier that is computationally efficient and effective to minimize unnecessary features from the NSL-KDD data set. Info Gain, Correlation, Relief, and Symmetrical Uncertainty are four additional feature selection approaches that are combined with the C4.5 decision tree algorithm to form IDS. The experiments used the WEKA open-source data mining program, and the results show that C4.5, when employing the Info Gain feature selection process, had the highest accuracy of 99.68% with 17 features. Symmetrical Uncertainty with C4.5, while having only 11 characteristics, is as promising, with 99.64% accuracy. The findings outperform those of previous research in this subject.
- [22], In this study, a wrapper methodology based on a genetic algorithm as a search strategy and logistic regression as a learning algorithm was used to select the best subset of features for network intrusion detection systems.

Several techniques for improving the detection rate of intrusion detection systems have been developed. However, such techniques struggle to create and update the signature of new malware, as well as producing a high number of false alarms or low detection rates.

- [29], Four different data mining algorithms were offered as basic classifiers for such ensemble approaches: J48 (decision tree), JRip (rule induction), and iBK are all examples of Nave Bayes (nearest neighbor). According to our research, the prototype, which uses four base classifiers and three ensemble algorithms, correctly detects current invasions at a rate of over 99%. However, it fails to detect fresh incursions at a rate of more than 60%. Bagging, boosting, and stacking do not enhance accuracy considerably. Stacking is the only strategy that significantly reduced the false positive rate (46.84%); yet, it is the most time-consuming technique to deploy, making it inefficient in the intrusion detection sector.

- [30], "K-Means + C4.5" is a suggested anomaly detection mechanism for distinguishing between abnormal and normal occurrences in a computer system. By using Euclidean distance similarity, the K-Means clustering algorithm separates the training data into *k* groups. In terms of accuracy, the hybrid method outperforms the individual methodology, although it has a high proportion of false alarms.
- [31]**,** A C5 DT classification and one class SVM combine to form hybrid IDS (HIDS) (OC-SVM). HIDS combines SIDS with IDSs based on anomalies (AIDS). A C5.0 DT Classification has been utilized to build SIDS, whereas one-class SVM has been utilized to build AIDS. This technique finds known intrusions with excellent detection accuracy and low false alarm rates, together with zero-day threats. The proposed HIDS is examined utilizing the NSL-KDD and ADFA datasets for Network Safety Laboratory-Knowledge Discovery. Research demonstrates that HIDS has greater DR and a lower number of FAR than SIDS and AIDS.
- [32], In the suggested study, the Hidden Naïve Bayes model (HNB) might be applied to dimensionality, high-related functionality, and large network data stream volumes intrusion detection issues. HNB is a paradigm of data mining that relaxes the conditional autonomy of the Naïve Bayes procedure. Experimental results showed that, with the standard Naïve Bayes model, leading extended Bayes models and 1999 winners of the KDD Cup, the HNB model demonstrates higher global performance in precision, error ratio, and error cost. Our model fared better in predicting accuracy than other leading cutting-edge models such as SVM. The results also show that our approach enhances the accuracy of Denial of Service (DoS)detection considerably
- [33], This work aims to remove redundant instances that result in an unbiased learning algorithm (ii) by using a wrapper-based feature selection approach to determine the appropriate subset of characteristics (iii) to achieve greater detection accuracy. The IDS is designed using a wrapper-based function selection method that enhances the specificity and sensitivity of the IDS and uses an iterative approach for neural group decision-making to produce optimum features. A comprehensive experimental assessment has been carried out to find anomalous grid patterns, including the suggested methodology with a family of six decision tree classifiers, namely, Decision Stump, C4.5, the Naïve Bayes Tree, Random Forest, The Random Tree, and the Representative Tree model.
- The feature selection technique [34,35], The control plane, data plane, and energy plane make up the three-plane software-defined green 5G architecture for big data. Networking and processing equipment that The data plane, which is part of the energy plane, can be powered by both standard grid and renewable energy sources. The control plane monitors the state of a system and makes necessary adjustments to increase energy efficiency and service quality. Furthermore, the purpose of this FRS study is to eliminate duplicate system monitoring data and software-defined architecture overhead. We propose an AIFS for mining latent rules among characteristics to incorporate features in software-defined architecture. Our solutions appear to be more efficient in the green 5G system, according to simulation results.

3. Hybrid CFS-FPA

The purpose of FS is to select a subset of the original set's attributes that is adequately representative of the outcomes, because the subset's attributions are crucial to the prediction. Wrapper, filter, and embedded feature selection are the most common feature selection methods [36].

Wrapper techniques use classification results to assess and pick feature subsets, whereas filter techniques determine if a dataset's relevance and feature selection are based on statistics.

Embedding techniques are much less computational than wrapper techniques, because they interact with the selection of features. A regularized risk function is utilized in embedded techniques to enhance feature designation and predictor parameters [37]. Moreover, making adjustments to the categorization model to enhance outcomes is complex [38].

Redundant and useless features are found in today's intrusion detection databases [39]. They decrease the effectiveness of data mining algorithms and lead to unintelligible findings [40]. Therefore, the first stage in this research is to minimize the dimensionality of the dataset and choose the function subset [41]. In this research, a hybrid strategy is developed to increase feature selection efficiency and classification accuracy by combining correlation-based feature selection (CFS) and the bagging method of ensemble learning (RF). Before looking for the optimal solution in the given search space, this approach evaluates the validity and redundancy of a function subset. Fig. 1 depicts the proposed model.

3.1. Correlation Feature Selection (CFS)

CFS is a conventional filter method that selects features by using a heuristic (correlation-based) evaluation function [42]. This function favors subsets with characteristics that are highly connected to the class but not to one another. Although unimportant characteristics with low-class affiliation should be disregarded, repeating characteristics are selected because they have a strong association with at least one of the other traits. Acceptance is determined by how well a feature predicts classes in parts of the instance space where other characteristics have yet to predict them. The evaluation function for feature subsets [43] in CFS is shown in Eq. (1).

$$\mu_s = \frac{k\overline{r_{c}f}}{\sqrt{k+k(k-1)+rff}} \tag{1}$$

where r_{cf} is the average degree of correlation between attributes and Label of category r_{ff} denotes the interrelationship average of the characteristics. In acquired subsets, a larger r_{cf} or a smaller random forest (RF) provides a higher evaluation value. To decrease the sizes of training and testing systems, a collection with the highest value achieved over the whole project is used.

3.2. Random_Forest (RF)

Breimanis suggested Random_Forest in Ref. [44], which is a decision_tree methodology that works by building several DT. It classifies hundreds of variables input according to their relevance without removing any variables. RF is a collection of classification trees, and each of them assigns the responsibility of finding the most prevalent class in the input data a single vote. When RF is utilized instead of other machine learning approaches, less parameters are offered, SVM and ANN, for example. A set of individual tree-structured classifiers in RF can be defined as:

$$\{h(x,\theta_k),k=1,2\ldots\} \tag{2}$$

where h stands for RF classifier and k represents a group of identical vectors dispersed randomly.

At input variable x, for the most prestigious class, each tree has a vote. The proportions and style of the tree structure are influenced by its use. For RF's success, establishing every decision-making tree is essential.

Outliers and parameters have little effect on RF, which has a low computation cost. In addition, overfitting is less of a problem than with a single DT, and the trees do not have to be trimmed [45]. The variance of an average of Bagging random variables, with the volatility of 2, each has a 1/B2 volatility. Eq. (3) illustrates the average variance assuming the variables were simply identically distributed but necessarily independent and had a positive P-relationship fairway.

$$p\sigma^2+\frac{1-p}{B}\sigma^2 \tag{3}$$

3.3. Penalizing attributes in the forest (FPA)

Unlike prior strategies that only use a subset of nonclass features, FPA develops an accurate collection of decision-making bodies using the strength of all non-class functions accessible in a source of information [46]. Simultaneously, to assure individual accuracy and to encourage excellent diversity, aspects connected to weight, such as weight assignment techniques and weight gaining methods, are considered. FPA will alter the weights of the characteristics in the most recent tree randomly within a weight range (WR) was demonstrated in the 4th equation.

$$wR^{\lambda}=\begin{cases}\left[0.000,\dfrac{-1}{e^{\lambda}}\right],\lambda=1\\[2ex]\left[e^{\frac{-1}{\lambda-1}}+P,e^{\frac{-1}{\lambda}}\right],\ \lambda>1\end{cases} \tag{4}$$

The attribute's level is as follows:

The p variable is used to prevent overlapping of the WR at various levels.

For instance, the value of an attribute in the root node is 1. The value of an assessed attribute for a root node infant is 2. Moreover, FPA features a mechanism that progressively increases the weights of properties unchecked in future trees, thereby lowering the adverse effects of maintaining weights not found in the most recent tree. Assume that an attribute Ai is evaluated at Tj*1*tree Level p with height and weight *i*. Then, compute Ai's weight increment value I as follows: 5th. Equation.

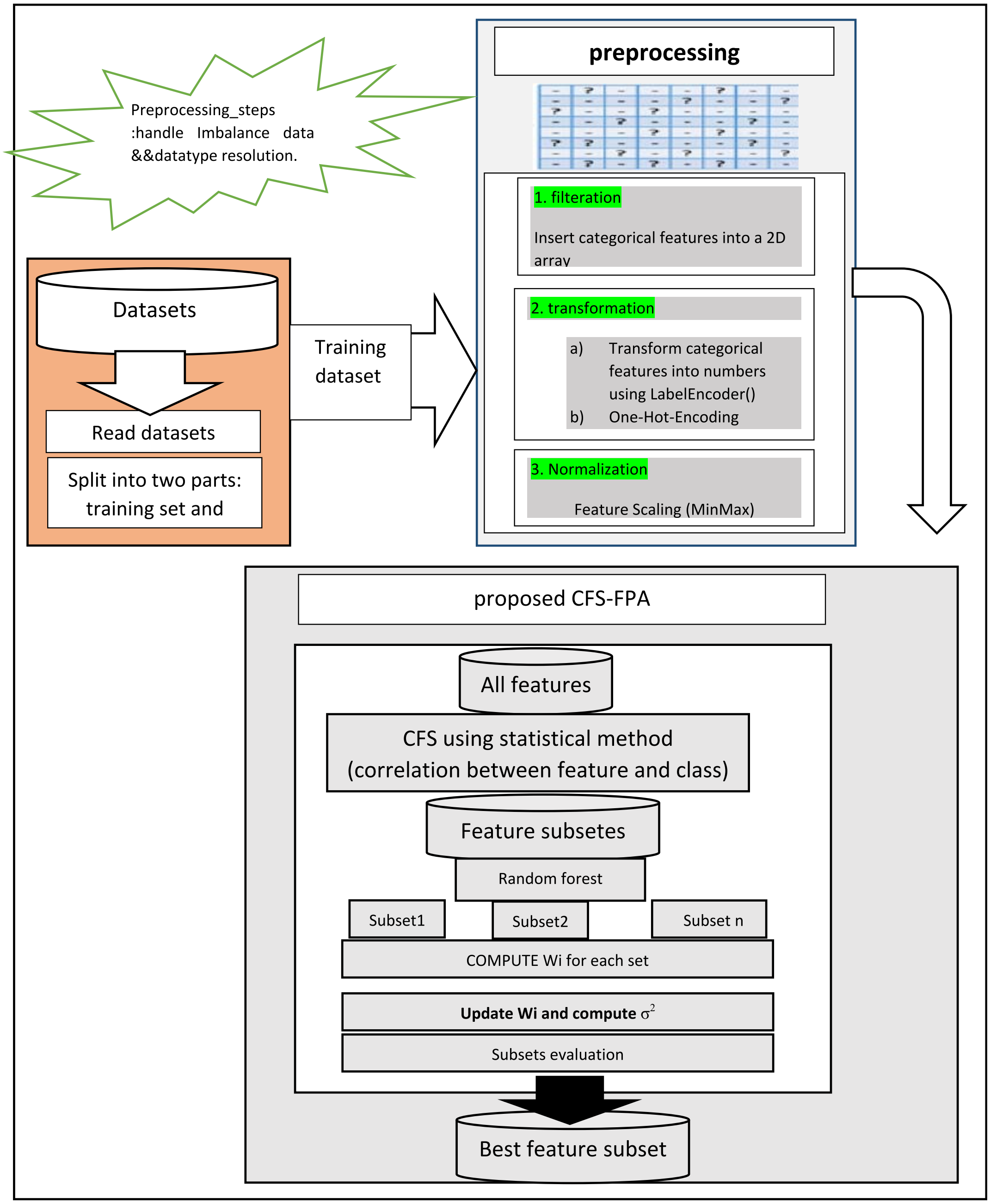


Fig. 1. Proposed hybrid CFS_FPA general structure.

$$\sigma_i = \frac{1.0 - \omega_i}{(n+1) - \lambda} \tag{5}$$

3.4. Hybrid CFS-FPA approach for feature selection

To establish the significance and correlation of a feature subset, Hybrid CFS-RF proposed a based feature selection methodology. To shape fitness functions and test the reduced feature subset's integrity, the CFS-RF technique uses a correlation-based feature technique. Given a feature subset, determination of the intercorrelation between average feature class and average feature correlation. (S = s1; s2; sk) S with k characteristics. Based on the outcome of the correlation-based evaluation function, CFS, as a traditional filter algorithm, may swiftly choose a subset of independently good features. Owing to functional duplication, this functionality subset may not be the optimum combination. Random forest is used to reduce dimensionality and remove redundant features, as well as to increase bagging variance reduction by reducing the connection between the trees without increasing variance too much. This is accomplished in the tree-growing method by selecting input variables at random by using Eq. (2). When growing a tree on a bootstrapped dataset, in particular. Algorithm 1, was demonstrated the main steps for this proposed method as following steps:

- Read datasets as the main input, and divide them into a training dataset and a testing dataset.
- Then the dataset is pre-processed, which is explained in detail in section 4.2.
- At the first step committee the initialization of each CFS using Eq. (1). This step is fast because this method of CFS is based on statistical operation can choose the most relevant feature to the target (class). In this process, redundancy is the problem, and FPA is used to reduce the redundancy and choose only features that are the most effective by using the weight to each feature through applied Eq. (5). And then generate sets of Random Forest RF (using 10 estimators [forest]) by applying Eq. (2).
- Begin all processes applied to the training dataset by computing CFS, then reduce the redundancy by applying FPA. Update the weight and compute the variance to choose the best set with less variance.
- Update the iteration.
- Output can choose the best subset of a feature that is useful in detecting the intrusion easily when applied to an intrusion detection system.

The two types of reasoning are rational and computational reasoning. First, the optimal representation in the space of hypothesis is impossible for a single classificatory. As a result, separate classifiers

Algorithm 1: Hybrid CFS-FPA

```
Input: datasets
Output: most effective features (X best)
    1. Split datasets into two parts: a training set and a testing set.
    2. Pre-processing the steps by using algorithm 2.
    3. Initialization: iteration t=1.
    4. Divided the datasets into classes
    5. for a training_set part do :
        a. At the first step committee the initialization of each CFS using Eq.$^1$.
        b. Generate new RF using Eq. $^2$.
        c. Initialize each feature weight value Wi by applying Eq. $^5$.
        d. Generate the number of RF as 10_forests (estimators), n_jobs=2,
           n_feature selection=30, and step=1.
        e. Xi selection from Xbest.
        f. While Xbest > Xi do :
            a. Use Eq. 3 to generate new Xnew.
            b. Compute Xnew CFS using Eq. 1.
            c. If Xi<Xnew and N(0,1)<Atti then
            d. Update Xi using Eq. 3.
            e. Compute Wi for each Xi generated from RF using Eq. 4.
            f. Compute $\sigma^2$ for each Xi using Eq. 5 generated from RF.
            g. Update Wi and $\sigma^2$
        g. end if
        h. end while
        i.  Xbest= Xnew
    6. Endfor
    7.  $t = t + 1$
    8. Output: the best subset selected (Xbest).
```

must be combined to improve prediction efficiency. Second, if the learning algorithm's training dataset is inadequate, then the result may be a faulty or erroneous hypothesis. In the latter instance, a single classifier may spend a large amount of time computing an acceptable hypothesis, thus increasing the likelihood of the approach failing.

4. Evaluation of the hybrid CFS-FPA approach according to results

The main aim of this paper is to build reliable IDSs with low false alarms and high accuracy. To achieve this objective, a hybrid approach called CFS-FPA, which combines CFS and FPA, is used to decide the best subset of the original features to remove unnecessary features and increase classification performance. Two datasets are used to test this proposed system: NSL_KDD and UNSW BN 2015. Experimental results were evaluated and executed by using Python language 8.3 Using a laptop with the following processor specifications: Intel(R) Core (TM), i7 10510UCPU@1.80 GHz 2.30 GHz, RAM 6.0 GB with 10G, and operating system type 64-bit operating system, x64-based processor. 4.1 Datasets for benchmarking a brief description. Finding an appropriate dataset is one of the difficulties that researchers face to evaluate IDSs. Obtaining a real-world dataset that accurately reflects network traffic without any anonymization or alteration is an issue that the cybersecurity research community has been grappling with for years [47]. In the event of a publication or transmission of the data, the data are severely anonymized or modified. Many of the important data items on which researchers rely would therefore be lost or become unreliable. As a result, some studies have decided to use the well-known data set of KDDCup’99 [48] or the NSL-KDD dataset [49]. That's a contemporary of her. Recently, great efforts have been made to construct datasets that represent genuine data. This research thus uses NSL-KDD and UNSW BN15 datasets to carry out experiments.

4.1. NSL_KDD dataset

In 2009, the NSL-KDD dataset [49] KDDCup’99, an updated version of the original dataset, was developed [48]. NSL-KDD had the KDDCup’99 benefits and problems. By deleting superfluous information, It addressed some of the flaws in the original dataset, reducing the number of cases while retaining sample diversity. The NSL-KDD dataset was designed to enhance prediction complexity, which is one of its most noticeable features. Different benchmark classifiers were used to categorize the records into five degrees of complexity, with each instance labeled with the number of correct predictions made [12]. The number of records picked for each difficulty level categorization is inversely proportional to the record percentage in the original KDDCup'99 dataset. The KDDTrain collection has 125.973 instances in this sample, with 58.630 attack traffic instances and 67.343 normal traffic instances. By comparison, the KDDTest set has a total of 22,544 instances with 11,850 extra events in the KDDTest subset. Table 1 shows the details of the dataset concisely.

4.1.1. UNSW NB15 dataset

We use the UNSW-NB15 attacks dataset [50] for our experimental procedures. Table 2 summarizes the 42 characteristics of the UNSW-NB15 in its clean format. Three of the 42 attributes are category (non-numeric), whereas the other 39 are numeric. The UNSW-NB15 is broken down into two primary datasetsUNSW-NB15-TRAIN (100%) is used to train various models, whereas UNSW-NB15-TEST (100%) is used to test the models that have been trained. We divided the UNSW NB15 TRAIN into two portions for our research: UNSW NB15 TRAIN-1 (75% of the overall training set) for training and UNSW NB15 VAL (25% of the whole training set) for validation before testing. The findings acquired throughout the training phase are checked against this second partition as a sanity check. Avoiding training a model on the evaluation or test set while employing this technique is crucial, as this practice might lead to a problem known as data leaking.

During the training process, data leakage happens when a model obtains the information it shouldn't, resulting in a bias in the final model. As a result, the model's performance when dealing with previously encountered data is poor [31]. The UNSW-NB15 contains network threats like backdoors, shellcode,

Table 1
NSL-KDD dataset.

Class	KDD_Train	KDD_Test
DoS	45,927	7,458
R2L	995	2745
PRB	11,656	2,421
U2R	52	200
NORMAL	67,343	9,711
Attack	58,630	12,833
Total	125,973	22,544

Table 2
UNSW_NB15 features.

Feature no.	Feature	Format	Feature no.	Feature	format
f1	du	float	f2	dtccpb	Int.
f3	proto	int.	f4	dwin	Int.
f5	service	categoric	f6	state	Flaot
f7	spkts	categoric	f8	dpkts	Float
f9	sbyte	float	f10	dbyte	categoric
f11	rate	int.	f12	sttle	Int.
f13	dttle	int.	f14	dload	Int.
f15	sloss	int.	f16	dloss	Int.
f17	sinpikt	int.	f18	dinpikt	Int.
f19	sjit	float	f20	djit	Int.
f21	swin	int.	f22	stcpb	Int.
f23	dwin	int.	f24	tcprtt	Int.
f25	smean	int.	f26	synack	Int.
f27	dmean	int.	f28	ackdat	Int.
f29	trans_depth	float	f30	resonse_body	Int.
f31	ct_src_port	float	f32	ct-srv-src	Int.
f33	ct_src_dest.	float	f34	ct-state	Int.
f35	Ct_item	int.	f36	ct_dst.	Int.
f37	Is_sm_ips	float	f38	Ct_src	Binary
f39	Ct_flow	int.	f40	Ct_src_sport	Int.
f41	Ct_src_des.item	int.	f42	Ct_ftp_login	Int.

reconnaissance, worms, fuzzes, DoS, generic, analysis, shellcode, and exploits. Within the data subsets, Table 3 illustrates the characteristics and distribution of values for each assault type.

4.2. Dataset preprocessing

The preparation of data is the longest, most important element of data mining. Typically realistic data are noisy, redundant, partial, and inconsistent, and are acquired from many sources [51]. Therefore, converting raw data into a format that can be used for analysis and information discovery is critical. Data filtration, data transformation, and data normalization are all part of the preprocessing phase in this study which uses Algorithm 2. Fig. 1 shows the distribution of datasets.

4.2.1. Data filtration

Before the filtration, datasets were divided into training and testing sets. The raw data will invariably contain aberrant and redundant occurrences as a result of platforms' heterogeneity, which may have a detrimental impact on classification accuracy. These records must be deleted from the datasets at the start of our trials to overcome this problem, and categorical characteristics must be inserted into a 2D array.

4.2.2. Data transforming normalization

The data sets have employed symbolic, continuous, and binary values. In NSL KDD datasets, for example, the "type of protocol" feature is provided. Symbolic values such as "tcp," "udp," and "icmp" are included. The conversion phase is critical because some

Table 3
UNSW-NB15 repartition instances.

Attack Type	UNSW-NB15	UNSW-NB15- TRAIN-1	UNSW NB15-VAL	UNSW-NB15-TEST
Norms	56,000	41,911	14,089	37,000
Generic	40,000	30,081	9919	18,871
Exploits	33,393	25,034	8359	11,132
Fuzzers	18,184	13,608	4576	6062
DoS	12,264	9237	3027	4089
Reconnaissance	10,491	7875	2616	3496
Analysis	2000	1477	523	677
Backdoor	1746	1330	416	583
5hellcode	1133	354	279	378
Worms	130	99	31	44

classifiers only take numerical inputs, and it has a considerable influence on IDS accuracy. To replace each value with an integer in this article, we use the following format: Transform category characteristics into numbers with:

a) LabelEncoder ()
b) One-Hot-Encoding

Furthermore, differing scaling among features may impair the classification outcome. As a consequence, normalization is a change that reduces the size of functions to a standard set of values. In our tests, we employed the MinMaxScale technique, which is a straightforward and quick way [52]. It can be described as:

$$\overline{x} = \frac{x - x_{m_in}}{x_{m_{ax}} - x_{m_{in}}} \tag{6}$$

and False Alarm Rate (FAR) [53]. Explains the statistical equations for the used evaluation criteria.

First, important features are determined by assessing the integrity of the reduced feature subset by using the suggested CFS-PFA-Ensemble approach in the feature selection step. Second, from the initial characteristics, candidates for the following step are identified.

Table 4 shows the numbers and names of selected attributes for the NSL-KDD and UNSW BN15 datasets. When used alone, CFS-FPA reduces dimensionality and eliminates superfluous attributes from a dataset. The recommended technique, in general, is based on selecting relevant features for all classes rather than a single class, which does not guarantee the performance of all types of assaults, particularly those with few cases in datasets. The known feature selection approaches may be used to identify intrusions because the classification conclusions for typical occurrences are largely consistent across varied datasets.

Algorithm 2: Preprocessing and Minimax Scaling

```
Input: Read $d1$ and $d2$ where d1 is NSL_KDD and $d2$ is UNSW BN15
Output: Normalize the dataset to $d1_{normalize}$ and $d2_{normalize}$.
Step 1: Data filtration
            • Remove anomalous and redundant instances from the datasets.
            • Split datasets into a training set (75%) and a testing set (25%).
Step 2: Transform the data
        for $i$ from 1 to $n$ do:
           if ($di$ nonnumeric input) then do:
     a) Transform categorical features into numbers using LabelEncoder ().
     b) One-Hot-Encoding
Step 3: Normalization
         Minmax Scaling computed by applying:
         $d_{n\ normalize} = d_n - (d_n)_{min} / (d_n)_{max} - (d_n)_{min}$
end if
end for
step 4: End
```

5. Discussion and results

An IDS's capacity to classify network traffic into the appropriate kind is used to evaluate its performance. To avoid the effects of data sampling while assessing IDSs, we compare the proposed model's output to no feature selection and various state-of-the-art approaches in terms of numerous detection metrics in this study, including Accuracy (ACC), Precision, Detection Rate(DR), F-Measure, Attack Detection Rate(ADR),

5.1. Model evaluation

Evaluation metrics used in IDS. Table 5 shows the confusion matrix for a two-class classifier that is widely used in an IDS. The instances in a predicted class are represented by each column of the matrix, whereas the instances in an actual class are represented by each row.

Typically, an IDS is evaluated by using the confusion matrix calculation as follows:

Table 4
Selected features name of datasets.

FS_name of NSL_KDD dataset (30 features)	FS _name of UNSW BN2015 dataset (30 features)
Count	“dst bytes”
Diff srv rate	“diff srv rate”
Dst bytes	“srv diff host rate”
Dst host count	“dst host count”
Dst host diff srv rate	“Dst host_srv_count”
Dst host error rate	“dst host_same_srv_rate”
Dst host same src port rate	“dst host_diff_srv_rate”
dst_host_same_srv_rate	“dst host_same_src_port_rate”
dst_host_serror_rate	“num access_files”
dst_host_srv_count	“num shells”
dst_host_srv_diff_host_rate	Num failed_logins
dst_host_srv_rerror_rate	“num root”
dst_host_srv_serror_rate	“su attempted”
flag_RSTR	“root shell”
flag_S0	“num compromised”
flag_SF	“serror rate”
“hot”	“count”, “srv_count
“num_compromised”	“is_guest_login”
Protocol_type_icmp	“is host_login”
Protocol_type_tcp	“num outbound_cmds”
rerror_rate	“same_srv_rate”
same_srv_rate	“srv rerror_rate”
serror_rate	“rerror_rate”
Serviceecr_i	“srv_serror_rate”
Service HTTP	“dst_host_srv_rerror_rate”
service_private	“dst_host_rerror_rate”
Src bytes	“dst_host_srv_serror_rate”
Srv. count	“dst_host_serror_rate”
Srv. rerror_rate	“dst_host_srv_diff_host_rate”

5.1.1. True positive rate (TPR)

The quantitative link between the number of attacks and the overall number of attacks is calculated. When all intrusions are identified accurately, the TPR is 1, which is rare for an IDS. Also known as the TPR, the rate of detection is defined In Eq. (7).

$$TPR = \frac{TP}{TP + FN} \tag{7}$$

5.1.2. False positive rate (FPR)

FPR is a quantitative measure of the connection between the number and a total number of normal instances identified as assaults. To measure the FPR, the following formula is presented in Eq. (8).

$$FPR = \frac{FP}{F_P + TN} \tag{8}$$

Table 5
Confusion matrix.

Actual class	Normal	intrusion
Normal	TN	FP
Intrusion	FN	TP

5.1.3. FNR (false negative rate)

The FNR indicates that the system for intrusion detection could not recognize and characterize the incursion as natural. The FNR is established in Eq. (9).

$$FNR = \frac{FN}{FN + T_P} \tag{9}$$

5.1.4. Accuracy

Accuracy is an IDS's accuracy obtained in categorizing normal and intrusion assaults, as assessed by Eq. (10).

$$Acc = \frac{TP + TN}{TP + TN + FP + FN} \tag{10}$$

5.1.5. Precision (P)

In Eq. (11), the ratio of total true positives (TP) to total true positives (TP) and false positives (FP) instances is the ratio of total true positives (TP) to total true positives (TP) and false positive (FP) instances.

$$\boldsymbol{P} = \frac{TP}{TP + FP} \tag{11}$$

5.1.6. Recall (R)

True positives (TP) are the percentage of total relevant outcomes accurately classified divided by the total number of true positive and false negative (FN) occurrences. Eq. (12) demonstrates this.

$$Recall = \frac{TP}{TP + FN} * 100\% \tag{12}$$

5.1.7. The F-measure (FM)

The FM is a measure of recall and precision. F-measure is used as an evaluation measurement when only one accuracy metric is required. Eq. (13) provides a good example.

$$F\_measure = \frac{2 * Recall * Precision}{Recall + Precision} \tag{13}$$

Table 6
Classification Performance for FS using NSL-KDD.

Classifier	Accuracy	Precision	F-measure	DR	FAR
a. Results of performance using original features (42 features)					
RF	0.949	0.944	0.947	0.949	0.021
FPA	0.945	0.942	0.944	0.945	0.028
Hybrid CFS_FPA	**0.994**	**0.993**	**0.993**	**0.992**	**0.016**
b. Results of performance using CFS-FPA (30 features)					
RF	0.949	0.944	0.947	0.949	0.021
FPA	0.945	0.942	0.944	0.945	0.028
Hybrid CFS_FPA	**0.997**	**0.998**	**0.997**	**0.998**	**0.004**

5.2. Comparison of the hybrid CSF_FPA ensemble with no feature-selection

We compare the suggested feature selection technique to the performance of the technique without feature selection in terms of identifying attacks from regular instances to assess the effectiveness of the proposed hybrid CFS FPA technique. As a result of the suggested hybrid CFS FPA method's gathering of key characteristics, the average values of several metrics, including as Acc, precision, DR, and F-Measure, have risen significantly. Table 6 summarizes the output of the basic and ensemble classifiers using the NSLKDD dataset. It is hypothesized that the ensemble classifier would not be strong enough in some measures if features were not selected. However, the hybrid CFS FPA Ensemble technique outperforms on both sets. NSL-KDD dataset On the basis of these results, our model is the most accurate and has the smallest FAR. The ensemble classifier that employs the original features performs worse in terms of accuracy and ADR than simple classifiers that employ the suggested feature selection approach, highlighting the relevance of the suggested feature selection approach. Furthermore, the proposed hybrid CFS FPA system ensemble model decreases the time overhead when applied to feature selection and ensemble model due to the dimensionality reduction of the subsets. Table 7 was demonstrated confusion_matrix performance categorization for a. NSL KDD and b. UNSW BN15.

5.3. Comparison with other FS methods

The benchmark datasets, as discussed in Section 4.1, depict a contemporary and complicated threat environment. Owing to the rising number of assault groups and their extremely uneven records, any machine learning technique confronts difficulty. We compare our proposed IDS model to various well-known feature selection approaches, such as IG (Information Gain) [54], IGR (Information Gain Ratio) [55], GA (Genetic Algorithm) [56], and PSO (Particle Swarm Optimization) [57], by conducting tests on two datasets. In this comparison research, we also employ standard measures such as Acc, F-Measure, DR, and FAR. To measure the effectiveness of the suggested IDS, a comparison was made in terms of the number of selected features and the time it took to choose them. Compared with several FS techniques based on the same recommended voting-based ensemble classifier, Fig. 2 illustrates our model's average performance. First, our proposed model outperforms previous feature selection-based techniques in every dataset, as demonstrated in Fig. 2(a). Similarly, our proposed model outperforms existing feature selection techniques in terms of F-Measure on all datasets, as shown in Fig. 2(b), by extracting more particular feature subsets. Our proposed model, as shown in Fig. 2(c), has a high attack detection rate. Furthermore, our proposed CFS-FPA-based model yields the lowest FAR values of 0.004% and 0.008%, respectively, on the

Table 7
Classification Performance for FS using UNSW_BN15.

Classifier	Accuracy	Precision	F-measure	DR	FAR
a. Results of performance using original features (49 features)					
RF	0.979	0.982	0.996	0.989	0.004
FPA	0.966	0.982	0.981	0.981	0.019
Hybrid CFS_FPA	**0.982**	**0.982**	**0.999**	**0.990**	**0.002**
b. Results of performance using CFS-FPA (30 features)					
RF	0.992	0.992	0.992	0.992	0.004
FPA	0.990	0.989	0.990	0.989	0.003
Hybrid CFS_FPA	**0.995**	**0.995**	**0.995**	**0.995**	**0.08**

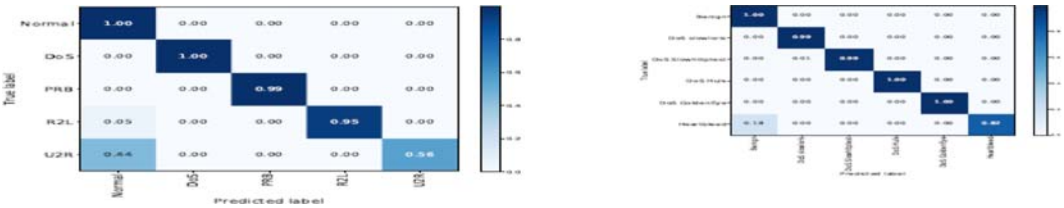
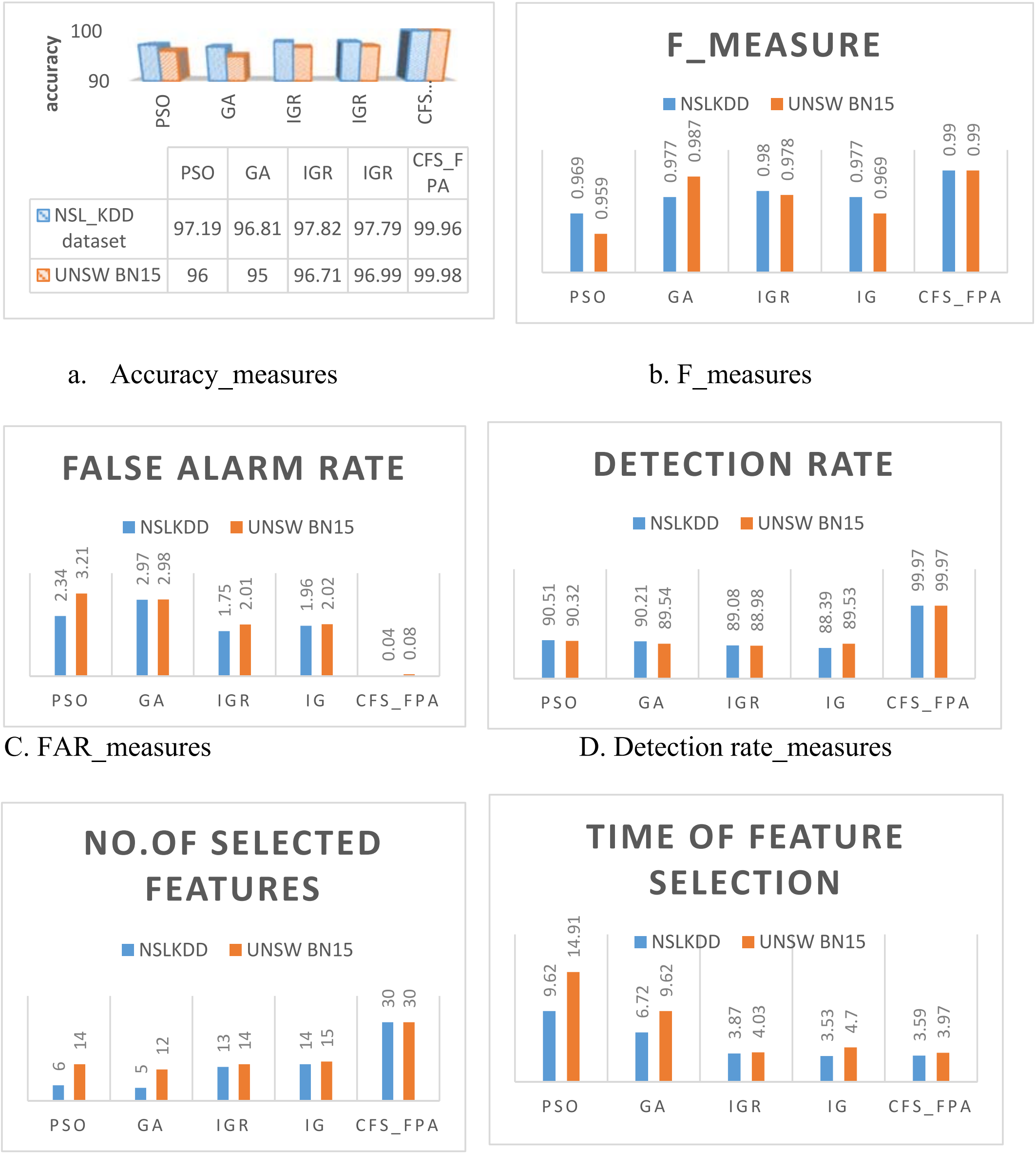


Fig. 2. Normalization process a. NSL_KDD dataset b. UNSWNB15 dataset.



	PSO	GA	IGR	IGR	CFS_FPA
▨ NSL_KDD dataset	97.19	96.81	97.82	97.79	99.96
▨ UNSW BN15	96	95	96.71	96.99	99.98

a. Accuracy_measures

b. F_measures

C. FAR_measures

D. Detection rate_measures

E. Number of selected features

F. Time of feature selection

Fig. 3. Performance measures of NSL and UNSW BN15 datasets.

Table 8
Confusion_matrix explain the true negative of all classes of datasets.

a.NSL_KDD dataset with five classes

	Normal	DOS	probe	R2L	U2R
Normal	**310**	**1**	**0**	**15**	**0**
DOS	**0**	**9666**	**0**	**11**	**0**
probe	**0**	**16**	**9548**	**0**	**0**
R2L	**18**	**7**	**0**	**9511**	**2**
U2R	**0**	**1**	**0**	**2**	**9711**

b.UNSW BN15 dataset

	normal	intrusion
Normal	**16774**	**0**
Intrusion	**0**	**35829**

basis of the NSL-KDD and UNSW BN15 datasets, as shown in Fig. 2 (d). Compared with earlier feature selection methodologies, our proposed model dramatically reduced FAR on each dataset, assuring the IDS's efficacy. Figs. 2(e) and 3(f) show the number of features selected using various approaches and the time it took to pick them, respectively, which may reflect the efficiency of an IDS.

Although the proposed methodology takes longer time than IG and IGR, CFS-FPA chooses fewer features and has considerably higher accuracy than IG and IGR, as shown in Fig. 2 (a). GA and PSO are dependent feature selection methods that acquire fewer features than CFS-FPA on the UNSW BN15 dataset, but they take longer to FS on all five sets and do not improve detection accuracy. To explain in detiels these results confusion matrix was demonstrated in Table 8.

5.4. CFS_FPA Complexity_Time and runtime

To evaluate the complexty time of proposed algorithm with runtime using big o notation.

For both datasets NSL_KDD and UNSW BN15. Compute the complextity time of the proposed Algorithm 1 as follow:

Algorithm 1: Hybrid CFS-FPA for feature selection

```
Input: datasets
Output: most effective features (X best)
    9.  Split datasets into two parts: a training set and a testing set. O(1)
    10. Pre-processing the steps by using algorithm 2.
    11. Initialization: iteration t=1.  O(1)
    12. Divided the datasets into classes  O(1)
    13. for a training_set part do :   o(n)
        j. At the first step committee the initialization of each CFS using Eq.1. o(1)
        k. Generate new RF using Eq. 2.  O(1)
        l. Initialize each feature weight value Wi by applying Eq. 5.  O(1)
        m. Generate the number of RF as 10_forests (estimators), n_jobs=2, n_feature
           selection=30, and step=1.  O(1)
        n. Xi selection from Xbest.  O(n)
        o. While Xbest > Xi do :   o(n)
            h. Use Eq. 3 to generate new Xnew.  O(1)
            i. Compute Xnew CFS using Eq. 1. O(1)
            j. If Xi<Xnew and N(0,1)<Atti then  o(1)
            k. Update Xi using Eq. 3.  O(1)
            l. Compute Wi for each Xi generated from RF using Eq. 4.  O(1)
            m. Compute $\sigma^2$ for each Xi using Eq. 5 generated from RF.  O(1)
            n. Update Wi and $\sigma^2$  O(1)
        p. end if
        q. end while
        r. Xbest= Xnew  O(1)
    14. Endfor
    15. $t = t + 1$  O(1)
    16. Output: the best subset selected (Xbest).
```

Therefore; the time complexity of this algorithm is: Big O:O(N2) and run time increase when the input is increases.

6. Conclusions

Although various machine learning strategies for enhancing the efficacy of IDSs have been described, current IDSs are not successful. On the basis of the recommended feature selection by using hybrid approaches, in this study, we describe a novel intrusion detection approach for dealing with imbalanced and high-dimensional network traffic. To determine the optimum subset based on function correlation, a hybrid CFS_FPA algorithm was introduced for a sample of 30 features. The final experimental results of CFS_FPA when using NSLKDD dataset are: acccuracy is 0.997%, precision is 0.998%, F-measure is 0.997%, DR is 0.998%, and FAR is 0.004**.** The UNSW NB15 results are: Accuracy is 0.995%, Precision is 0.995%, F-measure is 0.995%, DR is 0.995%, and FAR is 0.008. Compared with the no-feature-selection technique, the outcomes on a variety of measures are favorable.

Our technique surpasses similar FS methods in terms of: Acc, DR, F-Measure, and performance. FAR should be kept to a bare minimum. Furthermore, our approach surpasses existing classification algorithms and the suggested CFS-FPA Ensemble methodology. In the intrusion detection business, this can give a major competitive advantage as evidenced by comparison with state-of-the-art techniques. Although the suggested CFS FPA Ensemble technique is more efficient, further work may be necessary to increase its capacity to deal with rare network traffic threats. Except the methods used in the paper, some of the most representative computational intelligence algorithms can be used to solve the problems, like monarch butterfly optimization (MBO), earthworm optimization algorithm (EWA), elephant herding optimization (EHO), moth search (MS) algorithm, Slime mould algorithm (SMA), and Harris hawks optimization (HHO).

References

[1] X. Sun, J. Dai, P. Liu, A. Singhal, J. Yen, Using Bayesian networks for probabilistic identification of zero-day attack paths, IEEE Trans Inf Forensics Secur 13 (2018) 2506–2521, https://doi.org/10.1109/TIFS.2018.2821095.

[2] M. Alazab, Profiling and classifying the behavior of malicious codes, J Syst Software 100 (2015) 91–102, https://doi.org/10.1016/j.jss.2014.10.031.

[3] I. Sumaiya Thaseen, C. Aswani Kumar, Intrusion detection model using fusion of chi-square feature selection and multi class SVM, J King Saud Univ - Comput Inf Sci. 29 (2017) 462–472, https://doi.org/10.1016/j.jksuci.2015.12.004.

[4] S. Rajagopal, P.P. Kundapur, K.S. Hareesha, A stacking ensemble for network intrusion detection using heterogeneous datasets, Secur Commun Network 2020 (2020) 1–9, https://doi.org/10.1155/2020/4586875.

[5] 2015 IEEE symposium on computational intelligence for security and defense applications, CISDA 2015 - proceedings, in: 2015 IEEE symp comput intell secur def appl CISDA 2015 –proc, 2015.

[6] A. Khraisat, I. Gondal, P. Vamplew, J. Kamruzzaman, Survey of intrusion detection systems: techniques, datasets and challenges, Cybersecurity 2 (2019) 1–22, https://doi.org/10.1186/s42400-019-0038-7.

[7] S.H. Hashem, Enhance network intrusion detection system by exploiting br algorithm as an optimal feature selection, in: Handb res threat detect countermeas netw secur, 2014, pp. 17–32, https://doi.org/10.4018/978-1-4666-6583-5.ch002.

[8] J.H. Assi, A.T. Sadiq, NSL-KDD dataset classification using five classification methods and three feature selection strategies, J Adv Comput Sci Technol Res 7 (2017) 15–28.

[9] G. Kim, S. Lee, S. Kim, A novel hybrid intrusion detection method integrating anomaly detection with misuse detection, Expert Syst Appl 41 (2014) 1690–1700, https://doi.org/10.1016/j.eswa.2013.08.066.

[10] G. Omer, P. Kot, W. Atherton, M. Muradov, M. Gkantou, A non-destructive electromagnetic sensing technique to determine chloride level in maritime concrete, Karbala Int J Mod Sci 7 (2021) 61–71, https://doi.org/10.33640/2405609X.2408.

[11] M. Kabriti, E.D.A.M. Léonce, C. Merbouh, B. Abdelfattah, A. Achkir, Physical-chemical characterization and heavy metals assessment of waters and sediments of sebou watershed (top Sebou, Morocco), Karbala Int J Mod Sci 7 (2021) 18–29, https://doi.org/10.33640/2405-609X.2229.

[12] S.H. Moon, Y.H. Kim, An improved forecast of precipitation type using correlation-based feature selection and multinomial logistic regression, Atmos Res 240 (2020) 104928, https://doi.org/10.1016/j.atmosres.2020.104928.

[13] A. Khraisat, I. Gondal, P. Vamplew, J. Kamruzzaman, A. Alazab, A novel ensemble of hybrid intrusion detection system for detecting internet of things attacks, Electron 8 (2019) 1–18, https://doi.org/10.3390/electronics8111210.

[14] A. Al-Jobory, Z.Y. Mijbil, Mach-Zehnder quantum interference rules in hydrocarbons with substituents, Karbala Int J Mod Sci 7 (2021) 83–89, https://doi.org/10.33640/2405-609X.2517.

[15] H.M. Noman, M.N. Jasim, A proposed adaptive least load ratio algorithm to improve resources management in software defined network open flow environment, Karbala Int J Mod Sci 7 (2021) 40–47, https://doi.org/10.33640/2405-609X.2255.

[16] S.M. Kasongo, Y. Sun, Performance analysis of intrusion detection systems using a feature selection method on the

UNSW-NB15 dataset, J Big Data 7 (2020) 1−20, https://doi.org/10.1186/s40537-020-00379-6.

[17] G. Creech, J. Hu, A semantic approach to host-based intrusion detection systems using contiguous and discontiguous system call patterns, IEEE Trans Comput 63 (2014) 807−819, https://doi.org/10.1109/TC.2013.13.

[18] K.K. Sahu, S.C. Nayak, H.S. Behera, Multi-step-ahead exchange rate forecasting for South Asian countries using multi-verse optimized multiplicative functional link neural networks, Karbala Int J Mod Sci 7 (2021) 48−60, https://doi.org/10.33640/2405-609X.2278.

[19] M. Jabardi, A.S. Hadi, Twitter fake account detection and classification using ontological engineering and semantic web rule language, Karbala Int J Mod Sci 6 (2020) 404−413, https://doi.org/10.33640/2405-609X.2285.

[20] S. Aljawarneh, M. Aldwairi, M.B. Yassein, Anomaly-based intrusion detection system through feature selection analysis and building hybrid efficient model, J Comput Sci 25 (2018) 152−160, https://doi.org/10.1016/j.jocs.2017.03.006.

[21] H.S. Hota, A.K. Shrivas, Decision tree techniques applied on NSL-KDD data and its comparison with various feature selection techniques, in: Smart innov syst technol, 2014, pp. 205−212, https://doi.org/10.1007/978-3-319-07353-8_24.

[22] C. Khammassi, S. Krichen, A GA-LR wrapper approach for feature selection in network intrusion detection, Comput Secur 70 (2017) 255−277, https://doi.org/10.1016/j.cose.2017.06.005.

[23] Y. Feng, S. Deb, G.G. Wang, A.H. Alavi, Monarch butterfly optimization: a comprehensive review, Expert Syst Appl 168 (2021) 114418, https://doi.org/10.1016/j.eswa.2020.114418.

[24] Y. Feng, X. Yu, G.G. Wang, A novel monarch butterfly optimization with global position updating operator for large-scale 0-1 knapsack problems, Mathematics 7 (2019) 1−31, https://doi.org/10.3390/math7111056.

[25] Y. Feng, G.G. Wang, W. Li, N. Li, Multi-strategy monarch butterfly optimization algorithm for discounted {0-1} knapsack problem, Neural Comput Appl 30 (2018) 3019−3036, https://doi.org/10.1007/s00521-017-2903-1.

[26] G.G. Wang, S. Deb, L. Dos Santos Coelho, Earthworm optimisation algorithm: a bio-inspired metaheuristic algorithm for global optimisation problems, Int J Bio-Inspired Comput 12 (2018) 1−22, https://doi.org/10.1504/ijbic.2018.093328.

[27] J. Li, H. Lei, A.H. Alavi, G.G. Wang, Elephant herding optimization: variants, hybrids, and applications, Mathematics 8 (2020), https://doi.org/10.3390/MATH8091415.

[28] G.G. Wang, Moth search algorithm: a bio-inspired metaheuristic algorithm for global optimization problems, Memetic Comput 10 (2018) 151−164, https://doi.org/10.1007/s12293-016-0212-3.

[29] I. Syarif, E. Zaluska, A. Prugel-Bennett, G. Wills, Application of bagging, boosting and stacking to intrusion detection, in: Lect notes comput sci (including subser lect notes artif intell lect notes bioinformatics), 2012, pp. 593−602, https://doi.org/10.1007/978-3-642-31537-4_46.

[30] B. Schölkopf, J.C. Platt, J. Shawe-Taylor, A.J. Smola, R.C. Williamson, Estimating the support of a high-dimensional distribution, Neural Comput 13 (2001) 1443−1471, https://doi.org/10.1162/089976601750264965.

[31] A. Khraisat, I. Gondal, P. Vamplew, J. Kamruzzaman, A. Alazab, Hybrid intrusion detection system based on the stacking ensemble of C5 decision tree classifier and one class support vector machine, Electron 9 (2020) 1−18, https://doi.org/10.3390/electronics9010173.

[32] L. Koc, T.A. Mazzuchi, S. Sarkani, A network intrusion detection system based on a Hidden Naïve Bayes multiclass classifier, Expert Syst Appl 39 (2012) 13492−13500, https://doi.org/10.1016/j.eswa.2012.07.009.

[33] S.S. Sivatha Sindhu, S. Geetha, A. Kannan, Decision tree based light weight intrusion detection using a wrapper approach, Expert Syst Appl 39 (2012) 129−141, https://doi.org/10.1016/j.eswa.2011.06.013.

[34] S. Maza, M. Touahria, Feature selection algorithms in intrusion detection system: a survey, KSII Trans Internet Inf Syst 12 (2018) 5079−5099, https://doi.org/10.3837/tiis.2018.10.024.

[35] J. Mi, K. Wang, P. Li, S. Guo, Y. Sun, Software-defined green 5G system for big data, IEEE Commun Mag 56 (2018) 116−123, https://doi.org/10.1109/MCOM.2017.1700048.

[36] V. Hajisalem, S. Babaie, A hybrid intrusion detection system based on ABC-AFS algorithm for misuse and anomaly detection, Comput Network 136 (2018) 37−50, https://doi.org/10.1016/j.comnet.2018.02.028.

[37] V. Bolón-Canedo, N. Sánchez-Maroño, A. Alonso-Betanzos, Feature selection for high-dimensional data, Prog Artif Intell 5 (2016) 65−75, https://doi.org/10.1007/s13748-015-0080-y.

[38] H. Liu, L. Yu, Toward integrating feature selection algorithms for classification and clustering, IEEE Trans Knowl Data Eng 17 (2005) 491−502, https://doi.org/10.1109/TKDE.2005.66.

[39] N. Acharya, S. Singh, An IWD-based feature selection method for intrusion detection system, Soft Comput 22 (2018) 4407−4416, https://doi.org/10.1007/s00500-017-2635-2.

[40] X.Y. Chen, L.Z. Ma, N. Chu, M. Zhou, Y. Hu, Classification and progression based on CFS-GA and C5.0 boost decision tree of TCM Zheng in chronic hepatitis B, Evid Based Compl Altern Med 2013 (2013) 1−10, https://doi.org/10.1155/2013/695937.

[41] F. Salo, A.B. Nassif, A. Essex, Dimensionality reduction with IG-PCA and ensemble classifier for network intrusion detection, Comput Network 148 (2019) 164−175, https://doi.org/10.1016/j.comnet.2018.11.010.

[42] S. Singh, A.K. Singh, Detection of spam using particle swarm optimisation in feature selection, Pertanika J Sci Technol 26 (2018) 1355−1372.

[43] S. Singh, A.K. Singh, Web-Spam features selection using CFS-PSO, in: Procedia comput sci, 2018, pp. 568−575, https://doi.org/10.1016/j.procs.2017.12.073.

[44] S. Georganos, T. Grippa, A. Niang Gadiaga, C. Linard, M. Lennert, S. Vanhuysse, et al., Geographical random forests: a spatial extension of the random forest algorithm to address spatial heterogeneity in remote sensing and population modelling, Geocarto Int 36 (2021) 121−136, https://doi.org/10.1080/10106049.2019.1595177.

[45] Q. Feng, J. Liu, J. Gong, UAV Remote sensing for urban vegetation mapping using random forest and texture analysis, Rem Sens 7 (2015) 1074−1094, https://doi.org/10.3390/rs70101074.

[46] M.N. Adnan, M.Z. Islam, P.A. Forest, Constructing a decision forest by penalizing attributes used in previous trees, Expert Syst Appl 89 (2017) 389−403, https://doi.org/10.1016/j.eswa.2017.08.002.

[47] T. Aldwairi, D. Perera, M.A. Novotny, An evaluation of the performance of Restricted Boltzmann Machines as a model for anomaly network intrusion detection, Comput Network 144

(2018) 111−119, https://doi.org/10.1016/j.comnet.2018.07.025.

[48] S. Rosset, A. Inger, Knowledge discovery in a charitable organization's donor database, SIGKDD Explor 1 (2000) 85−90.

[49] M. Tavallaee, E. Bagheri, W. Lu, A.A. Ghorbani, A detailed analysis of the KDD CUP 99 data set, in: IEEE symp comput intell secur def appl CISDA 2009, 2009, pp. 1−6, https://doi.org/10.1109/CISDA.2009.5356528.

[50] N. Moustafa, J. Slay, Unsw-Nb15, A comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set), in: 2015 Mil commun inf syst conf MilCIS 2015 - proc, 2015, pp. 1−6, https://doi.org/10.1109/MilCIS.2015.7348942.

[51] J. Li, K. Cheng, S. Wang, F. Morstatter, R.P. Trevino, J. Tang, , et al.H. Liu, Feature selection: a data perspective, ACM Comput Surv 68 (2017) 1−45, https://doi.org/10.1145/3136625.

[52] A. Astorino, E. Gorgone, M. Gaudioso, D. Pallaschke, Data preprocessing in semi-supervised SVM classification, Optimization 60 (2011) 143−151, https://doi.org/10.1080/02331931003692557.

[53] S. Elhag, A. Fernández, A. Altalhi, S. Alshomrani, F. Herrera, A multi-objective evolutionary fuzzy system to obtain a broad and accurate set of solutions in intrusion detection systems, Soft Comput 23 (2019) 1321−1336, https://doi.org/10.1007/s00500-017-2856-4.

[54] A.I. Pratiwi, Adiwijaya, on the feature selection and classification based on information gain for document sentiment analysis, Appl Comput Intell Soft Comput 2018 (2018) 1−6, https://doi.org/10.1155/2018/1407817.

[55] L. Li, X. Zhang, M. Xue, Explaining information gain and information gain ratio in information theory, ICIC Express Lett 7 (2013) 2385−2391.

[56] L.M. Patnaik, S. Mandavilli, Adaptation in genetic algorithms, 2017, pp. 45−64, https://doi.org/10.1201/9780203713402.

[57] Y. Zhang, S. Wang, G. Ji, A comprehensive survey on particle swarm optimization algorithm and its applications, Math Probl Eng 2015 (2015) 1−39, https://doi.org/10.1155/2015/931256.