Journal of Big Data

RESEARCH Open Access

Network intrusion detection using data dimensions reduction techniques

Anita Shiravani[1*], Mohammad Hadi Sadreddini[1] and Hassan Nosrati Nahook[2]

*Correspondence:
a.shiravani@shirazu.ac.ir

[1] Computer Science and Engineering and Information Technology Department, Shiraz University, Shiraz, Iran
[2] Department of Computer Engineering, Higher Education Complex of Saravan, Saravan, Iran

Abstract

Due to the increasing growth of the Internet and its widespread application, the number of attacks on the network has also increased. Therefore, maintaining network security and using intrusion detection systems is of critical importance. The connection between devices leads to a large number of data being generated and saved. The era of "big data" emerges over time. This paper presents a new method for selecting effective features on network intrusion detection based on the concept of fuzzy numbers and scoring methods based on correlation feature selection for intrusion detection systems. The goal of this paper is to present a new approach for reducing data size using the concept of fuzzy numbers and scoring methods based on correlation feature selection for intrusion detection systems. In this method, to eliminate inefficient features and reduce data dimensions, number of features are defined as a fuzzy number, and the heuristic function of the correlation-based feature selection algorithm is expressed as a triangular fuzzy number membership function. To evaluate the proposed method, it is then compared to previous intrusion detection methods. The results show that the proposed method selects several features less than the conventional methods with a higher detection rate. The proposed method is compared with the correlation-based feature selection method on two datasets. The proposed method is evaluated and validated on KDD Cup, NSL-KDD and CICIDS datasets. The achieved accuracy is 99.9% which is 96.01% with CFS method.

Keywords: Feature selection, Big data, Network intrusion detection, Correlation feature selection, Genetic algorithm, Fuzzy concept

Introduction

Recent advancements in Information Technology (IT) have engendered the rapid production of big data, as enormous volumes of data with high dimensional features grow exponentially in different fields. Therefore, dealing with high-dimensional data creates new challenges in terms of data processing efficiency and effectiveness. To address such challenges, Feature Selection (FS) is among the most utilized dimensionality reduction methods, which is helpful in reducing the high dimensionality of large-scale data by picking up a small subset of related and significant features and eliminating unrelated and redundant features in order to construct effective prediction models. Today we are facing big and high-dimensional data, that increased number of features in datasets, this

has increased the computational cost. In this situation, Feature selection is an effective and an essential step in preprocessing.

On the other hand, as computer networks become an important part of the current world, the threats to it also increase day by day. To detect various threats, intrusion detection systems are needed. So The purpose of this study is to present a new feature selection method that is computationally efficient and effective. The goal of this paper is to present a new approach for reducing data size using the concept of fuzzy numbers and scoring methods based on correlation feature selection for intrusion detection systems. In general feature selection algorithms, there is no control over the number of selected features. However, by using the proposed method, the number of selected features can be modified and can be increased or decreased based on the problem at hand. In proposed method, to eliminate inefficient, number of features of the data set is defined as a triangular fuzzy number. In the next step, the heuristic function of the correlation feature selection algorithm (CFS) is selected as the membership function of the aforementioned triangular fuzzy number. In the next step, the genetic algorithm is used to search the problem space and select the optimal features from the built subsets. Moreover, the proposed method employs the scoring function of the CFS algorithm as a measure of merits regarding each subset created from the data set. Consequently, by using the score created by the function, the subset with the highest score is selected, and its features are chosen by the genetic algorithm. The results shown that the proposed approach can reduce the false alarm rate and control the number of features in network intrusion detection systems.

The structure of the article is as follows. "Related work" section discusses previous related work. "Background" section describes some of the background information about CFS algorithm, Fuzzy concept and Genetic algorithm. Consequently, "The proposed method" section discusses the details of the proposed feature selection method. "Experimental environment" section describes preparation phase. The results of the proposed method presented in "Results and discussions" section. Proposed method comparison with other methods are presented in "Comparison" section. Finally, "Conclusion" section expresses the conclusion.

Intrusion detection systems

An attack is defined as an activity that violates a network security policy. Any attempt to establish or obtain unauthorized access to the information of an individual or organization is considered as intrusion. An intrusion detection system is a software and hardware used to perform the process of detecting unauthorized use or attacking a computer or telecommunications network. To prevent confidential data from getting in the hands of an attacker, some measures need to be taken. Here comes the need for an effective system, which can classify the traffic as an attack or normal [25]. Intrusion Detection Systems can do this work with perfection. Intrusion detection systems (IDS) are divided into two categories according to performance: signature-based IDS (SIDS) and anomaly-based IDS (AIDS) [4]. The IDS based on the Signature is a type of intrusion detection that detects attacks by matching patterns of previous known attacks. In SIDS, matching methods are used to determine a previous intrusion. As previously noted, when a signature matching one of the existing signatures is encountered, an alarm signal is triggered.

The disadvantage of the signature-based approach is that it cannot detect new attacks and needs constant updating.

AIDS research consists of two separate stages such as training phase and the testing phase. In the training phase of the experiment, a model is built of typical traffic behaviour and then in the testing phase, a new data set is used to validate the model. AIDS may be categorized according to the type of teaching used, for example, statistical, knowledge-oriented, and machine learning [5]. The biggest benefit of AIDS is the potential to detect zero-day attacks so it doesn't involve signatures of irregular users' behaviours. It activates a threat alert for activities that are in disarray [6, 7]. AIDS includes various possible impacts. They have the potential to detect secret cyber-security risks. Therefore, due to the weakness of SIDS in detecting new attacks, in this paper, we used a AIDS approach.

Feature selection methods

It has been observed that feature selection methods have enhanced the quality of prediction model by eliminating the irrelevant features and utilized only important features while building the prediction models [8]. The feature selection is an essential data preprocessing stage in data mining. The core principle of feature selection seems to be to pick a subset of possible features by excluding features with almost no predictive information as well as highly associated redundant features [9]. In the past several years, a variety of meta-heuristic methods were introduced to eliminate redundant and irrelevant features as much as possible from high-dimensional datasets [10]. Among the main disadvantages of present meta-heuristic based approaches is that they are often neglecting the correlation between a set of selected feature [9]. In this paper, for the purpose of feature selection we proposed hybrid method by considering the correlation between selected features. The goal of feature selection is to choose a subset of available features with the lowest redundancy with each other and the highest relevancy to the target class [11]. Traditional techniques to reduce the dimensions are divided into two main categories: feature extraction and feature selection. In the first approach, instead of the original features, secondary features with low dimensions are extracted. That means that a high dimensional space is transferred to low dimensional space. However, the second approach includes four sub-categories that include filter method, wrapper method, hybrid methods, and embedded methods [12]. The subset of features in the pre-processing step is selected in filter methods independent of any learner method [13].

Feature selection methods are classified into four general categories. Which two common methods are demonstrated in Figs. 1 and 2.

Related work

In this section, we review some of the related feature selection methods which are performed for Intrusion Detection Systems. Various methods have been suggested for features selection in IDSs.

Omar Fitian Rashid et al. [1] proposed a new features selection method proposed based on DNA encoding and on DNA keys positions that has three phases, the first phase, is called pre-processing phase, which is used to extract the keys and their positions, the second phase is training phase; the main goal of this phase is to select

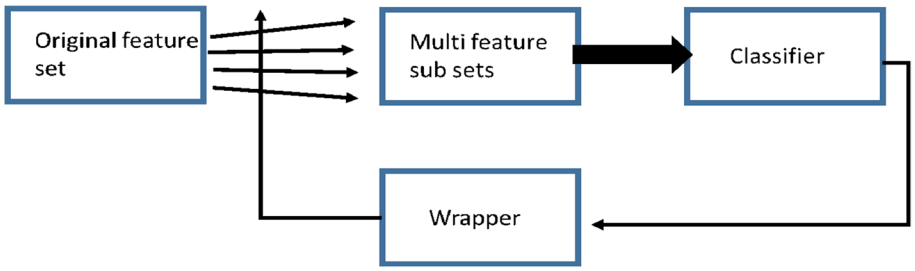


Fig. 1 Wrapper method

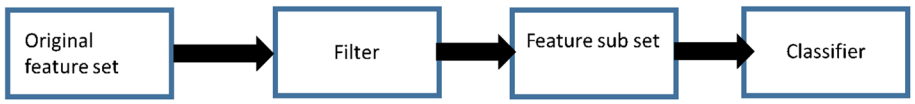


Fig. 2 Filter method

features based on the key positions that gained from pre-processing phase. In the training phase, the first step is to find specific training dataset records features based on keys positions (features that contain positions values) then encode all records of training dataset (for specific features). And the last step is to store the features numbers (DNA locations). The third phase is the testing phase, which classified the network traffic records as either normal or attack by using specific features. The first step is to Encode all records of the testing dataset (for specific features based on DNA locations) then Looking for the keys in these records after encoding, and as a last step, Calculate the normal records and attacks records based on matching or mismatching with these keys and then Calculate the experiment results (DR, FAR, accuracy, and the time) The performance is calculated based on the detection rate, false alarm rate, accuracy, and also on the time that include both encoding time and matching time. All these results are based on using two or three keys, and it is evaluated by using two datasets, namely, KDD Cup 99, and NSL-KDD. The proposed method is having a problem with building time which is much more than other methods. Shamman et al. [7] utilized the Information Gain-based algorithm. The algorithm chooses the features optimal number from dataset of NSL-KDD. Additionally, integrated selection of feature with the technique of machine learning namely as Support Vector Machine (SVM) by utilizing the algorithm of artificial bee colony as well as Optimization-Cuckoo Search Algorithm for optimizing SVM hyper parameters for dataset effective classification. Although information gain is usually a good measure for deciding the relevance of an attribute, it is not perfect. A notable problem occurs when information gain is applied to attributes that can take on a large number of distinct values. For example, suppose that one is building a decision tree for some data describing the customers of a business. Information gain is often used to decide which of the attributes are the most relevant, so they can be tested near the root of the tree. One of the input attributes might be the customer's credit card number. This attribute has a high mutual information, because it uniquely identifies each customer, but we do *not* want to include it in the decision tree: deciding how to treat a customer based on their credit card number is unlikely to generalize to customers we haven't seen before.

Filtering algorithms evaluate feature mainly based on the statistical performance of the dataset, and are widely used in intrusion detection systems because of their fast feature selection speed. Azhagusundari et al. [6] proposed a filtering algorithm based on information gain to solve the problem of redundant features in network data. It calculates the information gain of each feature and screens feature set by setting a threshold. Osanaiye et al. [14] proposed that combining IG, GR, and ReliefF select the combined features of network data, and a decision tree is used for detecting screened feature set. The results show that it effectively reduces the redundant features of network data and significantly improves the detection rate of intrusion detection systems. Considering the correlation between each feature of network data, Ambusaidi et al. [15] proposed selecting feature set by calculating mutual information of features and detecting the selected feature set by using a least squares support vector machine. It has higher detection rate and lower computational cost on KDD Cup 99 dataset. Gu et al. [11] proposed a hybrid algorithm for intrusion detection systems. It orders features by using a classifier at the filtering stage, and then search for features by using the Sequential Forward Selection (SFS) algorithm. SFS is also well known method to select the best feature. This method is consisting of two variants. One is called SFS and other is known as sequential backward selection (SBS). In SFS, features are sequentially added to an empty candidate set until the addition of further features does not decrease the criterion. In case of SBS, features are sequentially removed from full candidate set until the removal of further features increase the criterion. The problem is that the new features are added continuously in the selected features set. It does not give flexibility to remove the features that have been already added in case they have become obsolete after the addition of new features. Due to the problems of the mentioned methods, using a combined method can improve the performance. As seen in the previous methods, there is no control over the number of selected features. In fact, no mechanism has been designed for it. By using fuzzy concepts, this mechanism can be designed and solved by combining with filtering methods. However, by using the proposed method, the number of selected features can be modified and can be increased or decreased based on the problem at hand. In this paper we proposed a feature selection method that used fuzzy concept which combined with a filter selection method and meta-heuristic search algorithm (genetic).

Background

Due to the extension of network-based services, the requirement of network security has increased than ever. Although, various technologies like cryptography, steganography, firewalls, and intrusion systems have been introduced to apply security on the information and network services [26]. To prevent confidential data from getting in the hands of an attacker, some measures need to be taken. Here comes the need for an effective system, which can classify the traffic as an attack or normal. Intrusion Detection Systems can do this work with perfection [27]. Intrusion detection systems detect an intrusion or attack and inform the administrator or the handler regarding the attack [28].

This section briefly discusses the following topics: Fuzzyfication, CFS, Genetic algorithm.

Fuzzyfication

In this section, the fuzzy concepts used in the projects are explained in detail. There is no absolute logic in the fuzzy system. A fuzzy set allows its members to have different degrees of membership ranging from [0, 1]. This concept helped us for design a controller mechanism for number of selected features. The fuzzy logic attempts to solve problems with a range of data that makes it possible to reach a set of accurate conclusions [16]. Membership functions play an essential role in the overall performance of fuzzy systems. They have different shapes like triangles, trapezoid, etc.... The only condition that these functions have to satisfy is that they must vary between 0 and 1 [17]. Since, there is an unlimited number of methods on fuzzy systems, there is an unlimited number of graphically plotting methods for the functions. Additionally, the membership function is a function that defines how each point of the input space is plotted between 0 and 1. The value assigned to each member indicates the merit of each member with that set. Triangular membership function, one of the different types of membership functions that used in this research, mentioned below [12].

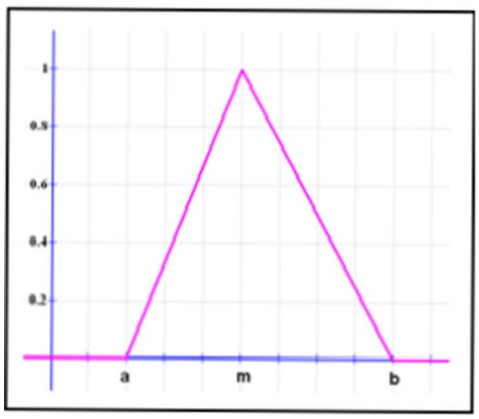
Triangular membership function

There are two types of triangular fuzzy numbers, symmetric and asymmetric. The triangular fuzzy number is defined using the lower bound a and the upper bound b and the middle-value m. Figure 3 shows a symmetrical triangular fuzzy number. In this research, a triangular fuzzy number has been used.

Membership functions assign a degree of membership to each member of the group that indicates how close that member is to the concept of that group [18]. In the proposed method, number of features for the desired data set are defined as a fuzzy number. The heuristic function used by the CFS to weigh the features is described as a triangular fuzzy number membership function.

CFS

Correlation measurement is based on the linear correlation method that is used to determine the linear relationships between two random variables and is known for its simplicity and low calculation cost. The effective correlation between two



$$\mu_A(x) = \begin{cases} 0, & x \le a \\ \frac{x-a}{m-a}, & a < x \le m \\ \frac{b-x}{b-m}, & m < x < b \\ 0, & x \ge b \end{cases}$$

Fig. 3 Triangular membership function

variables or features indicates the degree of relationship between them, which is equal to the rate of covariance divided by their standard deviation. Here cov is the covariance. σX is the standard deviation of X and σY is the standard deviation of Y. The given equation for correlation coefficient can be expressed in terms of means and expectations. Formula 1 shows this relation. If a relationship exists between the random variables, the coefficient is between 1 and -1 and if there is no relationship, it is equal to zero [10].

$$Corr(X, Y) = \frac{\mathrm{Cov(X, Y)}}{\sigma \mathrm{X} \sigma \mathrm{Y}}. \tag{1}$$

The main concept of the correlation-based feature selection algorithm is its heuristic search strategy. By default, the correlation-based feature selection algorithm uses the best-first method to determine the search space. This algorithm starts with an empty set of features, and at each stage of the search, all possible subsets of features are created. The new subsets are evaluated using the evaluation metrics. Consequently, according to the merit and weight obtained from the evaluation formula, they are added to the priority queue [19]. In the aforementioned text, correlation-based feature selection tends to select feature subsets with low correlation among the features along with features with high correlation with class. Correlation-based feature selection generally evaluates the suitability of a feature subset according to two concepts. The first is the relationship between the class and the features and the second is the correlation between the features. The function calculates the merit of a feature subset with k attribute by considering the average feature-class correlation (RCF), and the average feature-feature correlation (RFF).

CFS ranks attributes according to a heuristic evaluation function based on correlations. The function evaluates subsets made of attribute vectors, which are correlated with the class label, but independent of each other. The CFS method assumes that irrelevant features show a low correlation with the class and therefore should be ignored by the algorithm. On the other hand, excess features should be examined, as they are usually strongly correlated with one or more of the other attributes. The approach of this method helpe to select more useful features. CFS is one of well-known techniques to rank the relevance of features by measuring correlation between features and classes and between features and other features.

In general feature selection algorithms, there is no control over the number of selected features. However, by using the proposed method, the number of selected features can be modified and can be increased or decreased based on the problem at hand. In the first step, the number of features of the data set is defined as a triangular fuzzy number. In the next step, the heuristic function of the CFS is selected as the membership function of the aforementioned triangular fuzzy number. Moreover, the proposed method employs the scoring function of the CFS algorithm as a measure of merits regarding each subset created from the data set.

Also, in proposed method, the function has been used as a fuzzy number membership function where each subset gets a degree of merit, and with the concept of the triangular fuzzy number, the maximum amount of the function can be determined.

Genetic

Genetic algorithm is one of the meta-heuristic algorithms that has many applications in solving optimization problems and is one of the innovative methods based on objective functions. In the first stage, the best choice is made, then the chosen ones are considered as parents, and in the next stage, there is a crossover, and children are obtained from the parents. In the next stage, a mutation is made and a new population is created and this new population. It is considered as the initial population and then these steps continue again until the best answer is found and selected [2].

Fitness function

In the search process of the genetic algorithm, the fitness function is the sole basis for controlling the search direction. Fitness function assures that the survival probability of a single bit increased throughout the evolutionary process. Genetic algorithm is more efficient with large search space and has a little probability of reaching local optimal solution than other algorithms. Genetic algorithms are stochastic optimization procedure which works efficiently to choose the small subset of features for classification with a lower computational requirement. Often the fitness function of the genetic algorithms employed for feature selection task is manipulated to achieve parsimonious solutions concerning the size of the feature selection subset. To be able to identify better people from the population, we need to set a benchmark for evaluation. Depending on the type of each problem, the evaluation function is defined. For example, when looking to minimize a function, the merit value is different from when looking to maximize a function. At this point, since we are looking for a subset of features that have the highest value for data class recognition, the evaluation function is the merit function for weighing the attribute subset defined by CFS. The selection phase in the genetic algorithm produces a new generation of solutions by selecting the parents who have the highest qualifications. To apply the genetic algorithm and find the subset with the highest merit value, in the search space, the MATLAB R2016a optimization toolbox is used alongside Python 2.6.

The proposed method

This section reviews the details of the proposed feature selection method. The architecture is divided into three phases namely: Fuzzyfication, using CFS function and genetic algorithm. The proposed feature selection model aims at enhancing the performance of NIDSs. In general, feature selection can be seen as a hybrid optimization problem that includes at least two objectives, reducing the number of features and maximizing the classification performance [6]. Proposed method based on fuzzyfication, correlation feature selection and genetic algorithm, is performed via five phases.

1. Fuzzyfication and define a triangular fuzzy number (the number of features of the data set is defined as a triangular fuzzy number)
2. Defining the CFS evaluation function as a triangular fuzzy number membership function,
3. Changing Fuzzy number center

4. Find the highest level of merit with CFS for subsets
5. Search in that space with genetic algorithm and finally find the optimal subset.

In the proposed method, the fuzzy numbers represent the features. The heuristic function used by the CFS algorithm to weigh the attributes is expressed as the membership function of a triangular fuzzy number, which is a number between 0 and 1. The Fig. 4 shows the triangular fuzzy number. The triangular fuzzy number is defined using the lower bound a and the upper bound b and the middle-value m and is shown in the figure below. The middle value (m) can be altered using a parameter definition called p.

Additionally, the parameter p is defined in such a way that by altering the middle-value m, the maximum value of subsets merits, can be determined.

In other words, by changing the middle value, the base and side of the fuzzy triangle are determined, where the suitability of the feature set is maximum. This allows the evaluation function that the CFS algorithm uses to weigh the generated subsets and then identify each subset that has the highest value as the selected subset. This function is as follows. Formula 2 shows the criteria for calculating the weight of the subsets in the algorithm.

$$\text{Merits} = \frac{krcf}{\sqrt{k + k(k-1)rff}} \tag{2}$$

Consequently, by changing the center (middle-value) of the triangular fuzzy number, the value of this function changes according to the subset of features a genetic algorithm is then used to examine the subset of the feature set and, to identify better people from the population, a benchmark for evaluation needs to be set. Moreover, depending on the type of each problem, the evaluation function is determined. For example, when looking to minimize a function, the merit function is different than when looking to maximize a function. At this point, since we are looking for a subset of features that have the highest value for data class identification, the evaluation function is the merit function for weighing the feature subset determined by the CFS method. Each subset of features is weighed according to the correlation between the features of that subset and the correlation between them and the class. By changing the middle value of the triangular fuzzy number, the side of the triangle that has a subset of features with a higher degree of merit can be determined. Furthermore, the selection phase in the genetic algorithm produces a new generation of solutions by selecting the parents who have the highest

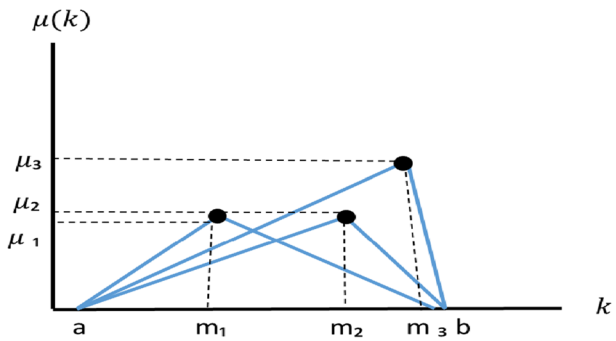


Fig. 4 The triangular fuzzy number

merits. By changing the value of p and obtaining the merit level corresponding to that p, we have obtained the merit level of a number (subsets) of features. To do this, the value of p should be initially changed and the number of different merits has to be calculated. Consequently, the features with the highest level of merits will be selected and look for the number of features related to that subset that has the maximum amount of merits. The overall procedure of our method is demonstrated in Fig. 5.

Experimental environment

All experiments were performed in an environment with the following specifications: The steps of the proposed method were performed using MATLAB R2016a and Python 2.6 coding on Windows 7×64 as the operating system. IBM SPSS and Weka tools have also been used in different stages, depending on the requirement.

Data used

KDD dataset

KDD Cup 99 is a popular dataset for evaluating intrusion detection systems. This collection contains more than one million training samples and two million test samples. Created in 1999, the KDD99 dataset is one of the most widely used databases for cybersecurity research. This dataset is based on the 1998 DARPA dataset by Cyber Systems and MIT Lincoln Laboratory Technology Group. The data set was collected from network data over a period of 9 weeks. Tulayi et al. performed statistical analysis on the KDDCUP dataset and adjusted the new NSLKDD dataset based on the original KDD dataset to solve problems [20]. The NSL-KDD database has undergone many improvements and changes to the KDD99 database. For instance, additional records have been removed from the set, which ensures that classification algorithms improve dramatically in terms of accuracy and timing, as well as improving the performance of machine learning algorithms, as each test set record is evaluated only once [21]. Attacks in the NSL-KDD database include 22 types of attacks in four categories, which are listed in Table 1.

NSL-KDD Database features are classified into 3 general types. Basic Features: this category shows the features related to TCP/IP connection. Traffic features: this group has features related to a time window and is mostly divided into host features and similar services. The third category is content features. Some attacks, such as R2L and U2R, do

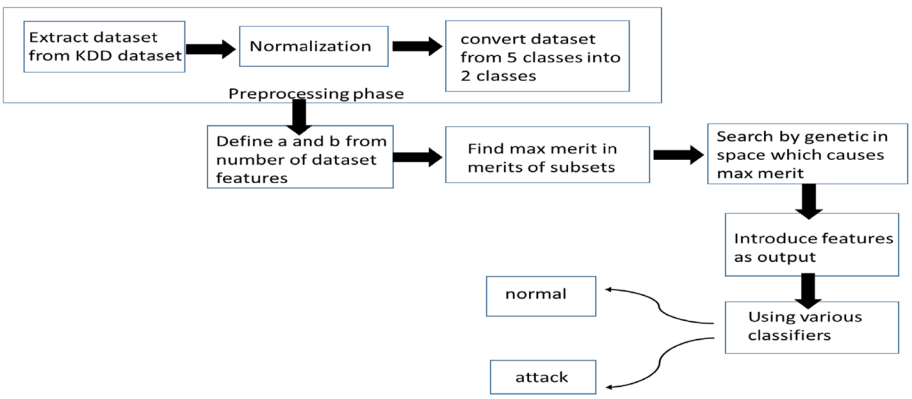


Fig. 5 P-Dataset

Table 1 Attacks in the NSL-KDD database

Categories	Attack types
DOS	Back, land, Neptune, pod, smurf, teardrop
R2L	ftp_write, guess_passwd, imap, multihop, phf, spy, warezclient, warezmaster
U2R	Buffer_overflow, loadmodule, perl, rootkit
Probe	nmap, ipsweep, portsweep, satan

not have significant sequential patterns. The nature of this type of attack is hidden in the content of the attack data. For example, the number of times the login attempt was unsuccessful is one of these features [22]. Figure 6 shows the overall structure of a data set record. The data set contains 42 features, of which the 42 feature indicates the label and the data class.

P-Dataset

The NSL-KDD train set is taken from the KDD train set and the test set of these data is also taken from the KDD test set. But in the data set used in this study, both test and train sets from the training set of the KDD set have been extracted, in this paper we call it, P-dataset and can be seen in Fig. 7.

To extract this data set, two sets of Train and Test in the ratio of 70 to 30 were considered from KDD data. Hence, the ratio of normal class data scatter and attacks in it are maintained as KDD data. The ratio of attacks to normal traffic in the KDD database and the proposed data set is 80 to 20. The size of the extracted data set and the percentage of normal class scatter and attack are given in Table 2.

This data set contains 22 types of attacks. The 22 types of attacks have been classified into the 4 general classes of attacks listed in Table 4-1. With a normal traffic class, a data set with 5 classes were determined. As specified, 70 to 30 of the test and training

```
0,tcp,telnet,SF,121,174,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,1,1,0.0
0,0.00,0.00,0.00,1.00,0.00,0.00,255,120,0.47,0.02,0.00,0.00,
0.00,0.00,0.01,0.03,guess_passwd
```

Fig. 6 The overall structure of the proposed method

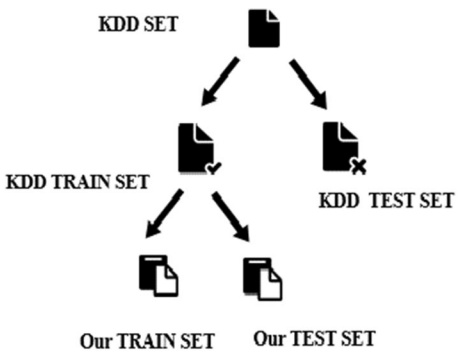


Fig. 7 The overall KDD dataset structure

Table 2 Specifications of the proposed data set

Dataset	Total Number of Instances	Train	Test	Number of classes	Attacks	Normal
KDD CUP	177,435	123,505	53,930	23	80%	20%
P-Dataset	123,505	86,453	37,052	5	80%	20%

Table 3 Details of the CICIDS2017 dataset

Dataset Name	CICIDS2017
Dataset Type	Multi-Class
Year of Formation	2017
Duration of Capture	5 Days
Attack Infrastructure	4 PCs, 1 Router, 1 Switch
Victim Infrastructure	3 Servers, 1 Firewall, 2 Switches, 10 PCs
Total number of Features	80
Number of Classes	15

sets have been extracted from the training data of the KDDCup99 database by maintaining the dispersion of the percentage of attack and normal data. The proposed data set includes 86,453 records in training data and 35,052 in test data. We have normalized the desired data set into 2 attack classes using IBM SPSS Statistic 23 software.

CICIDS

To evaluate the effectiveness of intrusion detection and learning systems, the Canadian Institute of Cyber Security presented a data set called CICIDS2017, which includes the latest threats and attacks and their characteristics [23]. This dataset has attracted the attention of many researchers because it includes attacks that have not been addressed by previous datasets. This dataset includes normal traffic and up-to-date attacks, which are similar to real-world data and have 80 features. To create the dataset, they created 22 abstract behaviours based on HTTP, HTTPS, FTP, SSH, and email protocols [23], and the details regarding the CICIDS2017 dataset can be seen in Table 3.

The data recording period started at 9 am on Monday, July 3, 2017, and ended at 5 pm on Friday, July 7, 2017, for a total of 5 days. Monday traffic is normal and includes only normal traffic. Attacks include Brute Force FTP, Brute Force SSH, DoS, Heartbleed, Web Attack, Intrusion, Botnet, and DDoS, which are carried out in the morning and afternoon on Tuesdays, Wednesdays, Thursdays, and Fridays [24]. Table 4 shows the features of the CICIDS2017 dataset.

The data for the different days and the labels, specified for those days, by kind and type of attack, are given in Table 5.

Dataset preparation

Preparation of CICIDC

The attack class label of this data set includes 14 categories of attacks. This data set is 15 classes, which categorizes all attacks into 7 classes and then, for better comparison, the

Table 4 Features of CICIDS dataset

NO	Feature	NO	Feature	NO	Feature
1	Source Port	28	Bwd IAT Total	55	Average Packet Size
2	Destination Port	29	Bwd IAT Mean	56	Avg Fwd Segment Size
3	Protocol	30	Bwd IAT Std	57	Avg Bwd Segment Size
4	Flow Duration	31	Bwd IAT Max	58	Fwd Avg Bytes/Bulk
5	Total Fwd Packets	32	Bwd IAT Min	59	Fwd Avg Packet /Bulk
6	Total Backward Packets	33	Fwd PSH Flags	60	Fwd Avg Bulk Rate
7	Total Length of Fwd Pck	34	Bwd PSH Flags	61	Bwd Avg Bytes/Bulk
8	Total Length of Bwd Pck	35	Fwd URG Flags	62	Bwd Avg Packet /Bulk
9	Fwd Packet Length Max	36	Bwd URG Flags	63	Bwd Avg Bulk Rate
10	Fwd Packet Length Min	37	Fwd Header Length	64	Subflow Fwd Packets
11	Fwd pck Length Mean	38	Bwd Header Length	65	Subflow Fwd Bytes
12	Fwd Packet Length Std	39	Fwd Packets/s	66	Subflow Bwd Packets
13	Bwd Packet Length Max	40	Bwd Packets/s	67	Subflow Bwd Bytes
14	Bwd Packet Length Min	41	Min Packet Length	68	Init_Win_bytes_fwd
15	Bwd Packet Length avg	42	Max Packet Length	69	Act_data_pkt_fwd
16	Bwd Packet Length Std	43	Packet Length Mean	70	Min_seg_size_fwd
17	Flow Bytes/s	44	Packet Length Std	71	Active Mean
18	Flow Packets/s	45	Packet Len. Variance	72	Active Std
19	Flow IAT Mean	46	FIN Flag Count	73	Active Max
20	Flow IAT Std	47	SYN Flag Count	74	Active Min
21	Flow IAT Max	48	RST Flag Count	75	Idle Mean
22	Flow IAT Min	49	PSH Flag Count	76	Idle packet
23	Fwd IAT Total	50	ACK Flag Count	77	Idle Std
24	Fwd IAT Mean	51	URG Flag Count	78	Idle Max
25	Fwd IAT Std	52	CWE Flag Count	79	Idle Min
26	Fwd IAT Max	53	ECE Flag Count	80	Lable
27	Fwd IAT Min	54	Down/Up Ratio		

Table 5 Collected data on different days related to CICIDS dataset

DAY Activity	Attacks Found
Monday	Benign
Tuesday	Benign, FTP-Patator, SSH-Patator
Wednesday	Benign, Dos GoldenEye, Dos Hulk, Dos Slowhttptest, Dos Slowloris, Heartbleed
Thursday	Benign, Web Attack-Brute Force, Web Attack-Sql Injection, Web Attack-XSS
Thursday	Benign, Infiltarion
Friday	Benign, Bot
Friday	Benign, PortScan
Friday	Benign, DDos

set is converted into a two-class normal-attack data set. Table 6 shows the new label and the old data label. Then the data was converted from CSV format to ARFF and prepared for testing.

Table 6 New label and the old data label of CICIDS

New Labels	Old Labels	Number of instances
Normal	Benign	535,908
Bot	Bot	1966
Brute Force	FTP-Patator, SSH-Patator	13,835
Dos/DDos	DDos, Dos GoldenEye, Dos Hulk, Dos Slowhttptest, Dos Slowloris, Heartbleed	294,506
Infiltarion	Infiltarion	36
PortScan	PortScan	158,930
Web Attack	Web Attack-Brute Force, Web Attack-Sql Injection, Web Attack-XSS	2180

Table 7 Invalid features of CICIDS dataset

Feature number in dataset	34	36	58	59	60	61	62	63
Name	Bwd PSH Flags	Bwd URG Flags	Fwd Avg Bytes/Bulk	Fwd Avg Packets/ Bulk	Fwd Avg Bulk Rate	Bwd Avg Bytes/Bulk	Bwd Avg Packets/ Bulk	Bwd Avg Bulk Rate

In the data set, raw data usually contains unusual and irrelevant samples that may negatively affect the accuracy of the classification. To solve the problem, it is necessary to remove these records from the data set at the beginning of the tests. For example, the Flow Packets/s feature in the CIC-IDS2017 dataset includes abnormal values such as infinite values or Not a Number (NaN). Additionally, the CIC-IDS2017 dataset contains 8 invalid features that have the same value for all records, which in turn generates infinite values or NaN. These features are listed in Table 7.

Preparation of P-Dataset

In the proposed data set, after extracting the desired set from KDD data, done by maintaining the scatter rate of attacks, the data is classified into five general classes and then converted from nominal to numerical data followed by normalization. For a better and more accurate comparison, we take all the attacks to the attack class and convert the data set to a set with 2 normal and attack classes, and compare the results at the end. At this stage, the preparation phase of the data set is over. Finally, the results of the proposed method are extracted and applied to the data set and CICIDS. This is examined below.

Evaluation metrics

For assessing the efficiency level of the proposed model, the following metrics employ several features. These metrics are: true positive (TP) or Recall (R), true negative (TN), false positive (FP) and false negative (FN).

True Positive Ratio (TPR) or Recall (R): TPR or Recall for a certain class, is the number of samples in the class that is correctly predicted, divided by the total number of samples in the class [4]. Formula 3 shows this relation.

$$\mathrm{TPR} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \tag{3}$$

False positive rate (FPR) is measured for estimating the quantity of the attack data that is identified as being normal data [4]. It is calculated as follows:

$$\mathrm{FPR} = \frac{FP}{FP + TN} \tag{4}$$

False negative rate (FNR) is measured for estimating the quantity of the normal data that is identified as being attack data [4]. It is calculated as follows:

$$\mathrm{FNR} = \frac{FN}{FN + TP} \tag{5}$$

Accuracy is represented in a percentage. It refers to the degree to which the instances are predicted correctly [4]. It is calculated as follows:

$$\mathrm{Accuracy} = \frac{(TP + FN)}{(TP + FP + TN + FN)} \tag{6}$$

Precision is represented by the ratio of the number of decisions that are considered correct. It is represented in the TP divided by the sum of FP and TP [4]. It is calculated as follows:

$$\mathrm{Precision} = \frac{TP}{TP + FP} \tag{7}$$

Results and discussions

After preparing the data set and normalizing them, proposed method implemented on data sets and in the proposed method, after changing the centre of the triangular fuzzy number, the maximum amount of merit is obtained and the desired space is searched by genetic algorithm. Moreover, the desired space with the genetic algorithm is determined and the desired number of features is obtained.

Experimental results on P-dataset

First we applied the proposed method to the P-dataset that we extracted from KDD. As aforementioned, in the extraction of the dataset, the scattering of normal class and attacks is maintained in the KDD dataset. The number of records in the training set is 86453 and the number of records in the test set is 35053. Table 9 shows the results and number of features obtained by applying the proposed method to the proposed data set. By changing the defined parameter, the degree of merit also changes. As shown in Table 8, the maximum value of merit is 94.88 and the 6 features were selected from the data set. The table below shows the values of p, m, and merits, extracted from the KDD data set and selecting the maximum merit value from the 42 features.

Table 8 Values of m, merits and p parameter of P- dataset

P	m	Merits
35	40.86	70.25
17	39.72	73.58
1.5	25.6	85.40
0.5	**14.6**	**94.88**
0.25	9.2	65.20
0.14	6.03	51.12

Table 9 Number of selected features by proposed method on P-dataset

Total Number of features	Number of selected features	P	Merits
42	6	0.5	94.88

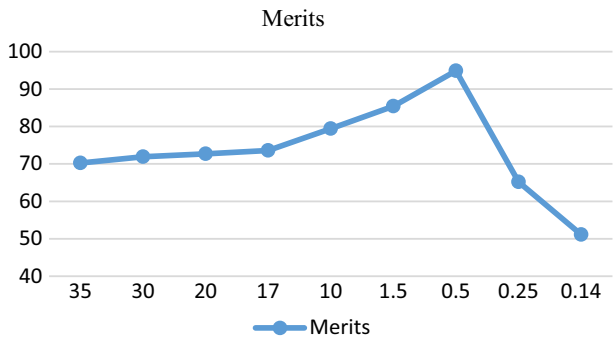


Fig. 8 The change in the merits concerning the change in the parameter p on the proposed data set

As the table above shows, from the data set with 42 features, the proposed method has selected 6 features. In the following sections, the number of selected features will be compared with other methods.

Consequently, Fig. 8 shows the change in the merits of the subsets concerning the change in the parameter p on the proposed data set. As it turns out, the maximum value of merit for p is 0.5.

At this stage of the research, the Random Forest, SVM, ANN, and KNN classifiers are used. We applied different classifications to the features selected by the proposed method and all the features of the above data set, and present the results in the table below to better evaluate the proposed method.

KNN

KNN is one of the simplest forms of machine learning algorithms mostly used for classification. It classifies the data point on how its neighbor is classified. KNN classifies the new data points based on the similarity measure of the earlier stored data points. KNN is Simple to implement and intuitive to understand. Can learn non-linear decision boundaries when used for classification and regression. No Training

Table 10 Results on P-dataset without feature selection algorithm

P-Dataset Without Feature Selection				
Binary Class				
Classifiers	**Accuracy**	**False positive**	**Recall**	**Precision**
Random Forest	86.38	0.249	0.864	0.86
ANN	88.15	0.27	0.882	0.87
KNN	89.22	0.26	0.89	0.871
SVM	88.29	0.256	0.882	0.871

Table 11 Results of proposed feature selection on P- dataset

P-Dataset with Proposed Feature Selection on Binary Class				
Classifiers	**Accuracy**	**False positive**	**Recall**	**Precision**
Random Forest	99.45	0.024	0.99	0.991
ANN	99.92	0.011	0.994	0.994
KNN	98.84	0.015	0.998	0.998
SVM	98.99	0.029	0.98	0.981

Time for classification/regression, The KNN algorithm has no explicit training step and all the work happens during prediction.

ANN

ANN is a complex adaptive system which can change its internal structure based on the information pass through it. It is achieved by adjusting the weight of connection. Each connection has a weight associated with it. A weight is a number that control the signal between two neurons. ANNs have the ability to learn and model non-linear and complex relationships, which is really important because in real-life, many of the relationships between inputs and outputs are non-linear as well as complex.

SVM

The advantages of support vector machines are: Effective in high dimensional spaces. SVM uses a high dimension space to find a hyper plane to perform binary classification. SVM approach is a classification technique based on Statistical Learning Theory (SLT). It is based on the idea of hyper plane classifier. The goal of SVM is to find a linear optimal hyper plane so that the margin of separation between the two classes is maximized. The SVM uses a portion of the data to train the system. It finds several support vectors that represent the training data. These support vectors will form a SVM model. The generalization ability of the SVM depends upon the value of the margin.

First, the classifiers to the data set are applied without applying any feature selection algorithms. Tables 10, 11, 12, 13, 14, 15, 16, 17, 18 show the results separately for each method applied. Consequently, on the proposed data set, classifications are applied without using any feature selection algorithms, and then the results are compared to when the proposed method and CFS algorithm are applied. The table is as follows:

Table 12 Number of selected features by proposed method on NSL dataset

Merits	P	Number of selected features	Total Number of features
91.58	1.5	7	42

Table 13 Results on NSL dataset without feature selection algorithm

NSL Dataset Without Feature Selection Binary Class				
Classifiers	**Accuracy**	**False positive**	**Recall**	**Precision**
Random Forest	80.21	0.815	0.802	0.801
ANN	83.45	0.789	0.831	0.832
KNN	82.98	0.799	0.83	0.829
SVM	83.35	0.759	0.833	0.83

Table 14 Results of applying proposed feature selection method on NSL dataset

NSL Dataset with proposed feature selection on Binary Class				
Classifiers	**Accuracy**	**False positive**	**Recall**	**Precision**
Random Forest	95.85	0.511	0.962	0.97
ANN	95.39	0.348	0.951	0.968
KNN	95.39	0.302	0.963	0.971
SVM	96.89	0.335	0.974	0.975

Table 15 The number of selected features by common method and proposed method

NSL Dataset	
Feature selection method	**Number of selected feature**
CFS + best first	10
Gain ratio and ranker	9
Chi Squared Eval + Ranker	10
Proposed Method	7

Table 16 The number of features selected by the proposed method on CICIDS dataset

Merits	P	Number of selected features	Total Number of features
99.77	0.5	3	80

Table 17 Results on CICIDS without any feature selection

Classifiers	CIC dataset without feature selection			
	Accuracy	**False positive**	**Recall**	**Precision**
Random Forest	70.46	0.701	0.705	0.694
ANN	70.04	0.752	0.7	0.74
KNN	70.39	0.727	0.704	0.677
SVM	70.08	0.749	0.701	0.791

Table 18 Results on CICIDS with proposed feature selection

Classifiers	CICIDS Dataset With proposed feature selection			
	Accuracy	False positive	Recall	Precision
Random Forest	70.18	0.672	0.702	0.716
ANN	69.48	0.695	0.699	0.71
KNN	69.95	0.68	0.7	0.727
SVM	69.48	0.695	0.699	0.71

Then we applied the classifiers to the selected features from the P- dataset using the proposed selection method. The results are shown in Table 11.

Comparing the above tables, it is clear that in all classifiers, the proposed method has been able to increase accuracy and performance by selecting useful features and reducing the false alarm rate compared to when all features are used.

Experimental results on NSL dataset

The results and number of features selected by the proposed method on the NSL dataset are shown in the following tables.

As the table above shows, from the NSL dataset with 42 features, the proposed method has selected 7 features. In the following sections, the number of selected features will be compared with other methods. Consequently, on NSL dataset, classifications are applied without using any feature selection algorithms, and then the results are compared to when the proposed method and CFS algorithm are applied. The table is as follows:

Then we applied the classifiers to the selected features from NSL dataset using the proposed selection method. The results are shown in Table 14.

Comparing the above tables, it is clear that in all classifiers, the proposed method has been able to increase accuracy and performance by selecting useful features. In the results table of the implementation of the proposed method, the false alarm rate has been significantly reduced compared to when all features are used.

We have applied the methods shown in the table below to the NSL dataset. Table 15 shows the number of selected features, by the various common methods and the proposed method, on the NSL dataset.

As shown in the table above, the number of features selected by the proposed method is less than the number of features selected by other methods.

Experimental results on CICIDS

The proposed method is now applied to the CICIDS data set with 43,293 training set records. The number of features selected by the proposed method on this dataset is 3 features and shown in Table 16.

As the table above shows, from the data set with 80 features, the proposed method has selected only 3 features. In the following sections, the number of selected features will be compared with other methods.

Figure 9 is a graph of the change of the two variables P and merit on the CICIDS dataset.

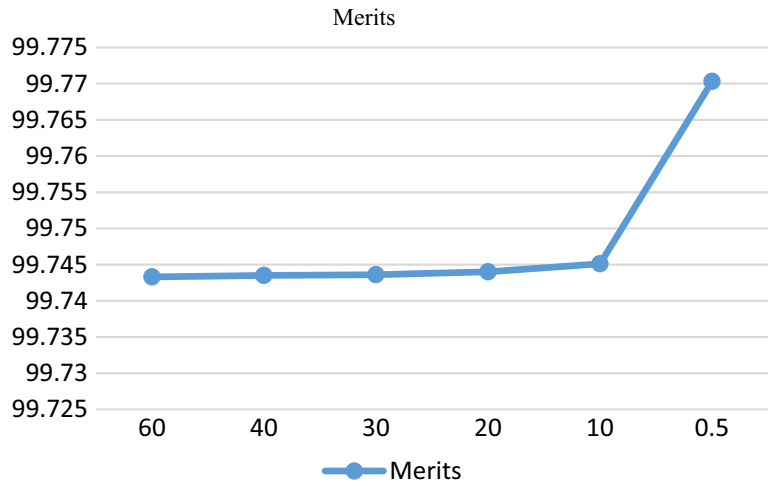


Fig. 9 Graph of the change of the two variables P and merit on the CICIDS dataset

Initially, classifiers were applied to the CICIDS dataset without any feature selection algorithms, so that we have a table of results for comparison with when the proposed methods and the CFS algorithm were applied.

Table 17 shows the results of applying classification to the CICIDS dataset without applying any feature selection algorithms.

Consequently, the classification is applied to the selected features of each set by the proposed method and is shown in Table 18.

Comparing the above tables, in the results table of the implementation of the proposed method, the false alarm rate has been reduced compared to when all features are used.

Comparison

Comparing the results of CFS feature selection with the proposed method. At this stage of the research, the proposed method is compared with the CFS feature selection method. The main aim is to obtain fewer features with higher detection rates. To do this, the features are first selected from the CICIDS dataset by the CFS feature selection algorithm, and then the number of features is compared with the proposed method.

Table 19 Number of selected feature by CFS and proposed method on CICIDS dataset

CICIDS Dataset	
Number of Selected Features by Proposed Method	**Number of Selected Features by CFS**
3	4

Table 20 Number of selected feature by CFS and proposed method on P- dataset

P- Dataset	
Number of Selected Features by Proposed Method	**Number of Selected Features by CFS**
6	10

Table 21 Results of CFS method and proposed method on CICIDS dataset

Classifiers	CICIDS Dataset			
	With CFS Method		With Proposed Method	
	Accuracy	False positive	Accuracy	False positive
Random Forest	68.25	0.7	70.18	0.672
ANN	69.48	0.73	69.48	0.695
KNN	70.74	0.711	69.95	0.68
SVM	69.48	0.725	69.48	0.695

Table 22 Results of CFS method and proposed method on proposed dataset

Classifiers	P- Dataset Binary Class							
	With CFS Method				With Proposed Method			
	Accuracy	False positive	Recall	Precision	Accuracy	False positive	Recall	Precision
Random Forest	87.29	0.458	87.29	88.35	95.95	0.011	95.95	96.89
ANN	96.01	0.053	96.01	97	99.92	0.029	99.93	99.99
KNN	96.32	0.089	96.32	97.45	98.84	0.025	98.85	99.87
SVM	95.87	0.091	95.88	96.88	98.99	0.014	98.99	99.99

Tables 19 and 20 show the number of features selected by each method. In the proposed method, 3 features have been selected from the CICIDS data set and 4 features have been selected from the CFS feature selection method.

In the proposed method, 6 features are selected from the selected data set and 10 features are selected from the CFS feature selection method.

Moreover, the classifications are applied to the data set without applying any feature selection algorithms. The results are then compared with the proposed method. Table 21 shows the results of applying the classifications on the CIC dataset first by selecting the CFS feature selection algorithm and then by proposed method.

As shown in the table above, the performance of the proposed method is higher than the performance of CFS method, and the false positive rate in the proposed method is lower due to the selection of more appropriate features. In addition, the proposed method by reducing the number of features, has reduced the complexity of classifiers and also reduced the dimensions.

We then compare the proposed method and the CFS method on the proposed data set. Table 22 shows the results of applying classifiers, first by applying the CFS method, and then by the proposed feature selection method on the proposed data set with two classes.

As shown in the table above, the performance of the proposed method is higher than the performance of CFS method, and the false positive rate in the proposed method is lower due to the selection of more appropriate features. In addition, the proposed method by reducing the number of features, has reduced the complexity of classifiers and also reduced the dimensions.

Figure 10 shows a comparison of the accuracy of the proposed method and the CFS method on the proposed data set.

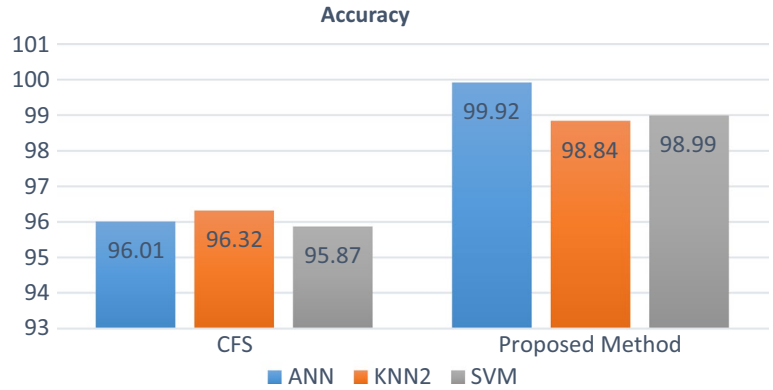


Fig. 10 Comparison proposed method and CFS method on P- dataset

Table 23 Number of selected features, selected by different methods and proposed method on different datasets

Data set	Total number of Features in dataset	Propose Method	CFS
CICIDS	80	3	4
NSL-KDD	42	7	12
P- dataset	42	6	10

Table 24 Selected features by CFS method and proposed method on P-dataset

No.	Features Selected by CFS Method	Features Selected Proposed Method
1	Protocol-type	Protocol-type
2	Service	Service
3	Flag	Flag
4	src-byte	src-byte
5	dst-byte	dst-byte
6	land	land
7	Srv-count	–
8	Count	–
9	Diff-srv-rate	–
10	Dst-host-same-src-port-rate	–

Consequently, Table 23 shows the number of features selected by the proposed method and other methods on different datasets.

As shown in Table 24, in all datasets, the number of features selected by the proposed method is less than the number of features selected by CFS method.

Figure 11 shows a comparison of the accuracy of the proposed method and the other methods mentioned in related work section, on the P- data set and NSL dataset (Table 25).

Table 25 Selected features by CFS method and proposed method on CICIDS-dataset

No.	Features Selected by CFS Method	Features Selected Proposed Method
1	Source port	Source port
2	Destination port	Destination port
3	Protocol	Protocol
4	Label	–

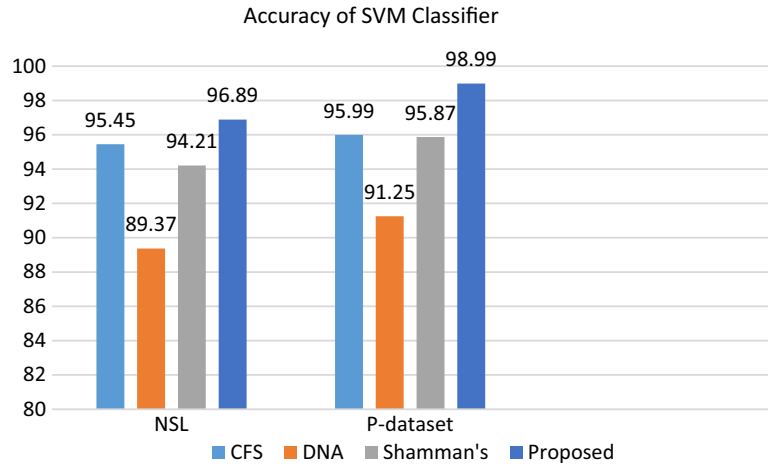


Fig. 11 The comparison results of the proposed method with other method

Conclusion

Generally, an intrusion detection system is full of big data with irrelevant features that make it difficult to classify the network and slow down the detection speed. By using a pre-processing step, the system performance can be improved. Reduce data dimensions, as a pre-processing step, eliminates the unnecessary features from the data set. This reduces the size of the data and thus speeds up the whole process. Reducing high dimensional sets includes feature extraction and feature selection, which are basic steps in building a detection system [19]. This research focuses on the feature selection stage in intrusion detection systems. As seen in the comparison tables, the performance of the proposed method is better than the correlation-based feature selection method in the mentioned data sets. In the proposed method, different values of the p were used and the most optimal value was selected. Moreover, fewer features have resulted in close or higher classification accuracy results. The main purpose of feature selection is to minimize the number of features with an average classification accuracy equal to or higher than the number of features. In this research, a method was proposed to improve the correlation-based feature selection method.

After applying different feature selections to the data set, CFS was chosen as the most efficient feature selection method. Consequently, the heuristic function algorithm was selected to evaluate the data subset and is also considered as a fuzzy set membership function. This estimates the number of data set features as a triangular fuzzy number such that the two bases of the triangle represent the most and the least number of data

set features. To control the number of features of the selected subset, a parameter p was defined, which is the ratio of the right base to the left base of a triangular fuzzy number. Additionally, a genetic algorithm was used to optimize and determine the search space of the subset of the features that have the highest value. The results show that the proposed method reduced the false alarm rates and increased the efficiency of intrusion detection systems. As shown in the comparisons, this method obtained a smaller number of features with a classification accuracy closer to or greater than the previous methods for different data sets with different classifications. In the proposed method, the number of features can be controlled by setting a fuzzy triangular number parameter.

Abbreviations

IDS	Intrusion detection system
SIDS	Signature-based IDS
AIDS	Anomaly-based IDS
CFS	Correlation feature selection
SVM	Support vector machine
SFS	Sequential Forward Selection
SBS	Sequential Backward Selection
NAN	NOT a Number
TP	True Positive
TN	True Negative
FP	False Positive
FN	False Negative
TPR	True Positive Rate
FPR	False Positive
FNR	False Negative Rate

Acknowledgements
Not applicable.

Author contributions
All mentioned authors contribute in the elaboration of the paper. All authors read and approved the final manuscript.

Funding
Not applicable.

Availability of data and materials
The datasets analysed during the current study are publicly available online in the cited sources, [http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html].

Declarations

Ethics approval and consent to participate
Not applicable.

Consent for publication
Not applicable.

Competing interests
The authors declare that they have no competing interests.

Received: 2 January 2022 Accepted: 29 January 2023
Published online: 03 March 2023

References

1. Pillai SG, Soon L, Haw S. YAGO for Web Information. Singapore: Springer; 2019.
2. Liu Y, et al. A hybrid feature selection algorithm combining information gain and genetic search for intrusion detection. J Phys Conf Ser. 2020;1601:3. https://doi.org/10.1088/1742-6596/1601/3/032048.
3. Sarvari S, Mohd Sani NF, Mohd Hanapi Z, Abdullah MT. An efficient anomaly intrusion detection method with feature selection and evolutionary neural network. IEEE Access. 2020;8:70651–63. https://doi.org/10.1109/ACCESS.2020.2986217.
4. Su T, Sun H, Zhu J, Wang S, Li Y. BAT: deep learning methods on network intrusion detection using NSL-KDD dataset. IEEE Access. 2020;8:29575–85. https://doi.org/10.1109/ACCESS.2020.2972627.

5. Jianjian D, Yang T, Feiyue Y. A novel intrusion detection system based on IABRBFSVM for wireless sensor networks. Procedia Comput Sci. 2018;131:1113–21. https://doi.org/10.1016/j.procs.2018.04.275.
6. Aghaeipoor F, Javidi MM. A hybrid fuzzy feature selection algorithm for high-dimensional regression problems: An mRMR-based framework. Expert Syst Appl. 2020;162:113859. https://doi.org/10.1016/j.eswa.2020.113859.
7. Al-Safi AHS, Hani ZIR, Abdul Zahra MM. Using a hybrid algorithm and feature selection for network anomaly intrusion detection. J Mech Eng Res Dev. 2021;44(4):253–62.
8. Chahira JM. Model for intrusion detection based on hybrid feature selection techniques. Int J Comput Appl Technol Res. 2020;09(03):115–24. https://doi.org/10.7753/ijcatr0903.1005.
9. Selvakumar K, et al. Intelligent temporal classification and fuzzy rough set-based feature selection algorithm for intrusion detection system in WSNs. Inf Sci (Ny). 2019;497:77–90. https://doi.org/10.1016/j.ins.2019.05.040.
10. GowdaKaregowda A, Jayaram MA, Manjunath A. Feature Subset Selection using Cascaded GA and CFS: A Filter Approach in Supervised Learning. Int J Comput Appl. 2011;23(2):1–10. https://doi.org/10.5120/2865-3711.
11. Gu Y, Li K, Guo Z, Wang Y. Semi-supervised k-means ddos detection method using hybrid feature selection algorithm. IEEE Access. 2019;7:64351–65. https://doi.org/10.1109/ACCESS.2019.2917532.
12. Cateni S, Colla V, Vannucci M. A fuzzy system for combining filter features selection methods. Int J Fuzzy Syst. 2017;19(4):1168–80. https://doi.org/10.1007/s40815-016-0208-7.
13. Alazzam H, Sharieh A, Sabri KE. A feature selection algorithm for intrusion detection system based on Pigeon Inspired Optimizer. Expert Syst Appl. 2020;148:113249. https://doi.org/10.1016/j.eswa.2020.113249.
14. Osanaiye O, Cai H, Choo KKR, Dehghantanha A, Xu Z, Dlodlo M. Ensemble-based multi-filter feature selection method for DDoS detection in cloud computing. Eurasip J Wirel Commun Netw. 2016;1:2016. https://doi.org/10.1186/s13638-016-0623-3.
15. Ambusaidi MA, He X, Nanda P, Tan Z. Building an intrusion detection system using a filter-based feature selection algorithm. IEEE Trans Comput. 2016;65(10):2986–98. https://doi.org/10.1109/TC.2016.2519914.
16. Dai J, Chen J. Feature selection via normative fuzzy information weight with application into tumor classification. Appl Soft Comput J. 2020;92:106299. https://doi.org/10.1016/j.asoc.2020.106299.
17. Indira P, Anuradha C, Murty PSRC. Fuzzy Based Feature Selection for Intrusion Detection System. Fuzzy. 2017;4(11):42–9.
18. Chen J, Mi J, Lin Y. A graph approach for fuzzy-rough feature selection. Fuzzy Sets Syst. 2020;391:96–116. https://doi.org/10.1016/j.fss.2019.07.014.
19. SamadiBonab M, Ghaffari A, SoleimanianGharehchopogh F, Alemi P. A wrapper-based feature selection for improving performance of intrusion detection systems. Int J Commun Syst. 2020;33(12):1–26. https://doi.org/10.1002/dac.4434.
20. Aggarwal P, Kumar S. Analysis of KDD dataset attributes - class wise for intrusion detection. Procedia - Procedia Comput Sci. 2015;57:842–51. https://doi.org/10.1016/j.procs.2015.07.490.
21. Meena G, Choudhary RR. "A review paper on IDS classification using KDD 99 and NSL KDD dataset in WEKA", 2017 Int. Conf Comput Commun Electron COMPTELIX. 2017;2017:553–8. https://doi.org/10.1109/COMPTELIX.2017.8004032.
22. Protić D. Review of KDD Cup '99, NSL-KDD and Kyoto 2006+ datasets. Vojnoteh Glas. 2018;66(3):580–96. https://doi.org/10.5937/vojtehg66-16670.
23. Panigrahi R, Borah S. A detailed analysis of CICIDS2017 dataset for designing Intrusion Detection Systems. Int J Eng Technol. 2018;7(324):479–82.
24. Ring M, Wunderlich S, Scheuring D, Landes D, Hotho A. A survey of network-based intrusion detection data sets. Comput Secur. 2019;86:147–67. https://doi.org/10.1016/j.cose.2019.06.005.
25. Bhati NS, Khari M. A new ensemble based approach for intrusion detection system using voting. J Intell Fuzzy Syst. 2020;42(2):969–79.
26. Bhati NS, Khari M. An Ensemble Model for Network Intrusion Detection Using AdaBoost, Random Forest and Logistic Regression. In: Applications of Artificial Intelligence and Machine Learning. Springer, Singapore; 2022. p. 777–789.
27. Bhati NS, Khari M. A new ensemble based approach for intrusion detection system using voting. J Intell Fuzzy Syst. 2022;42(2):969–79.
28. Bhati NS, Khari M. Comparative Analysis of Classification Based Intrusion Detection Techniques. In: 2021 5th International Conference on Information Systems and Computer Networks (ISCON) (pp. 1–6). IEEE. 2021.

Publisher's Note