ORIGINAL RESEARCH ARTICLE UHD JOURNAL OF SCIENCE AND TECHNOLOGY

Network Intrusion Detection using a Combination of Fuzzy Clustering and Ant Colony Algorithm

Yadgar Sirwan Abdulrahman

IT Department Kurdistan Technical Institute, Sulaymaniyah, Kurdistan Region, Iraq

ABSTRACT

As information technology grows, network security is a significant issue and challenge. The intrusion detection system (IDS) is known as the main component of a secure network. An IDS can be considered a set of tools to help identify and report abnormal activities in the network. In this study, we use data mining of a new framework using fuzzy tools and combine it with the ant colony optimization algorithm (ACOR) to overcome the shortcomings of the k-means clustering method and improve detection accuracy in IDSs. Introduced IDS. The ACOR algorithm is recognized as a fast and accurate meta-method for optimization problems. We combine the improved ACOR with the fuzzy c-means algorithm to achieve efficient clustering and intrusion detection. Our proposed hybrid algorithm is reviewed with the NSL-KDD dataset and the ISCX 2012 dataset using various criteria. For further evaluation, our method is compared to other tasks, and the results are compared show that the proposed algorithm has performed better in all cases.

Index Terms: Intrusion detection, Data mining, Fuzzy clustering, Ant colony

1. INTRODUCTION

Unusual behavior detection refers to a finding patterns process in a dataset that does not have the expected behavior. Network intrusion is also known as a set of unusual behaviors in the network. Mode detection provides essential information in a variety of applications that will improve network performance.

An intrusion detection system (IDS) is a device or software application that monitors the network to look for suspicious activity, threats, or policy breaching, and on encountering such activities, it alerts the security personnel. IDS monitors inbound as well as outbound network flow for abnormal behavior and then alert the admin or user that a network intrusion might be occurring. It performs the task by comparing signatures of a known malware against the system.

It monitors the user behavior, system processes, and system configurations for any unusual behavior. Security personnel is alerted on security breaches with data consisting of the addresses of the source, the target, and the type of attack.

The problem of intrusion detection is a complicated issue. A compromise must be made between detection accuracy, detection speed, intrinsic network dynamics, and high data volume for processing, and the methods used must be able to distinguish between state and abnormal behaviors. Be normal behaviors in the network. The IDS's primary purpose can be considered network display for any mode such as DoS, U2R, R2L, some of which are listed in Table 1 [1].

Access this article online

DOI: 10.21928/uhdjst.v5n2y2021.pp11-19

E-ISSN: 2521-4217

P-ISSN: 2521-4209

Corresponding author's e-mail: Yadgar Sirwan Abdulrahman, *IT Department Kurdistan Technical Institute, Sulaymaniyah, Kurdistan Region, Iraq.* E-mail: yadgar.abdulrahman@kti.edu.krd

Receied: 01-04-2021 Accepted: 07-07-2021 Published: 16-07-2021

TABLE 1: The network attack types classification.

Attack class	Attack name	Attack description
Probing	Probe	An attacker performs port scanning and monitoring activities to gather information or find vulnerabilities in a network
DoS	Denial of Services	An attacker fills a busy network resource (such as memory or bandwidth) with repeated requests, causing network resources to overflow and users' requests not to be answered
User to root	U2R	An attacker accesses a regular account and searches for a vulnerability to gain unauthorized root access to the system
R2L	Remote to Local	An attacker gains access to a system through a remote network and attempts to gain unauthorized local access through a remote system

Among these modes, distributed denial of service (DDoS) is one of the security threats associated with computer networks, especially the internet, which targets access to network resources, and the purpose of this mode is to disrupt network service. One of the most dangerous and most recent situations on the internet is not to disrupt the service, but to force the network and server to be unable to provide regular service to target network bandwidth. It is done with a victim who drowns the victim's network or processing capacity in information packets and prevents users and customers from accessing the service. One of the most common and significant threats on the internet today is a denial of service by interfering with configuration information. Routers and IP source fraud occurs, leading to reduced network performance.

For years, experts have warned about the poor security of internet-connected devices and equipment, and poor security has made them vulnerable configuration of equipment and the heterogeneity of operating systems such devices a very convenient yet easy target for attackers. One of the main exploits of hackers and destroyers of these devices and equipment is to capture them to execute the distributed model. During this state, an army of these hacked devices bombards it by sending simultaneous requests to the victim's server, which is called this type of hacked equipment with a net. Receiving a request from thousands and sometimes tens of thousands of devices with different IP addresses at the same time will eventually lead to slowing down or even stopping the server service to users. When a DDOS attack occurs, the first step is to determine what layer of the open systems interconnection model the attack is on; the mode is usually on layers 3 and 4 of the network and the scope of an attack depends on features such as volume and the number of packets sent per second. Layers 3 and 4 are very difficult to control. Dispose of it. The issue of identifying and providing a suitable solution is one of the biggest challenges facing network security professionals.

The methods of diagnosis and prevention that have been presented so far have either not been effective or have not been adequately responded to by attackers with increasing level of knowledge, and most of the detection is in the form of statistical methods and monitoring and control of network traffic. In the case of this type of attack and high traffic, if two attacks occur with different traffics combined in the network, this type of method will no longer be a good answer us, and because the attack speed and the network traffic volume is very high in a short time, it should be possible to detect and deal with the attack as soon as possible, in other words, the system notices the denial of service when the network attack increases and affects traffic volume and it is no longer possible to deal with, and when it detects an attack on the traffic volume network, it will be more careful with the defense layers on the internet, before the defense layer service can react. For example, if we can detect the mode in network routers and these hand-held routers can make a proper diagnosis, the probability of service denial mode is reduced and further risks are avoided. In this problem, the attack detection importance in the lower layers of the network, such as the network layer and the data link layer, is seen more, and in the higher layers, such as the application layer, it requires data packets to be examined, which will take many of our resources and time.

A computer network attack detection system is one of the most important parts to prevent illegal intrusions in the network. Detecting and detecting intrusion can reduce the misuse of individuals' personal information as well as prevent financial risks for users and service companies. Various algorithms and methods for this classification have been proposed in the previous works, each of which has its advantages and disadvantages. In this study, we present our proposed framework along with the improvements in the algorithms used to distinguish DoS/DDoS mode from normal network mode in the ISCXIDS2012 dataset, which are fully described in Section 3. This study tries to provide a suitable framework for attack detection using fuzzy clustering and feature selection.

This paper's structure is as follows: Section 2, we mentioned an overview of the related work. Section 3 describes the

problem and the proposed method. Section 4 will present the experiments and results and finally, the conclusion is stated in Section 5.

2. RELATED WORK

Much work has been done in the field of intrusion detection in computer networks, and in this section, we will briefly mention some of these studies.

In Chitrakar and Chuanhe [2], to solve the problem of high data volume requirements in works that use k-means or k-means clustering and Forward Neural Network, a combination of support vector machine (SVM) with k-means clustering The Kyoto 2006+ dataset is used in this work and the simulation results show that the use of SVM in any volume of data has higher classification accuracy. To evaluate the operations performed in this work, sensitivity criteria. And False Alarm has been used in Chitrakar and Chuanhe [2], to detect network intrusion, the combined method of K-means clustering with Naive Bayes classification with the same working criteria and the same dataset (Kyoto 2006+ dataset) has been used. The results of this work were somewhat weaker than the work done in Chitrakar and Chuanhe [2]. In Saifullah [3], they propose a defense mechanism to detect an attack using a distributed algorithm that runs a moderate load valve in the opposite direction of the router. The valve has a medium load because the traffic intended for the server is controlled [3]. The operation (increase or decrease) is performed using a perforated bucket in the router based on the number of connected users, who are directly or indirectly connected to the router. At the beginning of the algorithm, the remaining capacity is underestimated by the router. The remaining initialization capacity is the minimum or normal value at the beginning of the algorithm. The speed is updated (increase or decrease), sends to small routers based on server feedback, and finally multiplies all routers in descending order. The convergence of the whole server is loaded with an acceptable capacity range.

In Syarif *et al.* [4] there are three objectives: (1) effective feature selection and dimension reduction, –(2) a strong algorithm selection in the classification field, and –(3) unconventional detection, using clustering algorithms based on segmentation; to achieve the first goal genetic algorithm and particle swarm optimization (PSO) are used. For the classification operation, the nearest neighbor classification method has been used, and finally, by comparing different types of clustering methods, the expectation maximization method has had the best performance. The results for false-positive and accuracy criteria have been investigated using four classification methods, the highest accuracy being related to the decision tree. In Revathi and Malathi [5], just like [4] methods of optimizing collective intelligence have been used. This work uses simplified collective intelligence, which is a kind of simplified and improved PSO algorithm (SSO) along with the Random Forest method on the KDDcup 1999 dataset. In Singh and Singh [6] an attack detection method is presented in MANET based on the ACO algorithm. In this work, after the intrusion detection by the ant colony algorithm, the genetic algorithm has been used to retrieve the network, in which the number of recovered nodes has been investigated for the number of 10–80 replications and the probability of mutation between 0.2 and 0.4.

In both papers Cheng *et al.* [7] and Xia *et al.* [8] IP Address Interaction (IAI), the algorithm is proposed due to sudden traffic changes, the interaction between addresses, the asymmetry between many addresses, source distribution, and focus of targeted goals, IAI algorithm is designed to describe the network stream validity important features. The SVM classifier, which is sorted by IAI interval with normal attack current, is used to classify the current network stream validity and identify DDOS mode. The method is defined in real-time attack flow detection as well as attacker assessment power based on fuzzy reasoning. The process consists of two steps: (a) Statistical analysis of network traffic interval and (b) DDOS attack identification and evaluation based on intelligent fuzzy reasoning mechanism.

In Kamarudin *et al.* [9], the feature selection was performed using the Random forest genetic algorithm without an observer and a subset of the total features for each of the two datasets DARPA1999 and ISCX2012 was obtained. The use of random forest classification has been performed. In Vargas-Munoz *et al.* [10], an IDS system based on the Bayesian network is also presented. Eight features, the ICCX2012 feature set is used in the classification section.[11] Feature selection is based on Entropy. In their work, the raw features of the ISCX2012 dataset and five features based on Entropy values are considered. They classify MLP neural network, RNN (Alternative decision tree) ADT has been used.

3. PROPOSED METHOD

In this study, we present our proposed framework along with the improvements in the algorithms used to distinguish DoS/DDoS attacks in normal network mode in the ISCXIDS2012

dataset. All the steps to achieve attack detection using data mining and artificial intelligence can be summarized in the following sections.

- Data preprocessing
- Combined clustering using ant colony optimization and fuzzy clustering
- Classification and review of quantitative criteria.

After performing these steps, the constructed approach can be used. The most important feature of this study is to present a model for DDoS mode detection that improves the accuracy of the combination of the fuzzy c-means algorithm and the ant colony optimization and improves this algorithm in clustering, which improves the detection accuracy. In the following, the main steps of the proposed method are discussed.

3.1. Data Collection and Preprocessing

The study uses two datasets, NSL-KDD and ISCX datasets, which are described in the following. The number of training instances in each attack class is shown in both KDD Train (KDD cnp99) and NSL-KDD(KDD Train+) datasets in Table 2. The NSL-KDD dataset also includes two training datasets. KDD Train+ And KDD Train+_20% of which KDD Train+_20% is the improved version of KDD Train+. Test examples for the NSL-KDD dataset also include two sets Test KDD+ and KDD Test-21. KDD Test-21 has more difficulty distinguishing samples than KDD Test+ As can be seen, most of the samples removed from KDD CUP are in DOS mode with a removal rate of 82.98%. NSL-KDD is obtained by removing approximately 43.97% of the samples in the KUPD CUP99 dataset. In total, the NSL-KDD dataset has 25,192 samples and 43 features. (This dataset is comprised four sub-datasets: KDDTest+, KDDTest-21, KDDTrain+, and KDDTrain+_20Percent, although KDDTest-21 and KDDTrain+_20Percent are subsets of the KDDTrain+ and KDDTest+. From now on, KDDTrain+ will be referred to as train and KDDTest+ will be referred to as a test. The KDDTest-21 is a subset of the test, without the most difficult traffic records (Score of 21), and the KDDTrain+_20Percent is a subset of the train, whose record count makes up 20% of the entire train dataset. That being said, the traffic records that exist in the KDDTest-21 and KDDTrain+_20Percent are already in test and train, respectively, and are not new records held out of either dataset.)

TABLE 2: NSL-KDD data information

Class	KDDCUP'99 (KDD Train)	NSL-KDD (KDDTrain+)	% Reduction	% in NSL KDD
NORMAL	972,781	67,343	93.07	53.46
DOS	3,883,370	45,927	98.82	36.46
PROBE	41,102	11,656	71.64	9.25
U2R	52	52	0	0.04
R2L	1126	995	11.63	0.79
TOTAL	4,898,431	125,973	97.43	

As mentioned earlier, this article also uses the ISCX dataset [12]. The structure of the network used to generate this dataset is shown in Fig. 1. As shown in Fig. 1, the test structure consists of four separate LANs, and the fifth LAN consists of servers that provide web, email, DNS, and NAT services. All links are set on 10M bits/s.

The data began on Friday, June 11, 2020, and lasted exactly 7 days. This article examines the DDoS mode detection performed on Tuesday compared to the normal network mode (no attack) performed on Friday. Given that this dataset is available in pcap format, we use CICFlowMeter software. Together with winpcap software, we have used to extract 24 features. Then specify the data path in pcap format and the CSV data storage path to obtain user data in the preprocessing section.

4. CLUSTERING

4.1. Ant Colony Optimization Algorithm (ACOR) Algorithm

We discuss the design process of the ant colony optimization algorithm of the continuous domain for solving unconstrained optimization problems and constrained optimization problems based on the position distribution model of ant colony foraging [13].

Assuming the whole ant colony consists of m groups of substructure, each group contains n of ants. As shown in the following equation:

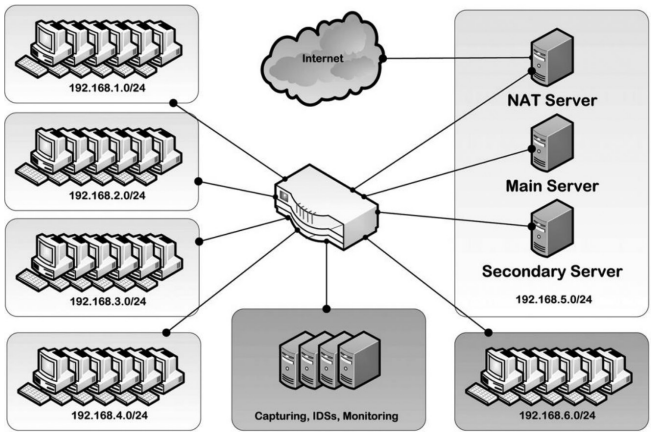

Fig. 1. Data generation network structure [12].

$$\begin{bmatrix} x_1 & x_2 & \dots & x_n \\ ant_{11} & ant_{12} & \dots & ant_{1n} \\ ant_{21} & ant_{22} & \dots & ant_{2n} \\ \vdots & \vdots & & \vdots \\ ant_{m1} & ant_{m2} & \dots & ant_{mn} \end{bmatrix} \quad (1)$$

the position ant_{ij} corresponding to the value x_j of the variable for j-ant in any sub colony i, the sub colony I of all the ants in the sequence of $\{ant_{i1}, ant_{i2},\dots, ant_{in}\}$ represents a solution of the optimization problem.

In the position distribution model of ant colony foraging, each ant releases pheromone according to the quality of a food source of their position; pheromones are dispersed in the entire space, with increasing distance of the source and the concentration decreasing. Therefore, we need to choose a probability density function as the distribution model of ant pheromone in the optimization algorithm of continuous domains. The Gaussian function is a common probability density function; we assume ants of the ant colony release pheromone externally on the function. At this point, j ant in any sub colony ant i corresponding to pheromone distribution model $\tau_{ij}(x)$ can be expressed as

$$\tau_{ij}(x) = \frac{1}{\sqrt{2\pi}\sigma_j} e^{-((x-\mu_{ij})^2/2\sigma_j^2)},$$
$$\sigma_j = \frac{(u_j - l_j)}{\Psi(1+1n(n))}, \quad (2)$$

where μ_{ij} is the position ant_{ij} of ant j in the sub colony of ants i, namely, the distribution center, $\sigma j(\sigma j>0)$ means the width of the distribution function, u_j is the maximum allowable value of the variable x_j, l_j is the minimum allowable value of the variable x_j, n is the dimension of solution for the optimization problem, ψ ($\psi > 0$) is a parameter, and σj is used to adjust the size.

Before updating the position of the ant colony, we need to choose a group as a parent from m sub colony. First, we use formula (3) to calculate each group of sub colony corresponding to the assessed value of the solution. Consider the following:

$$eval_i = \frac{1}{(1+e^{f(ant_{i1}, ant_{i2},\dots ant_{in})/T})}, \quad (3)$$

Where $f(ant_{i1}, ant_{i2},\dots, ant_{in})$ is the assessment value of the sub colony ant i; T (T>0) is the adjustment coefficient used to adjust the pressure of selection.

After the assessment value for each group of sub colony is obtained, we calculate the selected probability for each group of sub colony according to

$$pi = \frac{eval_i}{\sum_{j=1}^{m} eval_j}. \quad (4)$$

Finally, we select parent colony c according to formula (5)

$$\begin{cases} \arg\max(eval_i), & q \le q_0, \\ \quad i=1,2,\dots m & \\ C & q>q_0, \end{cases} \quad (5)$$

Where $(0\le q0\le 1)$ is a given parameter, q is a random variable is distributed in [0,1] uniformly. C is a random variable that is generated according to formula (5).

After getting the parent ant colony c, the ant pheromone distribution model function $\tau_{cj}(x)$ in the ant colony corresponding to random number generator for sampling, the k groups of children colony are generated. Then, according to the size of assessment value for each group of sub colony, we select the large assessment value of m group from (m+k) group of sub colony to achieve a position of ant colony update.

4.2. Basic Fuzzy C-means

In this section, the basic fuzzy C-means algorithm [14] will be briefly introduced. The objective function of this algorithm is defined as bellow:

$$J = \sum_{i=0}^{N} \sum_{j=1}^{C} \mu_{ij}^{m} d^2 (xi - vj) \quad (6)$$

The μ_{ij} determines the degree to which the i_th sample belongs to the center of the cluster j, and m determines the fuzzy degree. Here d^2 $(xi\text{-}vj)$ is the non-euclidean distance equal to $(xi\text{-}vj)^2$. As x_i is the i_th sample and v_j is the center of the j_th cluster. For this objective function, there are constraints $0 < \sum_{i=1}^{N} \mu_{ij} < N$ and $\mu_{ij} \varepsilon$ [0-1]. The values of the variables i and j are in the range of $1\le i\le$N and $1\le j\le$C.

Based on the objective function introduced in equation (6), equations for improving the centers and functions of the affiliation will be as follows:

$$\mu_{ij} = 1 / \sum_{l=1}^{C} \left(\frac{d^2(x_i - v_j)}{d^2(x_i - v_l)}\right)^{\frac{1}{(m-1)}}$$
$$v_j = \left(\sum_{i=1}^{N} \mu_{ij}^{m} x_i\right) \Big/ \sum_{i=1}^{N} \mu_{ij}^{m} \quad (7)$$

$$v_j = \left(\sum_{i=1}^{N} \mu_{ij}^{m} x_i\right) \Big/ \sum_{i=1}^{N} \mu_{ij}^{m} \quad (8)$$

4.3. ACOR Improvement

As mentioned before, the X matrix is the original data matrix with dimensions N*D, where N is the number of observations and D is the number of attributes. The output of the first layer is the X Matrix with dimensions N*R where R≤D is the selected properties. The second layer is responsible for clustering data in K clusters. The proposed method in this layer is to combine the fuzzy c-means clustering algorithm with the ACOR optimization algorithm. Furthermore, in the ACOR [13] algorithm and fuzzy c-means algorithm [14], changes have been applied to improve the performance of the proposed method, which is stated.

In the ACOR algorithm, to determine the weights, an exponential relationship is used, which causes a high difference for the weight of the answers with high and low rank. In other words, high-ranking answers are very important in each iteration, which reduces the breadth of finding the answer and catching the algorithm in the local minimum. For this reason, in this study, we have proposed the following weighting function to determine the weights. In this case, as the answer rank increases, the number of weight increases, but these changes occur much smoother than the case suggested in Socha and Dorigo [13]. The proposed function for means and standard deviation (σ = kq) (different) is shown in Fig. 2. As can be seen, it is an S-shape function. The cumulative distribution function (CDF) of the normal, or Gaussian, distribution with standard deviation σ and mean μ. And then, which is τ_{ij} (x) will be calculated as described below.

Instead of using formula 2 in, and from kq we can rewrite 2nd formula as formula 9. Then for a smoother objective function due to arithmetic estimation mentioned below. As in reference 15, we can compute erf(x). Its not a new formula. Its using an approximation function of the main objective function of ACOR algorithm. That gets us a smoother result in practice Now we can use this CDF estimation for formula (2) which is a PDF or Gaussian function:

$$Ae^{-B\left(\frac{x-\mu}{\sigma}\right)^2} \approx \frac{1}{2}\left(1+\operatorname{erf}\left(\frac{x-\mu}{\sigma\sqrt{2}}\right)\right) \quad (9)$$

Here, if we consider A=1/√2πσ and B=1/2, then from using formula (9) and (2) we have:,

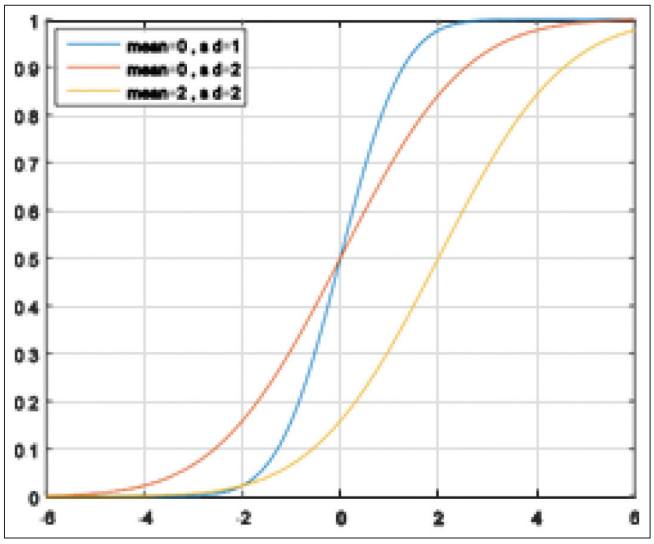

Fig. 2. Weighting functions for means and different standard deviation.

$$\tau_{ij} = \frac{1}{2} + erf\left(\frac{x-\mu_{ij}}{\sigma\sqrt{2}}\right) \quad (10\text{-a})$$

As we know then we can replace (σ = kq) to and reach

$$\tau_{ij} = \frac{1}{2} + erf\left(\frac{x-\mu_{ij}}{qk\sqrt{2}}\right) \quad (10\text{-b})$$

And we can calculate erf(x) as we have this in [15],

$$erf(x) = \frac{2}{\sqrt{\neq}}\int_0^x e^{-t^2} dt \quad (11)$$

The combination of ACOR and fuzzy c-means algorithms can be achieved in three ways. As follows:

- In the first case, the selection of cluster centers can be done by the ACOR algorithm in such a way that this algorithm calculates cluster centers based on the defined objective function; these centers are then applied to the fuzzy c-means clustering algorithm as the initial mean
- In the second case, after random generation of the first states, the fuzzy c-means algorithm of events is executed and according to the desired population in successive iterations, the centers of the clusters are obtained and then these centers are given to the ACOR algorithm as the initial population and this algorithm is based on the function. The defined goal performs the clustering operation. This will usually not produce the desired result
- In the third case, the ACOR algorithm starts clustering the data, with the difference that at the same time the

fuzzy c-means algorithm is applied and improves the location of the best available country. This method requires more time than the previous two modes.

In this paper, we used the first case, as discussed above. We use the ACOR algorithm to cluster center selection; these centers are then applied to the fuzzy c-means clustering algorithm as the initial mean. And as an improvement of ACOR weighting function, instead of formula (2) in [13] which is a PDF function, we used a smoother arithmetic estimation function or CDF function from formula (9). Then the weighting function will become as formula (10-a), by using (σ = kq), we'll get to the formula (10-b). And to calculate erf(x) we can use the formula (11). Now we have a new smoother weighting function as mentioned in formula (10-b), which can be easily calculated.

5. SIMULATION RESULTS

In this study, we used python with Pycharm IDE for implementation, and an HP laptop with 8gigabyte RAM, core i7 6600u CPU, windows 10. The first case is used and the results obtained with PSO-k-means, ICA-k-means, k-means, Fuzzy c-means ++, and DBSCAN methods and methods proposed in Kumar and Kumar [16], Kaur *et al.* [17] and Soheily-Khah *et al.* [18]. It should be noted that when using optimization methods for clustering, the defined objective function must be a function appropriate to the clustering problem. Defined for clustering, in which the objective function of the k-means problem is used, which is as follows.

$$Cost(c) = \sum_{i=1}^{N} \min\{\|xi - Cj\|\} \qquad j = 1,2,\ldots,K$$

Where the selected distance is between the sample and the center of the cluster.

To present and compare the results in this section, the training curves of hybrid algorithms along with the correct data clustering rate (D.R), Accuracy, and False Alarm have been calculated.

$$Detection\ Rate = TP/TP + FN \tag{12}$$

$$Accuracy = TP + TN/TP + TN + FP + FN \tag{13}$$

$$False\ Alarm = FP/FP + TN \tag{14}$$

The parameters used for the ACO algorithm are given in Table 3. Furthermore, the parameters of the Fuzzy c-means algorithm are obtained by the colonial competition optimization algorithm with the Davis Boldin clustering cost function.

Fig. 3 shows the training curve for the proposed method. As it turns out, the proposed method has the best performance in minimizing the cost function.

Table 4 shows all the parameters of relationships 2–3 for all methods using the ISCX dataset along with a comparison of the methods Kumar and Kumar [16] and Soheily-Khah *et al.* [18] and MBGWO [19] with the proposed method. In

TABLE 3: Parameters used in ACO Iteration algorithm (MAX)

Iteration(MAX(	Population	#of antes	α	k	Mu
1000	100	15	1	2	0.1

TABLE 4: Clustering evaluation indicators for different methods using all 24 attributes in the ISCX dataset

Clustering methods	Accuracy	False alarm rate	Detection rate
Proposed method	99.93	0.04	99.55%
ICA-Fuzzy c-means	97 %	0.06	96.2%
PSO k-means	94.6%	0.06	94.1%
Fuzzy c-means++	91.4%	0.08	91%
k-means	67.45%	0.12	67%
DBSCAN	68.67%	0.12	68%
(Kumar, 2013) method	95.2%	0.07	94.5%
(Soheily-Khah, 2018) method	99.91%	0.05	99.51%
MBGWO (M. Alzubi1 2019)	99.22	0.0064	99.10

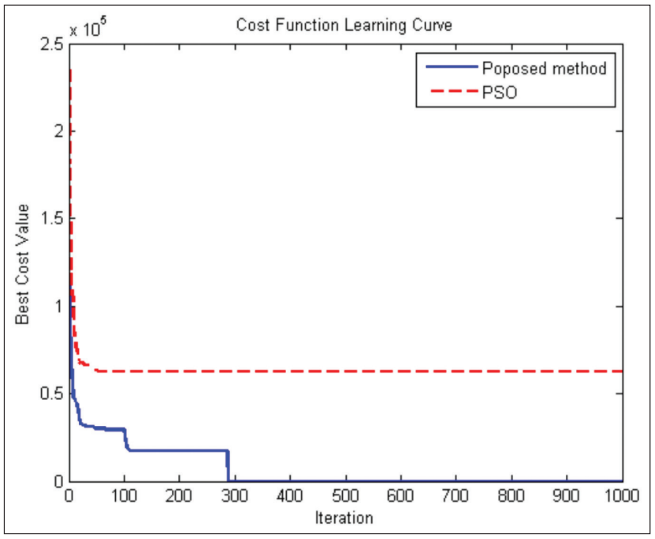

Fig. 3. Training curves for the proposed method and single-particle swarm optimization algorithm.

TABLE 5: Clustering evaluation indicators for different methods in the NSL-KDD dataset

Clustering methods	Accuracy	False alarm rate	Detection rate
Proposed method	86.98%	0.14	78%
ICA-Fuzzy c-means	84%	0.18	74.52%
PSO k-means	81.16%	0.212	70.1%
Fuzzy c-means++	74.14%	0.219	69.21%
k-means	63.35%	0.31	57%
DBSCAN	65.37%	0.31	59%
(Kaur, 2017) method	82.1%	0.211	72.57%

calculating the criteria, the total number of performances is Nt=10.

As shown in Table 4, the proposed method in clustering has shown the best performance in accuracy and detection rate indicators.

The following results are reviewed for the NSL-KDD dataset. In this dataset, the number of clusters is equal to four types of attacks and normal mode (a total of five clusters) has been considered. For this dataset, all the available features have been considered in the clustering section and no further reduction has been made. Table 5 shows all the parameters of relationships 2–3 for all methods using the NSL-KDD dataset with a comparison of the method [17] with the proposed method.

As can be seen in this case, the combined algorithm Fuzzy c-means and ACO for continuous domains give better results for three metrics met with a slight detection of the condition.

Tables 6 and 7 show the comparison of proposed algorithm to some recent deep network IDSs. Both NSL-KDD and ISCX dataset are included in the study. The algorithms are described in the table.

In Table 6, there are BGRU [20], long short-term memory (LSTM) [21], and ANN [22] accuracy calculated on NSL-KDD dataset. And two deep algorithms have better performance than proposed algorithm.

In Table 6, there are Deep CNN [23], LSTM [24], and predicting future tokens [25] accuracy calculated on NSL-KDD dataset. And two deep algorithms have better performance than proposed algorithm.

Experiments on ISCX dataset shows that the proposed method does well and is a bit better than the three deep algorithms compared.

TABLE 6: Clustering evaluation indicators for different methods in the NSL-KDD dataset

Algorithm	Accuracy	description
Jiang *et al*. (2019)	98.94	Training multiple long short term memory nets (one hidden layer) for different features extracted
Xu *et al*. (2018)	99.24	5-class classification GRU and Bidirectional GRU (BGRU) nets. Model has one layer with 128 GRU nodes, 3 feed-forward layers with 48 nodes BGRU gives best results with fast convergence
Vinayakumar *et al*. (2019)	78.5	ANN(shallow neural network) has five hidden layers with 1024, 768, 512, 256, and 128 nodes. ReLU activation
Proposed method	86.98	

TABLE 7: Clustering evaluation indicators for different methods in the ISCX dataset

Algorithm	Accuracy	Description
Zeng *et al*. (2019)	99.85	5-class classification Deep CNN (convolutional nural network): 2 1D convolutional layers, 1 fully connected layers
Chilamkurti (2018a)	99.91	Binary classification 30 embedding layers, 10 LSTM (long short term memory) layers, and sigmoid output layer
Radford *et al*. (2018)	97.01	anomaly detection by predicting future tokens (unsupervised) Token embedding layer
Proposed method	99.93	

6. CONCLUSION

In this study, the proposed framework for identifying DDoS is discussed. At first, the necessary preprocessing was performed on the data, and then the state diagnosis was performed using a combined fuzzy clustering algorithm and compared to some other methods. The results showed that, the proposed method of this study, has shown better performance in the quantitative criteria considered at most. In comparison to both classic methods and deep methods for intrusion detection, our proposed method is doing well on ISCX dataset, but for NSL-KDD dataset deep algorithms shown better performance.

As future work we should try to improve our proposed method to discover attacks, or if it will be possible, extend it to a deep method.

REFERENCES

[1] M. Mazini, B. Shirazi and I. Mahdavi. "Anomaly network-based intrusion detection system using a reliable hybrid artificial bee colony and AdaBoost algorithms". *Journal of King Saud University-Computer and Information Sciences*, vol. 31, no. 4, pp. 541-553, 2019.

[2] R. Chitrakar and H. Chuanhe. "*Anomaly Based Intrusion Detection Using Hybrid Learning Approach of Combining K-Medoids Clustering and Naïve Bayes classification*". IEEE, United States, 2012.

[3] A. Saifullah. "*Defending Against Distributed Denial-of-Service Attacks With Weight-Fair Router Throttling*", 2009. Available from: https://www.openscholarship.wustl.edu/cse_researchhttps://www.openscholarship.wustl.edu/cse_research/23. [Last accessed on 2021 May 10].

[4] I. Syarif, A. Prügel-Bennett and G. Wills. "Data mining approaches for network intrusion detection: From dimensionality reduction to misuse and anomaly detection". *Journal of Information Technology Review*, vol. 3, no.2, pp. 70-83, 2012.

[5] S. Revathi and A. Malathi. "Data Preprocessing for Intrusion Detection System using Swarm Intelligence Techniques". *International Journal of Computer Applications*, vol. 75, no. 6, pp. 22-27, 2013.

[6] K. Singh and K. Singh. "Intrusion detection and recovery of MANET by using ACO algorithm and genetic algorithm". *Advances in Intelligent Systems and Computing*, vol. 638, pp. 97-109, 2018.

[7] J. Cheng, C. Zhang, X. Tang, V. S. Sheng, Z. Dong and J. Li. "Adaptive DDoS attack detection method based on multiple-kernel learning". *Security and Communication Networks*, vol. 2018, p. 5198685, 2018.

[8] Z. Xia, S. Lu, J. Li and J. Tang. "Enhancing DDoS flood attack detection via intelligent fuzzy logic". *Informatica*, vol. 34, no. 4, pp. 497-507, 2010. Available from: http://www.informatica.si/index.php/informatica/article/view/323. [Last accessed on 2021 May 11].

[9] M. H. Kamarudin, C. Maple, T. Watson and N. S. Safa. "A new unified intrusion anomaly detection in identifying unseen web attacks". *Security and Communication Networks*, vol. 2017, p. 2539034, 2017.

[10] M. J. Vargas-Munoz, R. Martinez-Pelaez, P. Velarde-Alvarado, E. Moreno-Garcia, D. L. Torres-Roman and J. J. Ceballos-Mejia. "Classification of network anomalies in flow level network traffic using Bayesian networks". In: *2018 28th International Conference on Electronics, Communications and Computers, CONIELECOMP 2018*, vol. 2018, pp. 238-243, 2018.

[11] A. Koay, A. Chen, I. Welch and W. K. G. Seah. "A new multi classifier system using entropy-based features in DDoS attack detection". In: *International Conference on Information Networking*, vol. 2018, pp. 162-167, 2018.

[12] A. Shiravi, H. Shiravi, M. Tavallaee and A. A. Ghorbani. "Toward developing a systematic approach to generate benchmark datasets for intrusion detection". *Computers and Security*, vol. 31, no. 3, pp. 357-374, 2012.

[13] K. Socha and M. Dorigo. "Ant colony optimization for continuous domains". *European Journal of Operational Research*, vol. 185, no. 3, pp. 1155-1173, 2008.

[14] J. C. Bezdek. "*Pattern Recognition with Fuzzy Objective Function Algorithms*". Springer, United States, 1981.

[15] L. C. Andrews. "*Special Functions of Mathematics for Engineers*", 2021. Available from: https://www.books.google.nl/books?id=2caqsf-rebgc and pg=pa110 and redir_esc=y#v=onepage&q&f=false. [Last accessed on 2021 Jun 02].

[16] G. Kumar and K. Kumar. "Design of an evolutionary approach for intrusion detection". *The Scientific World Journal*, vol. 2013, p. 962185, 2013.

[17] A. Kaur, S. K. Pal and A. P. Singh. "Hybridization of K-means and firefly algorithm for intrusion detection system". *International Journal of Systems Assurance Engineering and Management*, vol. 9, no. 4, pp. 901-910, 2018.

[18] S. Soheily-Khah, P. F. Marteau and N. Bechet. "Intrusion detection in network systems through hybrid supervised and unsupervised machine learning process: A case study on the iscx dataset". In: *Proceedings-2018 1st International Conference on Data Intelligence and Security*, *ICDIS 2018*, pp. 219-226, 2018.

[19] Q. M. Alzubi, M. Anbar, Z. N. M. Alqattan, M. A. Al-Betar and R. Abdullah. "Intrusion detection system based on a modified binary grey wolf optimisation". *Neural Computing and Applications*, vol. 32, no. 10, pp. 6125-6137, 2020.

[20] C. Xu, J. Shen, X. Du and F. Zhang. "An intrusion detection system using a deep neural network with gated recurrent units". *IEEE Access*, vol. 6, pp. 48697-48707, 2018.

[21] F. Jiang, Y. Fu, B. B. Gupta, F. Lou, S. Rho, F. Meng and Z. Tian. "Deep learning based multi-channel intelligent attack detection for data security". *IEEE Transactions on Sustainable Computing*, vol. 5, no. 2, pp. 204-212, 2020.

[22] Y. Bengio, A. Courville and P. Vincent. "Representation learning: A review and new perspectives". *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 8, pp. 1798-1828, 2013.

[23] Y. Zeng, H. Gu, W. Wei and Y. Guo. "Deep-full-range: A deep learning based network encrypted traffic classification and intrusion detection framework". *IEEE Access*, vol. 7, pp. 45182-45190, 2019.

[24] A. Diro and N. Chilamkurti. "Leveraging LSTM networks for attack detection in fog-to-things communications". *IEEE Communications Magazine*, vol. 56, no. 9, pp. 124-130, 2018.

[25] B. J. Radford, L. M. Apolonio, A. J. Trias and J. A. Simpson. "*Network Traffic Anomaly Detection Using Recurrent Neural Networks*", 2018. Available from: http://arxiv.org/abs/1803.10769. [Last accessed on 2021 Jun 08].