HYBRID INTRUSION DETECTION MODEL FOR COMPUTER NETWORKS

Mohammad Besharatloo[1] and Atiye Rahimizadeh[2] and Masoud Besharatloo[3]

[1] Master of Artificial Intelligence and robotics, ökmen mechatronics and software industry trade limited company
[2]Artificial Intelligence and robotics, Aryan university, Babol, Iran
[3]Master student of mechatronic , Golestan university, Gorgan, Iran

ABSTRACT

Intrusion detection is an important research topic in network security because of increasing growth in the use of computer network services. Intrusion detection is done with the aim of detecting the unauthorized use or abuse in the networks and systems by the intruders. Therefore, the intrusion detection system is an efficient tool to control the user's access through some predefined regulations. Since, the data used in intrusion detection system has high dimension, a proper representation is required to show the basis structure of this data. Therefore, it is necessary to eliminate the redundant features to create the best representation subset. In the proposed method, a hybrid model of differential evolution and firefly algorithms was employed to choose the best subset of properties. In addition, decision tree and support vector machine (SVM) are adopted to determine the quality of the selected properties. In the first, the sorted population is divided into two sub-populations. These optimization algorithms were implemented on these sub-populations, respectively. Then, these sub-populations are merged to create next repetition population. The performance evaluation of the proposed method is done based on KDD Cup99. The simulation results show that the proposed method has better performance than the other methods in this context.

KEYWORDS

Intrusion detection system, Differential evolution, Firefly Algorithm, Support vector machine, Decision tree.

1. INTRODUCTION

Intrusion detection systems have been extensively adopted in order to boost the security of computer networks. Log processing in computer networks are highly demanded because they are overloaded and low quality. The processing cannot be carried out manually by network managers because it is very time-consuming and demands a general look at variety of logs [1]. Securing a computer network consists of the provision of three main principles, namely confidentiality, integrity, and availability, which are determined according to security prevention and detection of violation of security policies. Majority of studies on this issue considered preventive security in computer networks through mechanized systems of identification, validation, encryption, and/or adoption of firewalls. Nowadays, the structure of most presented approaches for intrusion detection is based upon feature selection and reduction. This trend will cause some repetitive and unnecessary features, which delay detection process, to be ignored. In this regard, adoption of an effective algorithm in

David C. Wyld et al. (Eds): SIPM, ITCA, CMIT, FCST, CoNeCo, SAIM, ICITE, ACSIT, SNLP - 2023
pp. 41-50, 2023. CS & IT - CSCP 2023 DOI: 10.5121/csit.2023.130906

feature selection is of great importance [2,3]. Several methods have been introduced in recent years to present an appropriate intrusion detection emphasizing feature selection. What follows is a review of studies on intrusion detection in computer networks. In [4], genetic algorithm was used to select feature subclasses for classification of features in order to classify decision tree to increase detection rate and decrease false warning rate in intrusion detection in network. A hybrid system for intrusion detection using dynamic swarm intelligence based on rough set was proposed in [5] to select features for classification of intrusion data.

In [6], an intrusion detection system was proposed using genetic algorithm for more effective detection of various intrusions in network. Furthermore, an intrusion detection method based upon cuttlefish algorithm (CFA) was recommended in [7]. In this paper, CFA and decision tree classification were used as a search strategy to obtain optimal subsets of features and to determine the selected feature subset, respectively. In [8], a GA-LR wrapper approach was proposed for feature selection. In this method, genetic algorithm and regression were adopted as search strategy and learning algorithm for intrusion detection systems to select the best feature subset, respectively.

In [9], the relationship between network flows and recorded information by baits was scrutinized. In this method, a confident system for validation of network traffic is required. In this paper, a database is developed through collecting information about network flow and recorded information about baits, based upon which worm flow signature is meticulously extracted. Data mining algorithms of Bayesian network and support vector for intrusion detection were determined in [10]. In this research, the best algorithm was sought by considering training sets obtained from the algorithms. In [11], intrusion detection systems by support vector machine were reviewed. It was concluded in this study that intrusions can be recognized from one another and interpretation error can be minimized by the classification of support vector machine. However, support vector machine method needs a long period of education. Therefore, education period can be substantially decreased and network accuracy can be increased by use of hybrid methods, e.g. clustering algorithm and support vector machine. In [12]. Intrusion detection systems based on unsupervised neural networks were studied. The results of this research indicated that various models of neural networks with each having specific features are able to detect intrusions and disconnections precisely in computer systems.

In [13], recent techniques of intrusion detection based on genetic algorithm were analyzed. Considering the necessity of studying feature selection in designing an intrusion detection system, a feature selection method based on a hybrid model of differential evolution and firefly algorithms was proposed in this study in order to select the optimal subset out of all features. This was done to achieve a more appropriate data representation and higher accuracy and efficiency of classification in intrusion detection systems.

In the proposed method, the determination of each subset of the selected features is carried out by iterative dichotomiser 3 (ID3) and support vector machine. In the next section, intrusion detection is taken into account. Section 3 will describe the proposed method accurately and explain it principles. Simulation results and conclusion are presented in sections 4 and 5, respectively.

2. FEATURE ASSIGNMENT IN INTRUSION DETECTION SYSTEM

The dimension of the used data in designing an intrusion detection system will be high and therefore, a set of features should be extracted from the input data with lower dimensions so that problem solving will be done with less complication. Majority of introduced algorithms to detect intrusions emphasize appropriate feature selection and reduction of input data dimension. Reduction of input data dimension based on the selection of appropriate feature set leads to the

elimination of unnecessary repetitive data and reduces the time needed for detection process. Therefore, adoption of an appropriate algorithm in the determination of the best subset of features among all the feature sets seems crucial. The best subset of features is able to demonstrate all the features of dataset and remove redundant and unrelated features from the dataset. As mentioned in Introduction, evolutionary algorithms have had favorable results in feature selection process in designing an intrusion detection system.

The proposed method in this paper was based on firefly and differential evolution algorithms, which is highlighted in the subsequent section.

3. THE PROPOSED METHOD

In the proposed method, a hybrid model of firefly and differential evolution algorithms was used as a search strategy to select the best subset of features and decision tree classifier and support vector machine were applied to determine the quality of the selected features.

To run optimization process in the proposed method, population is sorted in an ascending order in each iteration. Afterwards, the population is divided into two subpopulations where the first one includes the individuals with higher fitness while the second comprises the individuals with lower fitness. Then, an iteration from firefly algorithm is run on the first subpopulation (including individuals with higher fitness) and subsequently, an iteration from differential evolution algorithm is run on the second subpopulation. Next, the newly formed subpopulations are merged and again divided into two subpopulations following another sorting. Likewise, another iteration of firefly and differential evolution algorithms are run on them. This process continues until the ending condition, i.e. the maximum number of iterations, is achieved.

The block diagram below illustrates the proposed method. What follows is the explanation of detailed steps in the proposed method.

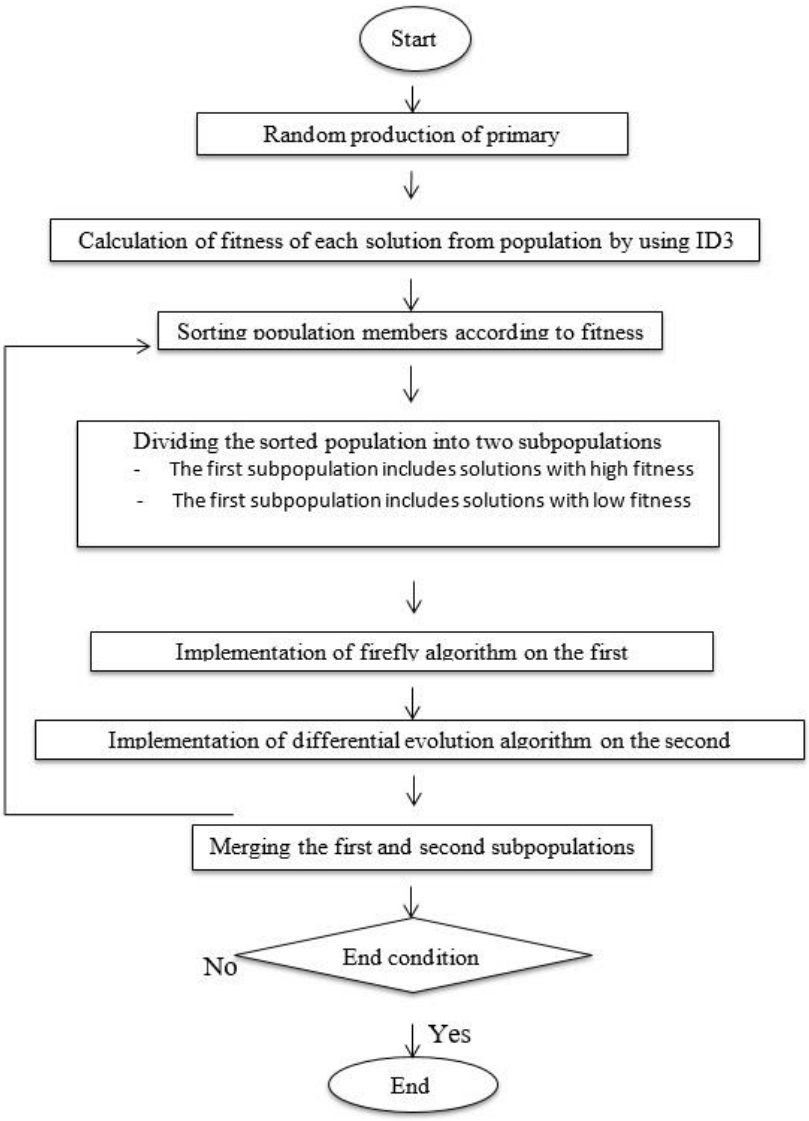

Figure 1: Block diagram of the proposed method

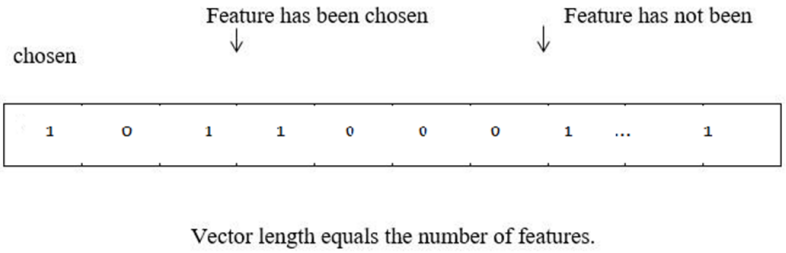

Figure 2: Encoding a vector for feature selection problem

Step 1: A primary population of M solutions are produced randomly. In the proposed algorithm, each solution from the population shows a possible solution from feature election problem, which means each solution can select a subset of features. The problem nature specifies the encoding manner for each solution. In feature selection problem, a binary vector X is used to encode n

features available in dataset. Therefore, the vector $(\ )\ 1\ 2\ ,\ ,\ ,\ n\ X = x\ x\ \ldots\ x$ is a population solution with $\{0,1\}\ i\ x \in$ and$1 \le i \le n$. If the ith feature in the current subset is selected, $1\ i\ x$ = and if not, $1\ i\ x =$. Fig. 2 depicts a sample of encoding a solution from population.

Step 2: Fitness level of each solution from population is calculated. In this step, a subset of feature selected by that solution is specified for each solution X from population and accordingly, the selected features of dataset is reduced. Afterwards, using this reduced dataset, decision tree classifier ID3 or SVM are implemented and each data class is tagged. Then, three criteria of detection system is calculated and finally, fitness function value is obtained. The fitness function is defined according to three efficiency criteria of intrusion detection system, which are attack detection rate (ADR), false positive rate (FPR), and accuracy rate (AR). The criteria are defined as below :

$$DR = \frac{Number\ of\ true\ detected\ attacks}{Total\ number\ of\ attacks} \quad (1)$$

$$FPR = \frac{Number\ of\ false\ detection\ in\ normal\ attacks}{Total\ number\ of\ normal\ attacks} \quad (2)$$

$$AR = \frac{Number\ of\ true\ detected\ samples}{Total\ number\ of\ samples} \quad (3)$$

Every intrusion detection system should improve ADR and decrease FPR. Therefore, higher values of DR and AR and lower values of FPR represent a better classification for intrusion detection systems. So, fitness value for each solution X from the considered population is calculated based upon these three criteria as follows:

$$F(X) = \alpha_1\ AR + \alpha_2\ DR + \alpha_3 (1 - FPR) \quad (4)$$

Step 3: Algorithm iteration counter is set at.
Step 4: If ending condition (the maximum number of algorithm iterations) is not achieved, steps 5-11 are run and otherwise, algorithm implementation ceases.
Step 5: The existing solutions in population are sorted in an ascending order based on their fitness. The sorted population is divided into two subpopulations with each including M/2 solutions. The first subpopulation includes the solutions with the highest fitness and the second subpopulation comprises the solutions with lower fitness values compared to the existing solutions in the first subpopulations.
Step 6: Steps of firefly algorithm are run to update the first subpopulation.
Step 6-1: Firefly population is considered equal to the first subpopulation.
Step 6-2: for each firefly from $1 \le p \le M/2$, step 6-2-1 is run.
Step 6-2-1: for each firefly from $1 \le q \le M/2$, step a is run.
Step a: If transparency of firefly p is lower than that of firefly q, steps b and c are run.
Step b: The movement of firefly p toward firefly q is as follows:

$$X'_p = X_p + \beta(p,q)(X_p - X_q) + \alpha\left(rand - \frac{1}{2}\right) \quad (5)$$

where a is a random parameter and rand forms a random number in [0,1]. The value is the absorption rate between two fireflies calculated according the following absorption strategy:

$$\beta(p,q)=0.5\times\left(\frac{1}{cost(p,q)+1}+return(p,q)\right) \quad (6)$$

Where return (p,q) is defined as follows:

$$return(p,q)=\frac{f(q)-f(p)}{f_{max}-f(q)} \quad (7)$$

Here, is the maximum value of current fitness among fireflies population.max f_{max} is the maximum value of current fitness among fireflies population. Moreover, f(p) and f(q) denote fitness of fireflies p and q, respectively. The function cost is calculated as follows:

$$cost(p,q)=\sum_{i=1}^{n}\left|X_{p,i}-X_{q,i}\right| \quad (8)$$

Where $X_{p,i}$ stands for the component i from the p firefly solution. Afterwards, in order to value each component of firefly as 0 or 1, sigmoid function or is calculated for each component tanh X_p according to one of the following equations:

$$f(X'_{p,i})=\frac{1}{1+\exp(-X'_{p,i})} \quad (9)$$

$$f(X'_{p,i})=\tanh(|X'_{p,i}|)=\frac{\exp(2*|X'_{p,i}|)-1}{\exp(2*|X'_{p,i}|)+1} \quad (10)$$

The final position of firefly p is obtained as follows:

$$X'_{p,i}=\begin{cases}1 & if\ f(X'_{p,i})>rand,i=1,..,n\\ 0 & otherwise\end{cases} \quad (11)$$

Step *c*: The new firefly is determined and its fitness is obtained.
Step 6-3: The current and new fireflies are sorted according to their fitness levels and the best ones are selected to build the first subpopulation.
Step 7: The steps of differential evolution algorithm are taken in order to update the second subpopulation.
Step 7-1: The primary population of differential evolution algorithm is considered equal to the second subpopulation.
Step 7-2: Mutation is run on each vector $^{x}\mathbf{i}$ of population as follows to obtain the vector $^{v}\mathbf{i}$

$$V_i=X_{best}+F*(X_{r_1}-X_{r_2}) \quad (12)$$

Where $\mathbf{r_1}$ and $\mathbf{r_2}$ are the selected random integers from {1,2,3,…,M/2,(R_1 ,R_2#I)} .The vector X_{best} will be the solution of the best individual found hitherto in the proposed algorithm. F denotes positive integer for scaling factor.

Step 7-3: Crossover in differential evolution algorithm is run on two vectors X_i , V_i based on the following equation to acquire the vector U_i :

$$u_i(j) = \begin{cases} v_i(j) & if\,(rand \le CR)or(j = j_{rand}) \\ x_i(j) & otherwise \end{cases} \quad (13)$$

Where I = 1,2,3,…,M/2 , j=1,2,3,…,n , and crossover rate is a fixed number between 0 and 1 to control the fraction of the newly formed parameter values from the vector V_i Also J_{rend} obtains at least one component from V_i .

Step 7-4: Selection in differential evolution algorithm compares to vectors X_i and U_i in terms of the value of objective function to specify the new vector placed in the population. Selection process is in a way that the individual with the highest fitness replaces the previous individual in the population and this previous individual is removed. Therefore, if objective function is defined for the purpose of maximization, the vector with larger objective function is placed in the next generation population. Selection is defined as below:

$$x_i = \begin{cases} u_i & if\ f(u_i) \ge f(x_i) \\ x_i & otherwise \end{cases} \quad (14)$$

Where **f (.)** returns objective function in terms of its input.
Step 8: Two resulting subpopulations from firefly and differential evolution algorithms are merged.
Step 9: The best solution found hitherto is specified.
Step 10: Algorithm iteration counter increases for one unit.
Step 11: If ending condition is not met, the process return to step 4.

4. SIMULATION RESULTS

This section presents the efficiency of the proposed algorithm by various tests. In all the performed simulations, a system with Intel® processor, U™ i5-4201 core, CPU@1.7GHz2.40 GHz, 8 GB memory and an 8.1 64-bit operating system was used.

4.1. Database

In the performed test in order to assess the efficiency of the proposed algorithm, the standard dataset KDD cup'99 was adopted [15]. This dataset has been extracted from the intrusion detection program DARPA 1998. It includes a wide variety of simulated intrusions in military network environment, which are commonly used to analyze intrusion detection methods. This dataset has 41 descriptive and continuous features for each record in addition to a class tag that shows whether the connection is normal or is an attack. There are 22 kinds of attack classified into four groups, namely denial of service attack (DoS), probing, user to root attack (U2R), and remote to local attack (R2L). Two separate datasets (i.e. educational and test datasets) are required for analysis of intrusion detection systems. Since data size in this database is too large, only 10 percent of data in this set was used for testing.

Here, educational dataset includes 494021 records, from which 97280 records are normal connections. Furthermore, test dataset includes 311029 records, from which 60593 records are normal connections. Table 1 shows the distribution of each intrusion in educational and test datasets. The size of these datasets is still too large to be used. Therefore, in the experiments run in this study, two educational and test datasets were haphazardly extracted. In addition, the number of samples of each attack was divided by 100 to keep the fraction of each attack in both educational and test datasets.

4.2. Simulation Details

The proposed intrusion detection systems in this paper is compared to the systems based upon firefly, differential evolution, and particle swarm optimization. Analysis of subset of selected features in all experiments are according to ID3 and SVM. An example of dataset record KDD cup'99 is shown in Fig. 3. Since SVM can only be used with numerical data, it is necessary to turn non-numerical features of data into numerical state. For instance, service feature with tcp value shown in Fig. 3 is considered to be 3 as numerical value. Therefore, the data in Fig. 2 is turned into the numerical form shown in Fig. 4 [14].

In the proposed method, population size in firefly, differential evolution, and particle swarm optimization algorithms is 30 and the maximum iterations of algorithms is considered 50. Moreover, the results in all the experiments are the mean of intrusion detection criteria in twenty independent runs.

4.3. Experiment Results

This section presents the results of the proposed method and firefly, differential evolution, and particle swarm optimization algorithms as well as the mode of using all features according to SVM and ID3.

Table 1: Distribution of different intrusions in dataset

Data	Normal	Probe	DoS	U2R	R2L	Total
Educational	97280	4107	39458	52	1124	49021
Test	60593	4166	229853	228	16189	311029

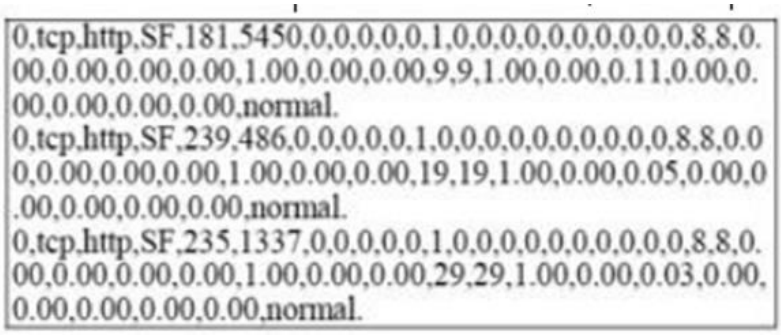

Figure 3: An example of main data in dataset KDD cup'99

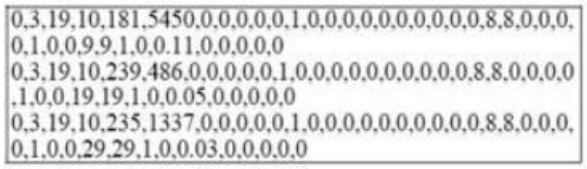

Figure 4: An example of main data after turning into numerical form

Figure 4: An example of main data after turning into numerical form Firstly, a binary with hamming distance from main type of firefly algorithm and use of two types of fitness functions were determined. In the first experiment, the proposed method is compared with firefly algorithm, which uses absorption strategy based on hamming distance in order to calculate the absorption between two fireflies. Results are obtained by using ID3 to calculate fitness of the proposed method, firefly algorithm, differential evolution algorithm, and adoption of all features according to two types of fitness functions. The first fitness function is defined as follows:

$$Fitness(X) = \alpha DR + \beta(1 - FPR) \qquad (15)$$

This equation states that DR and FPR are differently important in terms of the coefficients α and β where a €[0,1] and β = 1-a. These two weight coefficients are adjusted experimentally at a = 0.7 and β = 0.3 . The second fitness function is defined as below according to accuracy and number of selected features.

$$Fitness(X) \models \alpha_1 AR + \alpha_2 DR + \alpha_3 (1 - FPR) \qquad (16)$$

Here a_1 , a_2 and a_3 are weighing coefficients for each defined objective function, which equal 0.3 , 0.4 and 0.35.

respectively. Table 2 reports the results of three criteria (i.e. DR, FPR, and AR) for the proposed algorithms compared to other algorithms by using the first fitness function. Each intrusion detection system should improve attack detection rate and reduce false warning rate. Therefore, higher values of DR and AR and lower value of FPR indicates better classification performance for intrusion detection systems. As shown in Table 2, the proposed algorithm acted better than other method followed by differential evolution algorithm.

Table 3 presents the results of these three criteria for firefly, differential evolution, and the proposed algorithms and use of all features in adoption of the second fitness function. As seen in Table 3, the proposed method had better efficiency in terms of AR and DR followed by firefly and differential evolution algorithms.

In terms of FPR, firefly algorithm had the best efficiency followed by the proposed method. What follows is a comparison of the results obtained by the proposed method and other algorithms in terms of ID3. Table 4 reports the results of various criteria in the experiment in dataset for each method based on ID3. This table also presents the results of PSO algorithm.

According to the results, the proposed method yielded favorable performance in detection of intrusion type. As seen in table 4, the proposed method obtained the highest values in terms of AR and DR. Furthermore, the lowest value of FPR was achieved by the intrusion detection method based upon firefly algorithm. Here, the results of the proposed method and other algorithms based on SVM are reported. Table 5 shows the results of various criteria for each

method according to SVM. As seen in Table 5, the proposed method yielded the highest values among various optimization methods and was able to adopt all features.

Comparison of different methods for AR, DR, and FPR based on SVM and ID3 were shown in Figs. 5 and 6, respectively. As seen in Fig. 4, different methods with SVM reached higher AR accuracy than with ID3. Moreover, the proposed method yielded the highest values in both SVM and ID3. With regard to Fig. 7, SVM yielded higher values than ID3 in terms of analysis criteria for the efficiency of intrusion detection systems. However, the proposed method was unable to reach the highest values in both SVM and ID3. Results of FPR indicate the superiority of the proposed method over other methods.

Table 2: Percentage of analysis criteria for comparing different algorithms according to ID3 and the first fitness function.

	AR	FPR	DR
Firefly based on hamming	*91.44*	*0.664*	*92.13*
Decision tree with all features	*73.26*	*17.68*	*71.08*
Differential evolution	*91.87*	*0.726*	*92.50*
The proposed method based on hamming	*92.24*	*0.662*	*92.67*

Table 3 : Percnetage of analysis criteria for comparing difference algorithms accoording to ID3 and the second fitness function

	AR	FPR	DR
Firefly based on hamming	92.93	0.83	92.29
Decision tree with all features	17.68	71.08	73.267
Differential evolution	90.61	3.053	93.97
The proposed method based on hamming	92.96	1.81	94.63

Table 4 : Mean results of analysis criteria for the proposed method and other algorithms based on ID3

	AR	FPR	DR
Firefly	92.18	0.747	92.21
Decision tree with all features	45.47	44.38	72.17
Differential evolution	91.24	1.889	93.23
Particle swarm optimization	92.01	1.74	92.13
The proposed method	92.60	1.236	93.65

Table 5 : Percentage of analysis criteria fo the proposed method and other algorithms according to SVM

	AR	FPR	DR
The proposed method	95.87	1.41	95.02
Differential evolution	95.19	3.02	94.76
Firefly	95.45	3.45	94.89
Particle swarm optimization	95.22	3.85	94.99
SVM with all features	86.79	6.97	70.03

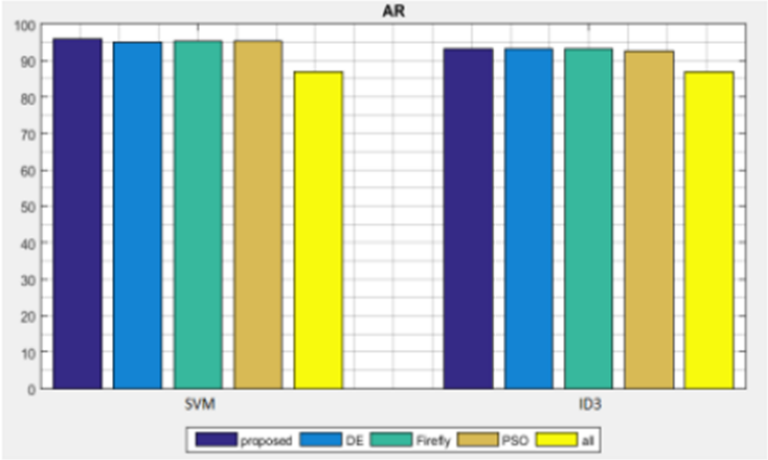

Figure 5: Comparioson of values related to AR for different methodsbased on SVM and ID3

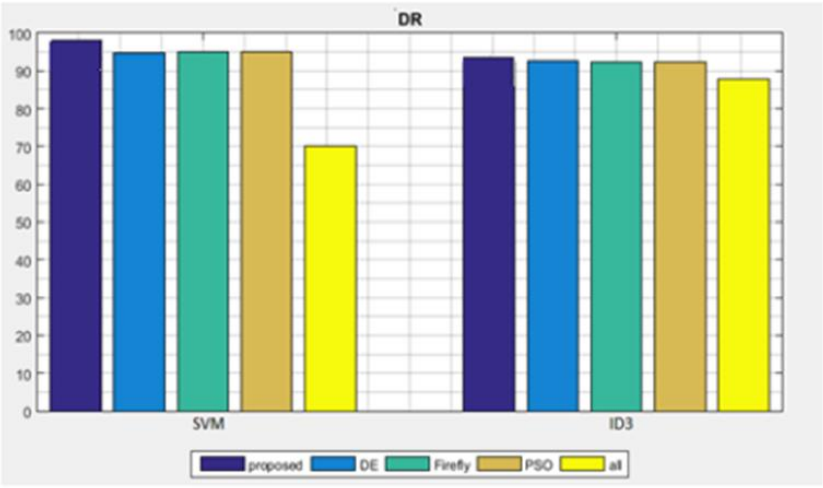

Figure 6 : Comparioson of values related to DR for different methodsbased on SVM and ID3

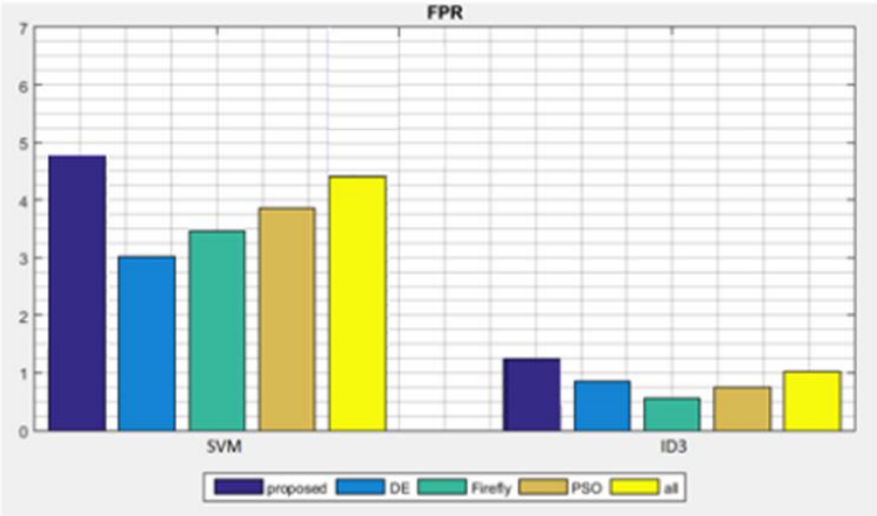

Figure 7 : Comparioson of values related to FPR for different methods based on SVM and ID3

REFERENCES

[1] Azarkasb, S. A., Gheidri, S. Intrusion detection through log correlation. M.Sc. thesis, Islamic Azad University, Qazvin Branch, Qazvin, Iran, 2009.

[2] Fazli Maghsoudi, H., Momeni, H. Comparison and determination of Bayesian network datamining algorithms and support vector for intrusion detection, Computer Engineering and Sustainable Development, Khavaran Higher Education Institution, Mashhad, Iran, 2013.

[3] Salehpour, N., Moradi, S., Nazari Farrokhi, M. A review on intrusion detection systems by using support vector machine, The 9th Symposium on Science and Technology Development, Mashhad, Iran, 2014.

[4] Khodabandehloo, R., Khalilian, N. Determination of intrusion detection systems based on unsupervised neaural networks, The 9th Symposium on Science and Technology Development, Mashhad, Iran, 2014.

[5] Seivandian, Z., Rastgari, H. Determination of the use of genetic algorithm in intrusion detection in computer networks, The 2nd National Conference on Computer Sciences and Engineering, Islamic Azad University, Najafabad Branch, Najafabad, Iran, 2014.

[6] P.Bahraini and R. Jalili, (2006), "A method for analyzing the correlation of alerts in network intrusion detection systems", end Master's thesis, Sharif University of Technology, Faculty of Computer Engineering, Iran

[7] D. J. Hand, H. Mannila, and P. Smyth, (2001), *Principles of data mining*: MIT press.

[8] J. Han, J. Pei, and M. Kamber ,)5133(,*Data mining: concepts and techniques*: Elsevier.

[9] T. M. Mitchell, (1997), "Machine learning. 1997," *Burr Ridge, IL: McGraw Hill,* vol. 45, no. 37, pp. 870-877.

[10] R. O. Duda, P. E. Hart, and D. G. Stork, (1973), *Pattern classification*: Wiley, New York.

[11] J. Markey, and A. Atlasis, (2011), "Using decision tree analysis for intrusion detection: a how-to guide," *SANS Institute InfoSec Reading Room.*

[12] X.-S. Yang), "Firefly algorithms for multimodal optimization." pp. 169-178.

[13] Y. Zhang, X.-f. Song, and D.-w. Gong, (2017), "A return-cost-based binary firefly algorithm for feature selection," *Information Sciences,* vol. 418, pp. 561-574.

[14] K. Chandrasekaran, S. P. Simon, and N. P. Padhy, (2013), "Binary real coded firefly algorithm for solving unit commitment problem," *Information Sciences,* vol. 249, pp. 67-84.

[15] R. Storn, and K. Price, (1997), "Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces," *Journal of global optimization,* vol. 11, no .4 ,pp. 341-359.

AUTHOR

NAME : Mohammad
SIR NAME : Besharatloo
DATE OF BIRTH : 12/06/1991
MARITAL STATUS : Single

EDUCATION:

2007-2011: Associate in computer engineering of Azad University Ramsar/Iran
2012-2014: B.A degree in software computer engineering of Azad University gorgan/Iran
2015-2018: M.A Degree in artificial intelligence and robotics of Mazandaran University babolsar/Iran