IJICTR
International Journal of Information & Communication Technology Research

Volume 13- Number 3 – 2021 (39 -47)

A Feature Selection Method Base on Fisher Score for Detecting SVM-Based Network Intrusions

Mohammad Hassan Nataj Solhdar*
Department of Shohadaye hoveyzeh Campus of technology
Shahid Chamran University of Ahvaz
Ahvaz, Iran
n.solhdar@scu.ac.ir

Received: 7 March 2021 - Accepted: 13 May 2021

Abstract—**The diagnostic process is based on the fact that malicious activity is different from the activity of a normal system. Detection of intrusion is a very complex process. In this paper, we propose Feature Selection to improve the velocity support vector machines (SVM) based intrusion detection system (IDS). The new model has used a feature selection method based on Fisher Score with an innovation in fitness function reduce the dimension of the data, increase true positive detection and simultaneously decrease false positive detection. In addition, the computation time for training will also have a remarkable reduction. We demonstrate the feasibility of our method by performing several experiments on NSLKDD intrusion detection system competition dataset. Results show that the proposed method can reach high accuracy and low false positive rate (FPR) simultaneously. Numeric Results and comparison to other models have been presented.**

Keywords: SVM; NSLKDD; feature selection; IDS; Fisher Score.

Article type: Research Article

[DOI: 10.52547/ijict.13.3.39]

I. INTRODUCTION

Machine learning is further than an artificial intelligence field. Researchers in the elementary days of fabricating the artificial intelligence as a scientific field found that machines learn from data [1]. They tried to solve this issue with varied symbolic technics and what was called “neural network” at that time. These technics were mostly perceptron and learning models which were indicated later that they were redesigning of the generalized linear models [2].

Detection of the known attacks is not difficult. Generally, signature-based or rule-based technics are used. However, the big challenge is the unknown attacks. One of the main developments of machine learning is an ensemble technic in recent years that makes high-precise classification by the combination of the higher-balanced classification components [3].

* Corresponding Author

Signature-based IDSs rely on the human for construction, test, and development of signatures. Therefore, many hours or days may be needed to produce a signature for an attack [4]. This long time is for the quick attacks. Nonetheless, a solution to be independent of the human is suggested for the mentioned problem [5]. Anomaly-based IDSs regarding machine learning add an extra advantage. Anomaly-based IDSs use Machine Learning technic which can implement a system to learn from data (experiences) and make the decision for unseen data [6].

Fig. 1 has shown that machine learning technic is used for the intrusive and non-intrusive behaviors [7]. As it is seen in figure 1, SVM is sub-branch of machine learning. SVM is one of the learning technics by supervision which is used for classification and regression [8].

This relatively new technic has shown better efficiency than the older one for classification problems solving - perceptron neural networks. The working basis of SVM classification is data linear classification, and it is tried in data linear division to select the line with more confidence margin [9]. States that, as a rule of thumb, the required cardinality of the training set for accurate training increases exponentially with the input dimension [11]. Thus, choosing a small subset of the thousands of possible features, i.e. feature selection, requires a small fraction of the training samples required if all features are used. Feature selection is Relatively the process of identifying those features that contribute most to the discrimination ability of the neural network. Only these features are then used to train the neural network and the rest are discarded. Proposed methods for selecting an appropriate subset of features are numerous [12]. Here, the dimensionality of a feature set is reduced by combining features while retaining characteristics that allow for accurate classification. Feature selection is the process of mapping all available features into a composite feature set of lower dimension [13].

Many feature selection techniques such as the principle components algorithm are based on the assumption that the greater the spread of the data in a particular axis, the greater the effect that will have on the discrimination ability of the neural network. This need not be true. Feature selection methods, on the other hand, generally are based on ranking different combinations of features in accordance to their classification performance and choosing the combination that achieves the highest ranking. Unlike feature selection, no preprocessing is required once the features are chosen.

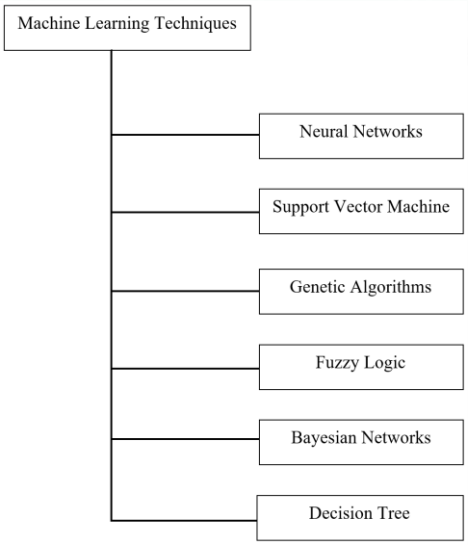

Figure 1. Classification of machine learning technics [10]

II. FEATURE SELECTION

Feature selection is one of the highly used problems in modeling. The problems of the real world ordinary have a lot of data that reducing the input has been always inevitable to model these problems by the present tools [14]. The meaning of feature selection is selecting a group of useful features from a group of complete features. Using these useful features not only can reduce the volume of present data for modeling, but also can improve the efficiency of the model [15]. Many researchers have found many technics for feature selection and used them on their data to be able to reduce processing. This point must be considered in feature selection that the final remained feature must cover the features of total data as much possible as generalizing the obtained results to all [16].

The extra and irrelevant features can have a negative effect on detection power of IDS. Now, it is tried in IDS input selection to eliminate the extra and irrelevant dataset. The advantages of feature selection or IDS can be as follow [17]:

- First, detection rate will be increase by feature selection and consequently reduce input data because of less data to be processed to detect the type of connection.
- Second, detection power may be increases by removing the non-effective data or data with negative effects in detection.

A. Related Work in feature selection

In the past decade, a number of performance criteria have been proposed for filter based feature selection, such as mutual information [18], Fisher score [19], ReliefF [20], Laplacian score [21], Hilbert Schmidt Independence Criterion (HSIC) [22] and Trace Ratio

[DOI: 10.52547/ijict.13.3.39]

criterion [23], among which Fisher score is one of the most widely used criteria for supervised feature selection due to its general good performance. Javidi et al. [24] and [25] attempt to construct a neural network using MLP in parallel. Several feature selections were implemented and compared in article [17]. We also suggested these algorithms to be compared with the suggested IDS. In this article, three technics of Bayesian, classification, and regression trees and the combination of these two technics were used. In this research, the researchers succeeded to reduce features using Bayesian network, and Markov covering properties for each group (each group shows one feature).

B. Bayesian network

Bayesian network is a directional non-cyclic graph that each node in this graph introduces one variable of the problem range (like features) and is shown by set of B= (N, A, Q). In this set, N is total nodes (features) and A is set of edges. Each edge in set A shows the probability of correlation among the related nodes that is weighted using conditional probability for each node in set N. Conditional probability for each node is in set Q. In Bayesian network, covering Markov for each node is a set of nodes including parents and ancestors of nodes, children, and other parent of node children [26]. The covering Markov set for a node includes all nodes that separate the mentioned nodes from the rest of network and this set is efficient to predict node behaviors [27].

1) Classification And Regression Tree

Decision tree classifies samples by arranging them from root node to the bigger nodes in tree. Each internal node in tree tests features of the sample and each existing branch from that node correspond with the possible value for that feature. Moreover, one classification is featured to each branch node. Each sample is classified by starting from tree root node and the detected feature test by this node, and movement in the corresponding branch with the featured value in the sample. This process is repeated for each sub-tree whose root is a new node [6].

When the output of a tree is a discrete set of possible values, that tree is called classification. When the tree output can be considered as the real number, it is called regression tree. CART is called to both mentioned procedures. CART is the abbreviation of Classification And Regression Tree.

According to table 3 about NSLKDD dataset, 17th, 12th, and 19th features used in this article are as follows:

12th features obtained from feature selection include:

12 features of CART:

C,E,F,L,W,X,Y,AB,AE,AF,AG,AI

17 features of CART:

A,B,C,E,G,H,K,L,N,Q,V,W,X,Y,Z,AD,AF

19 features of CART:

A, B, E, F, H, K, L, Q, S, T, V, W, X, Y, AB, AD, AF, AG, AI

III. NSLKDD DATASET

This data was used for the Third International Knowledge Discovery and Data Mining Tools Competition that is symmetrical with Fifth International Conference on Knowledge Discovery and Data Mining. The aim of this data is making IDS for the network that this model is able to differ between the “bad” connections called intrusion or attack and “good” connections called normal. This dataset includes a collection of standards for data including the extensive spectrum of the simulated intrusion in a military network environment.

NSLKDD is the collection of the suggested data to solve some innate problems of KDD'99 data collection that is mentioned in [28]. Table 1 and table 2 show the number of normal and total records in training and testing data and reduction rate of record numbers in comparison to KDD’99 dataset. Moreover, Table 3 shows all 41 features in NSLKDD dataset. Table 4 represents the number of observations for each attack sorted in one of the four intrusion states. Testing data introduces some new types of attacks, marked with gray shade. Observations for these attacks are not available during model training. The NSLKDD dataset includes a state for each set of features, where the state is either a normal connection or a type of attack as represented in Table 4. This means that each record in the data belongs to one of five major classes: Normal, DoS, Probe, U2R, and R2L. The values for each state are mapped to a numeric value. More specifically the Normal class was mapped to the number 1, Probe to 2, DoS to 3, U2R to 4, and R2L to 5.

TABLE I. RECORD NUMBERS ON TRAINING DATA

	Main record	**Different records**	**Reduction rate**
Attacks	3,925,650	262.178	93.32%
Normal	972,781	812,814	16.44%
Total	4,898,431	1,074,992	78.05%

TABLE II. RECORD NUMBERS IN TESTING DATA

	Main record	**Different records**	**Reduction rate**
Attacks	250436	29378	88.26 %
Normal	60591	47911	20.92 %
Total	311027	77289	75.15 %

[DOI: 10.52547/ijict.13.3.39]

TABLE III. NSLKDD DATASET FEATURE AND THE SCORE OF FEATURES BASED ON FISHER SCORE

Feature Name	Description	Labile	Fisher Score
Duration	length (number of seconds) of the connection	A	28
protocol_type	type of the protocol, e.g. tcp, udp, etc.	B	20
Service	network service on the destination, e.g., http, telnet, etc.	C	27
Flag	normal or error status of the connection	D	30
src_bytes	number of data bytes from source to destination	E	3
dst_bytes	number of data bytes from destination to source	F	7
Land	1 if connection is from/to the same host/port; 0 otherwise	G	1
wrong_fragment	number of ``wrong'' fragments	H	18
Urgent	number of urgent packets	I	5
Hot	number of ``hot'' indicators	J	6
num_failed_logins	number of failed login attempts	K	4
logged_in	1 if successfully logged in; 0 otherwise	L	36
num_compromised	number of ``compromised'' conditions	M	10
root_shell	1 if root shell is obtained; 0 otherwise	N	8
su_attempted	1 if ``su root'' command attempted; 0 otherwise	O	13
num_root	number of ``root'' accesses	P	12
num_file_creations	number of file creation operations	Q	11
num_shells	number of shell prompts	R	9
num_access_files	number of operations on access control files	S	15
num_outbound_cmds	number of outbound commands in an ftp session	T	40
is_host_login	1 if the login belongs to the ``hot'' list; 0 otherwise	U	41
is_guest_login	1 if the login is a ``guest''login; 0 otherwise	V	14
Count	number of connections to the same host as the current connection in the past two seconds	W	31
srv_count	number of connections to the same service as the current connection in the past two seconds	X	2
serror_rate	% of connections that have ``SYN'' errors	Y	32
srv_serror_rate	% of connections that have ``SYN'' errors	Z	33
rerror_rate	% of connections that have ``REJ'' errors	AA	24
srv_rerror_rate	% of connections that have ``REJ'' errors	AB	23
same_srv_rate	% of connections to the same service	AC	39
diff_srv_rate	% of connections to different services	AD	21
srv_diff_host_rate	% of connections to different hosts	AE	19
dst_host_count	count for destination host	AF	29
dst_host_srv_count	srv_count for destination host	AG	38
dst_host_same_srv_rate	same_srv_rate for destination host	AH	37
dst_host_diff_srv_rate	diff_srv_rate for destination host	AI	22
dst_host_same_src_port_rate	same_src_port_rate for destination host	AJ	17
dst_host_srv_diff_host_rate	diff_host_rate for destination host	AK	16
dst_host_serror_rate	serror_rate for destination host	AL	34
dst_host_srv_serror_rate	srv_serror_rate for destination host	AM	35
dst_host_rerror_rate	rerror_rate for destination host	AN	25
dst_host_srv_rerror_rate	srv_serror_rate for destination host	AO	26

TABLE IV. ATTACK DISTRIBUTION

Class	In training		Total	Testing		Total
	Attack names	Samples		Attack names	Samples	
DOS	teardrop	979	391,458	Apache 2	794	229853
	smurf	280,790		Back	1098	
	neptune	107,201		land	9	
	Pod	264		mailbomb	5000	
	Back	2203		neptune	58001	
	Land	21		pod	87	
				processtable	759	
				smurf	164091	
				teardrop	12	
				udpstorm	2	
Probe	satan	1589	4107	ipsweep	306	4166
	nmap	231		mscan	1053	

[DOI: 10.52547/ijict.13.3.39]

	ipsweep	1247		nmap	84	
	portsweep	1040		portsweep	354	
				saint	736	
				satan	1633	
	perl	3		buffer overflow	22	
	buffer overflow	30		loadmodule	2	
	rootkit	10	52	perl	2	
U2R	loadmodule	9		ps	16	70
				rootkit	13	
				sqlattack	2	
				xterm	13	
	ftp write	8		ftp write	3	
	Warezclient	1020		guess passwd	4367	
	Warezmaster	20		imap	1	
	Spy	2		multihop	18	
	guess passwd	53	1126	named	17	
	Imap	12		phf	2	
	multihop	7		sendmail	17	
R2L	Phf	4		snmpgetattack	7741	16,347
				Snmpguess	2406	
				warezmaster	1602	
				worm	2	
				xlock	9	
				xsnoop	4	
				httptunnel	158	

IV. SVM CLASSIFIER

Support vector machines (SVM) are an effective technique for solving classification and regression problems. SVM is originally an implementation of Vapnik's Structural Risk Minimization (SRM) principle [29], which is known to have low generalization error or equivalently does not suffer much from overfitting to the training data set. A model is said to overfit or has a high generalization error if it performs poorly on instances not present in the training set. SVM is particularly effective on data sets that are linearly separable, i.e. where hyperplane H can be found that partitions the instances into two classes such that instances in one class (almost) entirely fall on one side of H. Since there is an infinite number of candidate hyperplanes that can be selected, SVM selects the hyperplane H so that it maximizes its distance to the nearest data points in either class. This is referred to as margin maximization. So far, we have only considered the case where the data set is linearly separable. However, for many real-life data sets, such a hyperplane may not exist. In these cases, SVM uses a function to map the data into a different feature space where such separability is then possible. This transformation often comes in the form of mapping to a high-dimensional space. A function used to perform such a transformation is called a kernel function. Thus, kernel functions play a pivotal role both in the theory and application of SVM. The following kernel functions are commonly used along with SVM [30].

Linear kernel: $k(x_i, x_j) = x_i x_j$

Polynomial kernel: $k(xi, xj)=(yx^t_i\, x_j + r_d)^2$

RBF kernel: $K(x_i, x_j) = e^{\gamma\|x_i - x_j\|^2}$

Sigmoid kernel: $k(xi, xj) = tanh(yx^t_i\, x_j + r)$

V. SUGGESTED TECHNIC (FEATURE SELECTION BASED ON FISHER SCORE)

In many modeling problems where a large amount of data is to be given to a model for clustering or classification, it is possible that some data will delay the process and even lead to errors. The data that causes this is redundant or irrelevant. The purpose of reducing input is to remove this data from the input data set to the system.

Fisher score for each feature, it selects the top-m ranked features with large scores. Because the score of each feature is computed independently, the features selected by the heuristic algorithm is suboptimal. the other algorithm fails to select those features which have relatively low individual scores but a very high score when they are combined together as a whole. In addition, it cannot handle redundant features [31]. This motivates us to propose a Fisher score which can resolve these problems.

Fisher Score is a useful feature selection tool that works based on distance from data centers. The main point of the Fisher score is that the selected features cover the entire data space. The Fisher score is the highest score for a feature where the distance between data centers between different classes is large while it is short between data of one class [32]. Suppose our training example is as follows.

$$x_1, y_1 \ , \ldots, x_N \, , \, y_N \, , \ldots \qquad (1)$$

[DOI: 10.52547/ijict.13.3.39]

Where x∈ R^d and y∈ {+ 1, -1} and d is the sample dimensions and y_N is the class label. N is the number of training samples.

Also, N_1 is the number of positive samples (in the present case, the number of normal samples) and N_2 is the number of negative samples (in the present case, the number of attack samples), which will be used later. Fisher score is defined as follows:

$$F = \frac{S_b}{S_w} \quad (2)$$

Where S_b is the inter-class scattering matrix that describes the distance between two classes, and s_w is the intra-class scattering matrix that expresses the distance in a class.

S_b is defined as follows:

$$S_b = (\overline{m_1} - \overline{m})^2 + (\overline{m_2} - \overline{m})^2 \quad (3)$$

In this regard, $\overline{m_1}$ ،$\overline{m_2}$ and $\overline{m}$ است are the average of positive, negative and all classes, respectively.

$$\overline{m_1} = \frac{1}{N_1} \sum_{x\in+1} x \quad (4)$$

$$\overline{m_1} = \frac{1}{N_2} \sum_{x\in-1} x \quad (5)$$

$$\overline{m} = \frac{1}{N} \sum x \quad (6)$$

S_w is also defined as follows:

$$S_w = S_1 + S_2 \quad (7)$$

$$S_1 = \frac{1}{N_1} \sum_{x\in+1} (X - \overline{m_1})^2 = \partial_1 \quad (8)$$

$$S_2 = \frac{1}{N_2} \sum_{x\in-1} (X - \overline{m_2})^2 = \partial_2 \quad (9)$$

In the above relation ∂_1 and ∂_2 are the variance of positive and negative classes.

The values S_b and S_w can be written as follows

$$F = \frac{(\overline{m_1} - \overline{m})^2 + (\overline{m_2} - \overline{m})^2}{\sum_{x\in+1}(X - \overline{m_1})^2 + \sum_{x\in-1}(X - \overline{m_2})^2} \quad (10)$$

Therefore, the Fisher score for the r-th property is as follows:

$$F = \frac{(\overline{m_1}-\overline{m})^2 + (\overline{m_2}-\overline{m})^2}{\sum_{i=+1}^{2} \partial^{r2}} \quad (12)$$

In this research, Fisher score value was calculated for all NSLKDD dataset features using Fisher score calculation technic, and results of this arrangement are shown in table 3.

After arrangement, NSLKDD dataset features are available based on their importance. It means the first features of this arrangement are the most important ones among all and the last features are the least important of them. In this research, first to 25th features were selected by which the made SVM was trained and tested.

Fig. 2 shows the schematic of action process. NSLKDD dataset is divided into training and experimental data. The proper features were selected and are given to the suggested system to be trained and the trained machine is valuated using the experimental data. Finally, the obtained outputs were evaluated and the result is tested.

VI. Experimental Results

Three evaluation criterions were used in this design for training and experimental data [33]:

True positive rate shows the better efficiency as getting much closer to 1. That means the ratio of detected attack events correctly to total attack events.

False positive rate that is better as much closer to zero. That means the ratio of normal events detected as attacks to total considered events as normal.

Accuracy is a simple and straightforward measure of the quality of an algorithm. In this evaluation, the fraction face is the sum of the number of elements that have been correctly identified, and the denominator of the fraction is the sum of all events in all cases.

Precision is a measure that tells us what percentage of "True" in algorithms are correct.

$$Accuracy = \frac{(TP+TN)}{(TP+FP+FN+TN)} \quad (12)$$

$$Recall\ (TPR) = \frac{(TP)}{(TP+FN)} \quad (13)$$

$$Precision = \frac{(TP)}{(TP+Fp)} \quad (14)$$

$$FPR = \frac{(FP)}{(TN+FP)} \quad (15)$$

where,

True Negative (TN) = it detects normal data correctly.

True Positive (TP) = it detects attacks correctly.

[DOI: 10.52547/ijict.13.3.39]

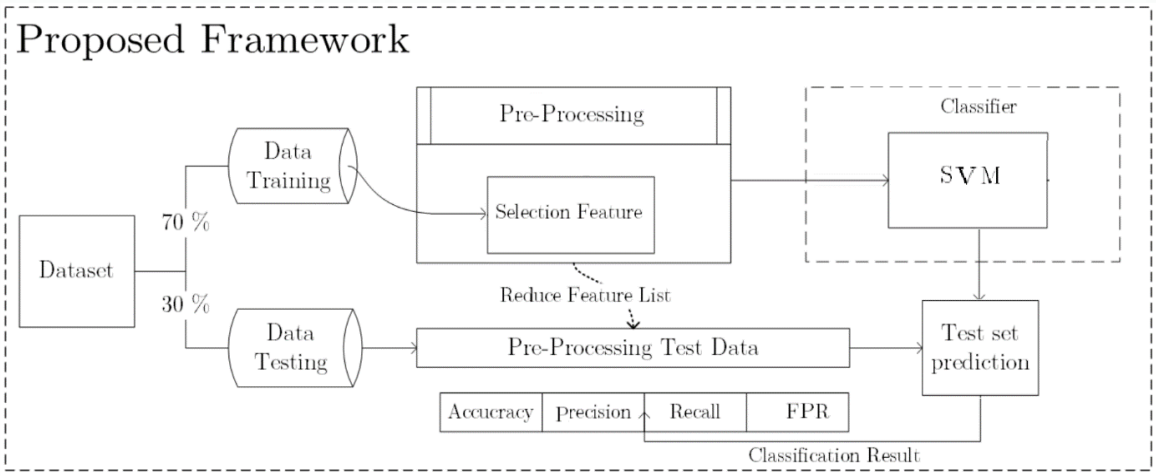

Figure 2. Showing the Action Process.

False Positive (FP) = normal events are known as attack.

False Negative (FN) = attack events are known as normal.

As it is observed in table 5 and 6, true positive rate is 96.05% in the designed SVM with the suggested feature selection (based on Fisher score). It was more acceptable number than 41, while it has very shorter implementation time than 41 features. Here we compare the proposed method with seven state of the art IDSs. As shown in Table 7, the proposed system has been compared with various other methods, and in some methods feature selection has been used. The classification results, show that the proposed IDS performs very well and shows a significant increase in TPR value. Although the FPR value is not the best one for our method, we should note again that TPR is a more important criterion than FPR for IDS tasks. Classifying a normal package as intrusion is an error that can be corrected by the operator in the next steps. However, classifying an intrusion as a normal package can have irreparable consequences for the system.

TABLE V. RESULTS OF DESIGNED SVM WITH TRAINING DATA

Number of Feature	Recall	Accuracy	FPR	Precision	Implementation time (s)
12	89.93	86.99	4.21	88.54	207
17	94.22	92.08	2.42	92.36	230
19	96.07	94.16	3.85	95.81	243
25	**97.05**	**93.87**	**2.88**	**95.14**	**271**
41	98.89	97.01	2.52	97.33	396

TABLE VI. RESULTS OF DESIGNED SVM WITH TEST DATA

Number of Feature	Recall	Accuracy	FPR	Precision	Implementation time (s)
12	86.67	82.78	4.99	85.34	87
17	89.76	85.34	3.35	89.22	101
19	92.38	87.98	4.23	91.87	122
25	**94.58**	**89.06**	**2.95**	**93.08**	**146**
41	95.34	90.23	2.90	94.43	260

[DOI: 10.52547/ijict.13.3.39]

TABLE VII. COMPARISON OF THE IDS WITH OTHER METHODS

Classifier	Feature Length	Feature Selection technique	TPR	FPR
K-means-NN[34]	41	-	93.83	9.88
Support Vector Machine [35]	5	mutual information concept +binary gravitational search algorithm	88.36	3.08
NBC-NBTree[36]	41	-	93.41	0.275
decision tree [37]	10	bee algorithm using membrane computing	89.11	1.76
LTMD[38]	41	-	93.32	0.06
multilayer SVM classifier [39]	12	hybrid kernel principal component analysis+ GA	94.22	2.87
Proposed IDS	**25**	**Fisher Score**	**94.58**	**2.95**

VII. CONCLUSION

In this research, it was tried to design an intrusion detection system based on a support vector machine and tried to increase the speed of the designed system by using feature selection, while true positive rate and false positive rate were at the desirable level.. The designed IDS were implemented by 12, 17, 19, 25, and 41 features, and their results were compared.

By selecting features, we are able to remove additional and irrelevant features, and therefore, we can avoid a problem called "dimension curse", and hence the accuracy of classification value is acceptable. The comparison of IDS with different features was examined and it was shown that feature selection has a positive effect on the speed of the detection system, while true positive rate is also acceptable.

REFERENCES

[1] D. M. Camacho, K. M. Collins, R. K. Powers, J. C. Costello, and J. J. Collins, "Next-Generation Machine Learning for Biological Networks," *Cell*, 2018.

[2] [2] S. Dilek, H. Çakır, and M. Aydın, "Applications of artificial intelligence techniques to combating cyber crimes: A review," *arXiv preprint arXiv:1502.03552*, 2015.

[3] M. Jabbar and R. Aluvalu, "RFAODE: A novel ensemble intrusion detection system," *Procedia computer science*, vol. 115, pp. 226-234, 2017.

[4] M. Jabbar, K. Srinivas, and S. S. S. Reddy, "A Novel Intelligent Ensemble Classifier for Network Intrusion Detection System," in *International Conference on Soft Computing and Pattern Recognition*, 2016: Springer, pp. 490-497.

[5] H. Yoo and T. Shon, "Novel approach for detecting network anomalies for substation automation based on IEC 61850," *Multimedia Tools and Applications*, vol. 74, no. 1, pp. 303-318, 2015.

[6] G. Agrawal, S. K. Soni, and C. Agrawal, "A SURVEY ON ATTACKS AND APPROACHES OF INTRUSION DETECTION SYSTEMS," *International Journal*, vol. 8, no. 8, 2017.

[7] J. Singh and M. J. Nene, "A survey on machine learning techniques for intrusion detection systems," *International Journal of Advanced Research in Computer Communication Engineering*, vol. 2, no. 11, pp. 4349-4355, 2013.

[8] V. Rodriguez-Galiano, M. Sanchez-Castillo, M. Chica-Olmo, and M. Chica-Rivas, "Machine learning predictive models for mineral prospectivity: An evaluation of neural networks, random forest, regression trees and support vector machines," *Ore Geology Reviews*, vol. 71, pp. 804-818, 2015.

[9] P. Aastha and N. Sharma, "A NOVEL TECHNIQUE FOR INTRUSION DETECTION SYSTEM FOR NETWORK SECURITY USING HYBRID SVM-CART," *International Journal of Engineering Development and Research*, vol. 5, no. 2, 2017.

[10] H. Elwahsh, M. Gamal, A. Salama, and I. M. El-Henawy, "A novel approach for classifying Manets attacks with a neutrosophic intelligent system based on genetic algorithm," *Security and Communication Networks*, vol. 2018, 2018.

[11] A. I. Madbouly and T. M. Barakat, "Enhanced relevant feature selection model for intrusion detection systems," *International Journal of Intelligent Engineering*, vol. 4, no. 1, pp. 21-45, 2016.

[12] L. Dong, J. Wesseloo, Y. Potvin, and X. Li, "Discrimination of mine seismic events and blasts using the fisher classifier, naive bayesian classifier and logistic regression," *Rock Mechanics Rock Engineering*, vol. 49, no. 1, pp. 183-211, 2016.

[13] C. A. Jensen, M. A. El-Sharkawi, and R. J. Marks, "Power system security assessment using neural networks: feature selection using Fisher discrimination," *IEEE Transactions on power systems*, vol. 16, no. 4, pp. 757-763, 2001.

[14] B. Ingre and A. Yadav, "Performance analysis of NSL-KDD dataset using ANN," in *Signal Processing And Communication Engineering Systems (SPACES), 2015 International Conference on*, 2015: IEEE, pp. 92-96.

[15] S. Aljawarneh, M. Aldwairi, and M. B. Yassein, "Anomaly-based intrusion detection system through feature selection analysis and building hybrid efficient model," *Journal of Computational Science*, 2017.

[16] Z. Xue-qin, G. Chun-hua, and L. Jia-jun, "Intrusion detection system based on feature selection and support vector machine," in *Communications and Networking in China, 2006. ChinaCom'06. First International Conference on*, 2006: IEEE, pp. 1-5.

[DOI: 10.52547/ijict.13.3.39]

[17] S. Chebrolu, A. Abraham, and J. P. Thomas, "Feature deduction and ensemble design of intrusion detection systems," *Computers & security*, vol. 24, no. 4, pp. 295-307, 2005.
[18] D. Koller and M. Sahami, "Toward optimal feature selection," Stanford InfoLab, 1996.
[19] [19] D. PEHRO and D. Stork, "Pattern classification," *D Wiley-Interscience Publication* 2001.
[20] M. Robnik-Šikonja and I. Kononenko, "Theoretical and empirical analysis of ReliefF and RReliefF," *Machine learning*, vol. 53, no. 1-2, pp. 23-69, 2003.
[21] X. He, D. Cai, and P. Niyogi, "Laplacian score for feature selection," in *Advances in neural information processing systems*, 2006, pp. 507-514.
[22] L. Song, A. Smola, A. Gretton, K. M. Borgwardt, and J. Bedo, "Supervised feature selection via dependence estimation," in *Proceedings of the 24th international conference on Machine learning*, 2007: ACM, pp. 823-830.
[23] F. Nie, S. Xiang, Y. Jia, C. Zhang, and S. Yan, "Trace ratio criterion for feature selection," in *AAAI*, 2008, vol. 2, pp. 671-676.
[24] M. M. Javidi and M. Hassan Nattaj, "A new and quick method to detect DoS attacks by neural networks.," *The Journal of mathematics and computer Science*, vol. 6, pp. 85-96, 2013.
[25] M. M. Javidi and M. Hassan Nattaj, "Network Attacks Detection by Hierarchical Neural Network," *Computer Engineering Applications Journal*, vol. 4, no. 2, p. 119, 2015.
[26] E. R. Hruschka Jr and N. F. Ebecken, "Towards efficient variables ordering for Bayesian networks classifier," *Data Knowledge Engineering*, vol. 63, no. 2, pp. 258-269, 2007.
[27] M. Hu, F. Shen, and J. Zhao, "Hidden Markov models based dynamic hand gesture recognition with incremental learning method," in *Neural Networks (IJCNN), 2014 International Joint Conference on*, 2014: IEEE, pp. 3108-3115.
[28] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, "A detailed analysis of the KDD CUP 99 data set," in *Computational Intelligence for Security and Defense Applications, 2009. CISDA 2009. IEEE Symposium on*, 2009: IEEE, pp. 1-6.
[29] F. Kuang, W. Xu, and S. Zhang, "A novel hybrid KPCA and SVM with GA model for intrusion detection," *Applied Soft Computing*, vol. 18, pp. 178-184, 2014/05/01/ 2014, doi: https://doi.org/10.1016/j.asoc.2014.01.028.
[30] S.-J. Horng *et al.*, "A novel intrusion detection system based on hierarchical clustering and support vector machines," *Expert Systems with Applications*, vol. 38, no. 1, pp. 306-313, 2011/01/01/ 2011, doi: https://doi.org/10.1016/j.eswa.2010.06.066.
[31] Q. Gu, Z. Li, and J. Han, "Generalized fisher score for feature selection," *arXiv preprint arXiv:*. 2012.
[32] Z. Xue-qin, G. Chun-hua, and L. Jia-jun, "Intrusion detection system based on feature selection and support vector machine," in *2006 first international conference on communications and networking in China*, 2006: IEEE, pp. 1-5.
[33] D. Gavrilis and E. Dermatas, "Real-time detection of distributed denial-of-service attacks using RBF networks and statistical features," *Computer Networks* vol. 48, no. 2, pp. 235-245, 2005.
[34] K. Faraoun and A. Boukelif, "Neural networks learning improvement using the K-means clustering algorithm to detect network intrusions," *INFOCOMP Journal of Computer Science*, vol. 5, no. 3, pp. 28-36, 2006.
[35] H. Bostani and M. Sheikhan, "Hybrid of binary gravitational search algorithm and mutual information for feature selection in intrusion detection systems," *Soft computing*, vol. 21, no. 9, pp. 2307-2324, 2017.
[36] N. Sharma and S. Mukherjee, "A novel multi-classifier layered approach to improve minority attack detection in IDS," *Procedia Technology*, vol. 6, pp. 913-921, 2012.
[37] K. I. Rufai, R. C. Muniyandi, and Z. A. Othman, "Improving bee algorithm based feature selection in intrusion detection system using membrane computing," *Journal of networks*, vol. 9, no. 3, p. 523, 2014.
[38] Y. Yuan, L. Huo, and D. Hogrefe, "Two layers multi-class detection method for network intrusion detection system," in *2017 IEEE Symposium on Computers and Communications (ISCC)*, 2017: IEEE, pp. 767-772.
[39] F.-J. Kuang and S.-Y. Zhang, "A Novel Network Intrusion Detection Based on Support Vector Machine and Tent Chaos Artificial Bee Colony Algorithm," J. Netw. Intell., vol. 2, no. 2, pp. 195-204, 2017.

Mohammad Hassan Nataj Solhdar received his B.Sc. degree in Computer Science from the University of Mazandaran, Mazandaran, Iran in 2010, and M.Sc. degree in Computer Science from the Shahid Bahonar University of Kerman, Iran in 2013. He is a faculty member of Shahid Chamran University of Ahvaz, Ahvaz, Iran. His research interests include Artificial Intelligent, Machine Learning, Neural Network, Computer Network Security, Intrusion Detection System. He has published several journal and conference papers in these fields.

[DOI: 10.52547/ijict.13.3.39]