Engineering and Technology Journal
Journal homepage: engtechjournal.org

Proposed Hybrid Classifier to Improve Network Intrusion Detection System using Data Mining Techniques

Sarah M. Shareef [a*], **Soukaena H. Hashim** [b]

[a] Production Engineering and Metallurgy Department University of Technology, Baghdad, Iraq. 70203@uotechnology.edu.iq

[b] Department of Computer Science, University of Technology, Baghdad, Iraq. 110015@uotechnology.edu.iq

*Corresponding author.

Submitted: 02/04/2019 Accepted: 11/05/2019 Published: 25/04/2020

K E Y W O R D S

=Data Mining, False Alarm, Network Intrusion Detection System, Naïve Bayes, Multinomial Logistic Regression.

A B S T R A C T

Network intrusion detection system (NIDS) is a software system which plays an important role to protect network system and can be used to monitor network activities to detect different kinds of attacks from normal behavior in network traffics. A false alarm is one of the most identified problems in relation to the intrusion detection system which can be a limiting factor for the performance and accuracy of the intrusion detection system. The proposed system involves mining techniques at two sequential levels, which are: at the first level Naïve Bayes algorithm is used to detect abnormal activity from normal behavior. The second level is the multinomial logistic regression algorithm of which is used to classify abnormal activity into main four attack types in addition to a normal class. To evaluate the proposed system, the KDDCUP99 dataset of the intrusion detection system was used and K-fold cross-validation was performed. The experimental results show that the performance of the proposed system is improved with less false alarm rate.

How to cite this article: S. M. Shareef and S. H. Hashim, "Proposed hybrid classifier to improve network intrusion detection system using data mining techniques," Engineering and Technology Journal, Vol. 38, Part B, No. ٠1, pp. ٦-14, 2020.
DOI: https://doi.org/10.30684/etj.v38i1B.149

1. Introduction

IDS is software that detects any activity that is normal or malicious, where this method is used to perform data security as a defense methodology of new cyber-attacks [1]. IDS is generating a number of false alarms and this problem has encouraged many researchers to find the solution to distinguish alerts to the less important incident and reduce false alarms which are false positive (FP) and false-negative (FN) [2].

IDS based on data mining technique can be used to enhance IDS in real time, remove the normal activity from alarm data for focusing real attacks and find abnormal activity that uncovers a real attack. It is the computational process of discovering patterns in data sets involving methods at the

intersection of artificial intelligence, machine learning, and database systems. Data mining applications can use different parameters to examine different data sets [3, 4].
Network Intrusion Detection Systems (NIDS) had become the most important component of recent network infrastructure due to the effects of increased security threats nowadays. Intrusion Detection System (IDS) is generating a good number of alarms however; it is deployed algorithmic procedures to reduce false positives [5]. Naive Bayesian Classifier is a simple probabilistic based on probability models that include independent assumptions. NB classifiers can be trained very efficiently in supervised learning. In many practical applications, parameter estimation for naive Bayesian models applies the method of maximum likelihood [6].
Multinomial Logistic Regression is a statistical process for representing the relationships among variables. Variables used to predict other variables are called predictor variables and sometimes called independent variables while the variables that are predicted called the response or dependent variable. The regression model is called a simple regression when there is only one predictor variable, whereas the regression model is called multiple regression if there is more than one predictor variable [7].
In his paper related work is presented in section 2. In section 3 the KDD99 dataset is described. In section 4 data pre-processing of datasets is explained, in section 5 feature selection is discussed, in section 6 the proposed system NIDS is discussed in detail. Section 7 presents the evaluation performance, in section 8 the experiments and results and finally the conclusion.

2. Related Work

A Survey is performed consisting the latest papers which carried out the training and testing based on Naive Bayesian and Multinomial Logistic Regression.
Keerthika and Priya have provided a proposal that focuses on naïve feature reduction in addition to the feature of selected methods such as gain ratio and information gain for reducing the redundant and irrelevant features. This proposal has used a naïve Bayes classifier for the design intrusion detection system which gives the best performance with the efficiency of the system [8].
Gupta et al. have presented NIDS to monitor the network or malicious activities and forbidden access to devices. NIDS is used to protect the data's features and integrity. The proposal has used NSL-KDD dataset to learn the manner of the attacks depending on the methods of data mining such as logistic regression and K-means clustering, hence it generates the rules to classify network activities. The results show that linear regression was very effective accuracy for detecting attacks which were 80% while, the K-means clustering showed 67% accuracy [9].
Belavagi and Munigal proposed a predictive and classification model for intrusion detection systems by using advanced machine learning algorithms such as logistic regression, naïve Bayes, support vector machine, and random forest classifiers. This proposal was evaluated with NSL-Kdd dataset. The experimental results show that the effective classifier was random forest which outperforms on other classifiers by comparing it performances like precision, recall, and accuracy and it has 99% accuracy to disentangle normal from abnormal data. However, logistic regression had 0.84% accuracy and naïve Bayes had 0.795 accuracies [10].
Manju has provided a proposal to analyze the network traffic into normal or abnormal by using probabilistic methods such as naïve Bayes and statistical methods like logistic regression classifiers. KDD99 benchmark dataset was used to perform the system. The results of the system show that naïve Bayes classifier had a prediction accuracy of 81.8% while the false positive was 19.2%, otherwise the logistic regression had a prediction accuracy of 87.6% and false-positive was 13.3%. The proposal has achieved high accuracy rate when identified the connections being normal or abnormal and lower false alarm rate with reducing number of features [11].

3. Dataset Description

Dataset is needed and used in information security research of which the network intrusion dataset from the KDD archive popularly is mentioned as KDDCup99 dataset, see table 1. The KDDcup99 dataset has been considered as point attraction by many researchers in the domain of intrusion detection systems. It is most widely used for evaluating IDS [12]. The 10% of KDDCup99 dataset is the original dataset that was approximately 494,020 records and each of which includes 41 features and was categorized as either normal or abnormal activity with exactly one fixed attack type.

Table 1: Number of model kddcup99 Datasets

Dataset of KDD	Total KDD	Corrected KDD	10% KDD
Total	4.898.430	311.029	494.020
Dos	3.883.370	229.853	391.458
R2L	1.126	16.347	1.126
U2R	52	70	52
Probe	41.102	4.166	4.107
Normal	972.780	60.593	97.277

4. Dataset Preprocessing

Data preprocessing is the main and essential stage to obtain final datasets that can be considered as a correct and beneficial for further data mining algorithms. The output after a reliable connection of data preprocessing processes is a final data set which can be useful for further DM algorithms [13]. Transformation data is the process that transforms the nominal values into numeric values so that it is convenient input for classification and improvement performance of the system, therefore to improve the fitness of data and to make it more expressing for mining, the data preprocessing should be done [14]. In order to apply data transformation, data sets will be converted from symbolic values to numeric values. For example, protocol type like tcp is assigned to 1, udp is assigned to 2 and icmp is assigned to 3 and so on for the rest [15].

5. Feature Selection Technique

Feature selection technique is considered as one of the important processes in data preprocessing for IDS which is used to remove the redundant and irrelevant feature as much as possible by using entropy as feature selection algorithms.

6. Training and Testing of the Proposed System

System analysis consists of two phases which are training and testing the proposed system. In the first phase, the proposed system will be implemented (Naïve Bayes, Multinomial logistic regression) classifiers on KDD Cup 99 dataset to detect the types of attacks in offline networks. The proposal utilizes number of kcross-validation to split the data into k equal times and applied the holdout method to divide the data randomly into two parts which are training and testing. In the second testing phase, is used to evaluate the performance of the system by using k fold of number times.

I. Naïve Bayes classifier

NB classifier is a probabilistic classifier based on Bayes theorem with an independence assumption. It is supervised learning and will be trained very efficiently; Let S be a training set of records. Each record X is explained by an n-dimensional attribute vector of feature values $[a_1,a_2,\ldots,a_n]$, Assume that there are m classes, C1, C2,…….Cm, the classifier will predict the record X belongs to the class which having highest posterior probability conditioned on X. Thus, the naïve Bayesian classifier predicts the record X belongs to the class Ci if and only if [16]:

P (Ci | X), i Є [1,m] > P(Cj | X),

for $1 \leq j \leq m, j \neq i$ then

$$c(X) = \arg\max_{c \in C} P(c) * P(X|c) \quad (1)$$

Where:

P(Ci | X) is the probability of feature to fall in class Ci

P (Ci) = the prior probability for class Ci.

C(X) is maximum posterior used to assign the class c having maximum p (X|c).

The class conditional independence is explained as this equation:

$$P(X|Ci) = P(a_1, a_2, \ldots, a_n|c) = \prod_{j=1}^{n} P(a_j|ci) \quad (2)$$

By simplifying the calculation of P(C) and $P(a_i|c)$, NB classifier was easily built.

Algorithm 1 : Naïve Bayes algorithm	
Input: Transformation kddcup99 (training and testing) dataset type of network traffic with normal & abnormal	**Output:** classification

Begin
#Training data
Step1: compute the prior probability for each class p(c).
Step2: For every attribute Compute frequency for each value V_j with class c1,c2
if freq.of V_j with c1=0 or freq.V_j with c2=0 Then compute the probability of each value within classes for all attributes + 1.
Else
Compute the probability of each value within classes for all attributes.
End for
# Testing data 	
Step3: Compute for every record in testing data :
1- the probability for each value V_j together with class c in training data
2- multiply probability of every value as Eq.(2)
Step4: To classify the record, Find the maximum value by multiply the outcome of Eq. (2) together with the probability of each class as Eq. (1)
End.	

II. Multinomial Logistic Regression Classifier

Multinomial Logistic Regression, also known as SoftMax Regression is a supervised learning algorithm that generalizes the logistic regression to classify problems which the output can take more than two possible values. Multinomial Logistic Regression requires more time to be trained comparing to Naive Bayes because it uses an iterative algorithm to estimate the parameters of the model. Gradient descent is a way to minimize an objective function J (w) parameterized by a model's parameters w. To reduce the error and updated weights from the beginning, it will compute the net input Z by multiplying inputs with weights in Eq. (3):

$$\mathrm{Z} = w_0x_0 + w_1x_1 + \cdots .. w_mx_m = \sum_{i=0}^{m} w1x1 = w^Tx \quad (3)$$

Where: W_0 is the bias unit, W is weight vector, x presents the feature vector of the training sample. The softmax function will calculate the probabilities of each class label as can be seen in Eq.(4):

$$\mathrm{p}(\mathrm{y} = \mathrm{j}|zi) = \varphi\, softmax\,(zi) = \frac{e^{zi}}{\sum_{j=0}^{k} e^{zi}} \quad (4)$$

where: $\mathbf{p}(\mathbf{y} = \mathbf{j}|\boldsymbol{zi})$ the probability of each class label j=1,2,…k with net input (zi). $\boldsymbol{e^{zi}}$: the exponential of net input zi. To minimize the error output by using a gradient descent algorithm, first it will need to define the cross-entropy function as in Eq. (5):

$$\mathrm{H}(T_i, O_i) = -\sum_m T_i . log(O_i) \quad (5)$$

Where:

$\mathrm{H}(T_i, O_i)$: The distance between the softmax probabilities and the encoding matrix. $\boldsymbol{T_i}$: the value of target output, $\boldsymbol{O_i}$: the value of actual output. The second will need to define the cost function j to minimize the error as Eq. (6):

$$J(W) = \frac{1}{n}\sum_{i=0}^{n} H(T_i, O_i) \quad (6)$$

Where: n the samples of the training set. Then, the iterative algorithm of gradient descent requires estimating the partial derivative of the cost function as following Eq. (7):

$$\nabla w_j J(W) = -\frac{1}{n}\sum_{i=0}^{n}[X^i(T_i, O_i)] \quad (7)$$

The weights update in the opposite direction of the gradient of the cost function as the following Eq. (8):

$$w_j \coloneqq w_j - \alpha \nabla w_j J(W) \quad (8)$$

Where: $\boldsymbol{w_j}$ is the weight vector for each class j, $\boldsymbol{\alpha}$ is learning rate.

See Algorithm 2 which displays the multinomial classifier.

Algorithm 2: Multinomial Logistic Regression Algorithm
Input: Normal and Abnormal network traffic **Output:** classification type of attack into four class with normal activity by MLR classifier
Begin #Training data **Step1:** Initialized the weights vectors randomly from input layer to output layer. **Step2:** Every feature in the sample represents input unit (X_i). **Step3:** Compute the net input Z of each unit by multiply the Inputs with weights as the following Eq. (3). **Step4:** Computes the probability that this training sample x belongs to class j given the net input z by using softmax function as Eq.(4) **Step5:** apply the argmax-index position for each row of training set to find the highest likelihood then assigned it to class label which is encoded by one-hot encoding. **Step6:** apply cross-entropy function which is the negative log probability of the right answer as calculated in Eq. (5). **Step7:** minimize the error by using a cost function, if the value of cost function is low then will predict the target classes, if the value of cost function is high then an error occurred will computed as Eq. (6). **Step8:** using the cost derivative to simplify the process as Eq.(7) and update the weights in opposite direction of the Gradient as Eq. (8). **Step9:** Repeat until cost function value is less. **End** #Testing data **Step1:** propagate the input unit (X_i) which represents by the Feature in sample into the neurons. **Step2:** Compute the net input Z of each unit as the following Eq. (3). **Step3:** Computes the probability by using softmax function as Eq. (4). **Step4:** apply the argmax-index position for each row to find The highest likelihood then assigned it to class label which is Encodes by one-hot encoding. End.

7. Performance Measures

The performance measure of IDS can be estimated by using various algorithms based on the following standard performance measures [17]:

- True positive (TP): Number of attacks is correctly identified attack event.
- True negative (TN): Number of normal is correctly identified normal event.
- False positive (FP): Number of normal is incorrectly identified attack event.
- False negative (FN): Number of attack is incorrectly identified normal event where a detector fails to detect the attack because the virus is new and no signature is yet available. Confusion matrix displayed in Table 2.
- Detection Rate (DR): It is the proportion of correctly classified positive examples splitted by the total number of examples that are classified as

Positive, as the following equation:

$$DR = \frac{TP}{TP+FN} \quad (9)$$

- Accuracy (Acc): It is the proportion of the rate patterns which are detected as normal or intrusion activity divided by the overall patterns, as shown in the following equation**:**

$$ACC = \frac{TP+TN}{TP+TN+FP+FN} \quad (10)$$

- Precision of a classifier is the proportion of positive predictions made by the classifier that is true, as in the following equation:

$$Precision = \frac{TP}{TP+FP} \quad (11)$$

- False Alarm Rate (FAR): It is the proportion of the rate samples which are incorrectly identified as attack to the overall samples of normal behavior as in the following equation:

$$FAR = \frac{FP}{TN+FP} \quad (12)$$

- F-measure: It is defined as the symmetric mean of recall and precision according to the following equation:

$$F - measure = \frac{2*Recall*Precision}{Recall+Precision} \quad (13)$$

- Specificity (SPC): It is also called as true negative rate (TNR) which awards an indication of the normal behavior that is specified correctly. Thus, it gives the probability that the algorithm can predict normal instances correctly.

$$Specificity = \frac{TN}{FP+TN} \quad (14)$$

8. Experimental Results

Two experiments will be implemented on the selected samples which were (329,510 records) of training and (164,510 records) of testing dataset. First experiment will be implemented on the first level based on Naïve Bayes algorithm was applied to detect the normal and attack traffic with 41 features of the dataset then applying this algorithm with 30 and 15 features of the dataset that were considered as the lowest entropy. Second experiment will be implemented on the second level based on Multinomial logistic regression algorithm was applied to detect five classes (four main attack types Dos, Probe, U2r, R2l, in addition to normal class) with 41 features of the dataset then applying this algorithm with 30 and 15 features of the dataset that were considered as the lowest entropy. There are several parameters that impact the performance of multinomial logistic regression which are the (w) parameters estimated to minimize the cost function, the value of learning rate=0.9 and the maximum number of iteration up to 100 which have an impact on the convergence the output of multinomial logistic with target output. The result of the first level was compared and evaluated with evaluation metric DR, Accuracy, precision, FAR, F-measure, specificity for 41 feature with k=1, k=2 and k=3 as shown in Table 3.In the first level, the calculation of time including training and testing data, where the time was recalculated more than once (three times as shown in Figure 1.

The experiments showed that after examining the results of the data, it was found that the accuracy in the first level selecting 41 features has the best results than the second level selecting 41 features, While in the second level using 30 features gave best results than the first level when selecting 30 features. Finally, by selecting 15 features, it is found that the first and second levels had the best result as displayed in Tables 3 and 4 respectively. In the first level, the proposed system showed best result in false alarm rate using 41 and 30 features, while when using 15 feature showed the lowest result of error rate, in the second level, the system showed best result when using 41 and 30 features of error rate while using 15 feature showed the lowest results of error rate. The performance of the system using 41 features gave the best result in all evaluation metrics of FAR, Accuracy, Recall, Precision, F-measure, Specificity than using 30 and 15 features. Table 5 displays the comparison of the experimental result between the prior studies and the proposed system.

Table 2: confusion matrix

Actual	Class	Predicted Class	
		Negative Class	Positive Class (Attack)
Normal	True negative (TN)	False positive (FP)	
Attack	False negative (FN)	True positive	

Table 3: Evaluation metric of first classifier

Classifier algorithm	No of feature	Dataset testing with k-fold	DR	Accuracy	Precision	FAR	F-measure	Specificity
0.98 Naïve Bayes	41	K1	0.99	0.98	0.99	0.01	0.99	
		K2	0.99	0.98	0.99	0.02	0.99	0.97
		K3	0.80	0.80	0.80	0.9	0.80	0.04
	30	K1	0.99	0.98	0.99	0.02	0.99	0.97
		K2	0.99	0.98	0.99	0.01	0.99	0.98
		K3	0.80	0.80	0.80	0.9	0.80	0.04
	15	K1	0.99	0.98	0.99	0.02	0.99	0.97
		K2	0.96	0.96	0.96	0.01	0.96	0.87
		K3	0.80	0.80	0.80	0.9	0.80	0.04

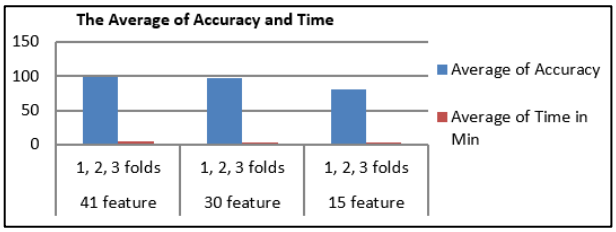

Figure 1: The average of accuracy and time

Table 4: Evaluation metric of second classifier

Classifier algorithm	Type of class	No of feature	Dataset with k-fold	DR	Accuracy	Precision	FAR	F-measure
Multinomial logistic regression	Dos	41	K=1	0.99	0.99	0.99	0.0005	0.99
			K=2	0.99	0.96	0.95	0.1	0.97
			K=3	0.99	0.95	0.95	0.1	0.97
		30	K=1	100	0.99	0.99	0.00002	0.99
			K=2	100	0.99	0.99	0.0001	0.99
			K=3	100	100	100	0	100
		15	K=1	0.99	0.81	0.81	0.8	0.89
			K=2	100	0.80	0.80	0.9	0.89
			K=3	0.99	0.82	0.81	0.8	0.89
	Normal	41	K=1	0.99	0.99	0.99	0.00007	0.99
			K=2	0.85	0.96	0.98	0.003	0.91
			K=3	0.84	0.96	0.96	0.007	0.90
		30	K=1	100	0.99	0.99	0.000007	0.99
			K=2	0.99	0.99	0.99	0.000007	0.99
			K=3	100	100	100	0	100
		15	K=1	0.60	0.82	0.96	0.001	0.70
			K=2	0.64	0.81	0.97	0.0004	0.76
			K=3	0.66	0.82	0.97	0.001	0.74
	Probe	41	K=1	0.97	0.99	0.97	0.0001	0.97
			K=2	0.80	0.99	0.94	0	0.96
			K=3	0.80	0.99	0.93	0	0.96
		30	K=1	0.99	0.99	100	0	0.99
			K=2	100	100	100	0	100
			K=3	100	100	100	0	100
		15	K=1	0.98	0.98	100	0	0.70
			K=2	0/96	0.99	0.93	0.000007	0.77
			K=3	0.92	0.99	0.90	0.00001	0.80
	R 2 L	41	K=1	0.95	0.99	0.97	0.00006	0.96
			K=2	0.81	0.99	0.75	0	0.77
			K=3	0.83	0.99	0.79	0	0.70
		30	K=1	100	0.99	100	0	0.99
			K=2	0.99	0.99	0.99	0.00006	0.99
			K=3	100	100	100	0	100
		15	K=1	0.60	0.99	0.74	0.0002	0.72
			K=2	0.66	0.99	085	0.0001	0.75
			K=3	0.68	0.99	0.87	0.0001	0.72
	U 2 R	41	K=1	0.77	0.99	0.60	0.0001	0.70
			K=2	0.74	0.99	0.68	0	0.73
			K=3	0.72	0.99	0.70	0	0.80
		30	K=1	0.99	100	100	0	100
			K=2	0.76	0.99	0.94	0.00006	0.84
			K=3	100	100	100	0	100
		15	K=1	0.60	0.99	0.63	0.00002	0.60
			K=2	0.66	0.99	100	0	0.66
			K=3	0.65	0.99	0.75	0.00002	0.70

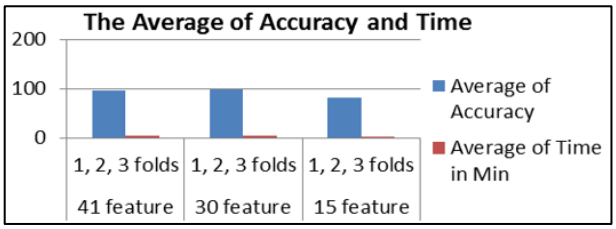

Figure 2: The Average of Accuracy and Time

Table 5: Comparison between the Prior Studies and Proposed System

Reference	Reference name	Method used	Accuracy rate	The accuracy of proposed system	
				Level 1	Level 2
[11]	Performance Evaluation of Intrusion Detection System Using Classification Algorithms	Logistic regression	87.7%	-	97%
		Naïve bayes	81.8%	98%	-
[9]	Network Intrusion Detection System Using various data mining techniques	Logistic regression	80%		97%
[10]	Performance Evaluation of Supervised Machine Learning Algorithms for Intrusion Detection	Logistic regression	84%		97%
		Naïve bayes	79%	98%	

9. Conclusions

Depending on the results of the proposed system, there are several conclusions can be suggested as follows:

1- The use of supervised machine learning classifiers such as Naive Bayes, Multinomial Logistic Regression gives high efficiency and accuracy for the proposed system.

2- Using cross-validation technique estimates and compares the performance of different algorithms and finds the best one from available data. Since its very large dataset will be applied cross-validation to avoid falling into overfitting.

3- The proposed system has proved that the false alarm rate was decreased gradually from the first level to the two levels as shown in Tables (3) and (4).

4- Multinomial Logistic Regression classifier takes more time in training and testing data than NB classifier.

References

[1] Y. R. Mukund, S. S. Nayak and K. Chandrasekaran, "Improving false alarm rate in intrusion detection systems using Hadoop," Conf. on Advance in Computing, Communications and Informatics (ICACCI), India, Jaipur, sept.21-24, 2016.

[2] N. Gupta, K. Srivastava and A. Sharma, "Reducing false positive in intrusion detection system," (IJCSIT) Int. Journal of Computer Science and Info. Technologies, vol.7 (3)1600-1603, ISSN: 0975-9646, 2016.

[3] J. A. Khan and N. Jain, "A survey on intrusion detection systems and classification techniques," IJSRSET, vol 2, Issue 5, print ISSN: 2395-1990, online ISSN: 2394-4099, 2016.

[4] S. Singh and M. Bansal, "Improvement of intrusion detection system in data mining using neural network," Int. journal of Advanced Research in Computer Science and software Eng., vol 3, ISSN: 2277 128X, Issue 9, 2013.

[5] A. Islam and M. Islam, "A novel signature based traffic classification engine reduce false alarms in intrusion detection systems," Int. Journal of Computer Networks and Communications (IJCNC) , vol 7, No.1, 2015.

[6] B. B Gupta, R. C. Joshi and M. Misra, "Estimating strength of DDos attack using various regression models," Int. Journal of Multimedia Intelligence and security, vol 1, No. 4, 2010.

[7] J. Soni and D. Xaxa, "An improved naïve bayes classifier for intrusion detection system," (IJIACS) Int. Journal of Innovations and advancement in computer science, Vol 5, ISSN: 2347-8616, Issue 6, 2016.

[8] G. Keerthika and D. S. Priya, "Feature subset evaluation and classification using naïve Bayes classifier," (JNCET) Journal of Network Communications and Emerging Technologies, vol 1, Issue 1, 2015.

[9] D. Gupta, S. Singhal, S. Malik and A. Singh, "Network intrusion detection system using various data mining techniques," Int. conf. on Research advances integrated navigation system (RAINS-2016), 2016.

[10] M. C. Belavagi and B. Muniyal, "Performance evaluation of supervised machine learning algorithms for intrusion detection," (IMCIP) Int. Multi-conf. on Information Processing-2016, Elsevier, vol 89, pages 117-123, 2016.

[11] C. Manju, "Performance evaluation of intrusion detection system using classification algorithms," Int. Journal of Innovative Research in Science, Eng. and Tech., vol 6, ISSN (online): 2319-8753, ISSN (print): 2347-6710, , Issue 7, July 2017.Available: http://www.ijirset.com.

[12] M. K. Siddiqui and Sh. Naahid, "Analysis of Kdd cup 99 dataset using clustering based data mining," Int. Journal of database theory and application, vol.6, pp.23-34, No.5, 2013.

[13] I. A. Abdulminem and S. H. Hashim, "A proposal to detect computer worms (malicious codes) using data mining classification algorithms," Eng. and Tech. Journal, Vol 31, No 2, 2013.

[14] V. D. Katkav and D. S. Bhatia, "Lightweight approach for detection of denial of service attacks using numeric to binary preprocessing,"(CSCITA) Int. Conf. on Circuits, Systems, Communication and Info. Tech. Application, 2014.

[15] K. Goeschel, "Reducing false positives in intrusion detection systems using data-mining techniques utilizing support vector machines, decision trees and naïve Bayes for off-line analysis," Int. Conf. on 30 March-3 April 2016, USA, 7506774, July 2016.

[16] S. M. Shareef and S. H. Hashim, "Intrusion detection system based on data mining techniques to reduce false alarm rate," Eng. and Tech. Journal, Vol. 36, Part B, No. 2, 2018.

[17] Y. Wahba, E. Elsalamouny and G. El Taweel, May, "Improving the performance of multi-class intrusion detection systems using features reduction," (IJCSI) Int. Journal of computer science Issues, Vol 12, Issue 3, ISSN (print):1694-0814, ISSN (online):1694-0784, 2015. Available: http://www.ijcsi.org.