Volume : 4 | Issue : 3 | March 2015 • ISSN No 2277 - 8179

Research Paper

The Novel K-Nearest Neighbor and Back Propagation Recursive Least Squares Algorithms for Intrusion Detection Systems

Mathematics

KEYWORDS : Intrusion Detection System, Neural Network, Back Propagation Recursive Least Squares Algorithm, Novel k-Nearest Neighbor Algorithm.

Dr. Sadeq Al-Hamouz University of Middle East, Jordan

Prof.Dr. Reyadh Naoum University of Middle East, Jordan

Omar Al-Sadoon University of Middle East, Jordan

ABSTRACT

This research is focusing on the most important security aspect which is the intrusion detection systems (IDS). The system is designed to detect and classify intrusions to four basic attacks that every computer or network is most likely to be exposed to. The design is based on Novel K-Nearest Neighbor (K-NN) machine learning algorithm , and on multilayer neural network(MLP) which was trained using back propagation recursive least squares algorithm(BPRLS). Several stages are included within the proposed system, first stage includes data environment, which represents NSL-KDD99 dataset for intrusion detection, and the stage of data codification, and the stage of data preprocessing and categories classification stage using Novel K-Nearest Neighbor (K-NN) machine learning algorithm, and the Neural Network training stage using Back Propagation Recursive Least Squares (BPRLS) Algorithm. The proposed intrusion detection scheme performance was evaluated using two hidden layers with two different numbers of neurons each time. The results confirmed the effectiveness of the proposed scheme in terms of the accuracy, precision, TPR (recall), TNR (specificity), FPR(false positive rate), FNR(false negative rate) and MSE (mean square error). Back propagation recursive least squares algorithm (BPRLS) recorded a detection rate of 0.959726 with false negative of 0.0402742 and mean square error of 0.2213 during experiment. The proposed intrusion detection system "results" was compared to other intrusion detection systems "results" which were designed using supervised learning such as enhanced resilient back propagation or unsupervised learning such as Kohenon.

Introduction

When the 21st century introduced networks, specifically networks internet, the world benefited from its unlimited chances but along with infinite challenges, for instance, a normal running of a network was able to bring great business, health and educational services developments in addition to new route of financial incomes to the society by initiating a genuine electronic businesses

And labor market, yet, the same network was also able to cause unexpected disasters due to insufficient security.

In 2013,(Chakraborty,2013) defined the intrusion as an unauthorized entry to another's property or area, while from computer sciences perspective it is known as the activities settle the basic computer network security objectives versus to confidentiality, integrity, and availability. The Intrusion Detection was also elucidated as the task of detecting, preventing and possibly reacting to the attack in a network based computer systems (Poojitha et al., 2010).

Practically, the process of Intrusion Detection may be recognizes as the method of monitoring and analyzing the events inside the network or the computer system for any possible threat indication of threats for practices of computer security, as well as observing the acceptable usage policies or standard security policies ,(Chakraborty,2013).

Intrusion Detection System (IDS) can be considered as second gate for security following the firewall. The (IDS) can be exploited to detect an assortment of intrusion behaviors and also to activate the intercept and the dynamic reactor to the vicious intrusion before any jeopardization operation occur to the network system, (Yichun et al. 2012).

Research Scope

The aim from this research is to design an anomaly intrusion detection system (IDS) which is able to distinguish between usual and unusual network traffic, and classify them according to their main type. The proposed system will be a combination of machine learning algorithms of Novel K-nearest neighbor(K-NN) and neural network which are going to be trained using Back Propagation Recursive Least Squares (BPRLS)Algorithm.

Related Works

The performance for a method of unsupervised learning for irregularity detection was evaluated by ,Riad et al. (2013) using K-means algorithm with the KDD Cup 1999 network dataset. There were 22 types of sub attacks have been founded in the dataset organized in four major classes (Probe, Dos, U2R and R2L). The final results showed a detection rate of 0.9766% for type (DoS) attacks, with false alarm rate of about 0.003%, on the other hand the detection rate of type (Prob) attacks was 0.0659 % with false alarm of 0.013%, and type (U2R) attacks detection rate was 0.00061 % with false alarm about 0.0004%, finally the detection rate of type (R2L) attacks was 0.381 % with false alarm about 0.0022%.

In 2007, Pervez et al (2007) presented a comparative study on different architecture types of neural network for intrusion detection system. Their back propagation neural network system works in tow main steps which include data collection, and training. Self-Organizing Maps (SOMs) were used along with the unsupervised learning to identify anomalies. The results showed that back propagation neural network system has a detection rate of 96% with false alarm of 6%, which are better results than the self-organizing maps. However, the researchers here didn't mention the number of layers or neurons they have used.

Wang and Ma (2009) were able to decrease the number-based traffic features as well as the host-based traffic features as an input to the (ANN) for anomaly intrusion detection scheme. They used the KDD99 to perform the evaluatation of their system by employing (Resilient Back Propagation) neural networks with different input numbers. (LogSig) was also used as transfer function, and their results showed that upon using 9 basic features as an input with architecture of (ANN) as 9-17-1 the detection rate was 85.7% with a training time of 126.753 seconds, but upon using 18 basic features+ time based as an input with architecture of (ANN) as 18-36-1, the detection rate was 88.4% with a training time of 202.562 seconds, yet upon using 22 basic features+ content as an input with architecture of (ANN) as 22-45-1 the detection rate was 85.3% with a training time of 278.684 seconds, whereby using 28 basic features+ time based+ host-based as an input with architecture of (ANN) as 28-60-1 the detection rate was 86.1% with a training time of 369.542

seconds, while using 31 basic features+ content+ time based as an input with architecture of (ANN) as 31-65-1 yielded a detection rate of 86.9% with a training time of 409.919 seconds, and upon using 41 basic features+ content+ time based +host-based as an input with architecture of (ANN) as 41-85-1 the detection rate was 89.6% with a training time of 537.843 seconds. From all the above we conclude that the best result achieved when using an input number 18 to the neural network where the detection rate was 88% with a training time of 200 seconds and the mean square error about 0.000573.

An irregularity intrusion detection scheme through employing the "Redial Bases Function (RBF)", which acts as a feed forward neural network considering single hidden layer and 31 neurons was offered by Bi et al. (2009). Researches employed the data from (KDDcup'99) to train their proposed method. They converted the string type in the records into numeric type before applying the data into the (RBF) neural network, and then deleted the similar numeric values columns in the records to achieve a faster computation and convergence. Their experiment showed that the performance of (RBF) network could not be improved with increasing the hidden layer nodes. Only by selecting the proper hidden layer the detection rate was 87% at nodes 31.

In order to classify the type of attacks which were found in the KDD99 data set, Mukhopadhyay et al. (2011) designed an anomaly based intrusion detection system. They have learned the neural network with 5000 samples (record) and later they took 41 attributes as an input to the neural network. Experiment parameters were, the number of the input layers nodes which was 41, and the number of the hidden layers which was 1 with 12 neurons, as well as a training algorithm that was based on conjugate gradient. The performance measure was the mean square error. Results showed that the detection rate was 95.6% with false alarm rate of 4.4%, and the mean squares error of 0.0088598 at epoch 237.

Further, Norouzian et al. (2011) was able to design a system for network intrusion detection purposes based on (ANN) using "Multi-Layer Perceptron (MLP)", in order to detects the attacks and classify them in 6 different groups (Smurf, Teardrop, Satan, Guest, Warezclient ,Buffer overflow). They employed the 10% of the KDD99 data set to evaluate the performance of their system, and also considered 13 features from each record, while (MLP) trained was trained using back propagation algorithm with three hidden layers. The results showed a detection rate of 90.78%.

AL-Rashdan et al. (2010) illustrated their work of intrusion detection model based on hybrid neural network (hopfeild+ kohonen with conscience function) and support vector machine(SVM) , by employing the (KDD cup99) data set to evaluate the performance of their hybrid model. Their model consisted of three stages (K-.medoid) was used for clustering, while in the second stage, they used Hopfield network and Kohonen (SOM). At stage three which represents the testing stage, (SVM) was used as a testing unit. The Results showed a detection rate of 92.5% and false alarm of 3.5%. The main advantage of this system is to combine the supervised and the unsupervised learning.

By using a learning vector quantization (LVQ) and enhanced resilient back propagation (ERBP) of(ANN) ; Naoum and Alsultani. (2013) presented other hybrid intrusion detection system model. In this system the supervised "Learning Vector Quantization (LVQ)" as the first phase of classification was trained to detect intrusions. It consists of 41 input nodes and 23 hidden nodes according to the sub attacks numbers and one output node, and the mean square error were used as performance measures. Later, the results of the (LVQ) will be combined with the results of the neural network which was trained by using the enhanced resilient back propagation (ERBP) classifier to provide maximum classification rate. A second stage of classification includes a multilayer perceptron(MLP) as supervised method which trained using an enhanced resilient back propagation(ERBP) training algorithm with 41 input nodes and one hidden layer as well as 32 neurons; the number of reached epochs was 364. The network performance was the mean square error (MSE). The performance and the (ERBP) convergence speed were tried to be enhanced through deriving the optimal value for the learning factor. NSL-KDD99 dataset was employed during the evaluation. Results showing that 97.06 detection rate were obtained with 2% false negative. One of the main issues in this experiment is the low-frequent attacks, where (U2R) have the lowest detection rate about 70.3% among other classes. Due to the low-frequent of attacks compared to the high-frequent ones, and the leaning small sample size, it becomes not easy for the (ANN) to learn the characters of the attacks thus leading to a lower detection precision.

Naoum et al. (2012) employed (K-NN) and (ERBP) of (ANN) to implement a new hybrid system for intrusion detection. The performance and the (ERBP) convergence speed were tried to be enhanced through deriving the optimal value for the learning factor. First Norm was used in the k-nearest neighbor was implementation instead of Euclidean distance, in fact they have used the first nearest neighbor (k) equals 1. The enhanced resilient back propagation neural network was trained using an optimal number for the neurons and hidden layers; therefore it was trained only with single hidden layer and 34 hidden neurons. The evaluation was performed on the NSL-KDD99 anomaly intrusion detection dataset. The proposed system had a classification rate (5 classes) of 97.2% with false negative rate of about 1%.

Research Methodology

The proposed system contain two algorithms for detecting intrusion, using machine learning algorithms of Novel K-nearest neighbor(K-NN), and neural network which are going to be trained using back propagation recursive least squares(BPRLS) algorithm. The following figure illustrates the methodology for (BPRLS) and Novel (K-NN) algorithms for intrusion detection. The first step is to provide the connection records of the NSL KDD99 data set from http://nsl.cs.unb.ca/NSL-KDD/ sit, each connection record consist of 41 futures. Kayacik et al,(2005) Represented the features name for each connection record with brief description for each future and the type of value corresponding to that feature in the NSL KDD99 data set. The Columns 2, 3, 4 and 42 for each connection records are in string format, and the rest columns are in numerical format, codification can be done by converting each string feature type founded in the records which are protocol type (column-2), service (column-3), flag (column-4) and label (column-42) to numeric values, this can be done by using custom mapping table for each string attribute type as illustrated below in the following tables.

Table 1: protocol column mapping table

Protocol name	Protocol number after transformation
Tcp	2
Udp	3
Icmp	4

Table 2: flag column mapping table

Flag name	Flag number after transformation
'SF'	4
'REJ'	5
'S0'	6

'RSTO'	7
'RSTR'	8
'SH'	9
'S3'	10
'S1'	11
'RSTOS0'	12
'S2'	13
'OTH'	14

Table 3: service column mapping table

Service name	Service number after transformation	Service name	Service number after transformation
'http'	16	'login'	60
'private'	17	'kshell'	61
'gopher'	18	'sql_net'	62
'telnet'	19	'time'	63
'ftp_data'	20	'hostnames'	64
'other'	21	'exec'	65
'remote_job'	22	'ntp_u'	66
'eco_i'	23	'nntp'	67
'smtp'	24	'ctf'	68
'ftp'	25	'daytime'	69
'ldap'	26	'shell'	70
'pop_3'	27	'IRC'	71
'courier'	28	'pop_2'	72
'discard'	29	'printer'	73
'ecr_i'	30	'tim_i'	74
'imap4'	31	'pm_dump'	75
'domain_u'	32	'red_i'	76
'mtp'	33	'netbios_ssn'	77
'systat'	34	'rje'	78
'iso_tsap'	35	'X11'	79
'csnet_ns'	36		
'finger'	37		
'uucp'	38		
'whois'	39		
'nnsp'	40		
'netbios_ns'	41		
'domain'	42		
'ssh'	43		
'netstat'	44		
'name'	45		
'supdup'	46		
'uucp_path'	47		
'Z39_50'	48		
'netbios_dgm'	49		
'urp_i'	50		
'auth'	51		
'bgp'	52		
'vmnet'	53		
'http_443'	54		
'efs'	55		
'echo'	56		
'klogin'	57		
'link'	58		
'sunrpc'	59		

Table 4: labels column mapping table

label name	label number after transformation
Normal	1
DoS	2
Prob	3
R2L	4
U2R	5

The proposed intrusion detection system can be summarized as illustrated below in the following flow chart.

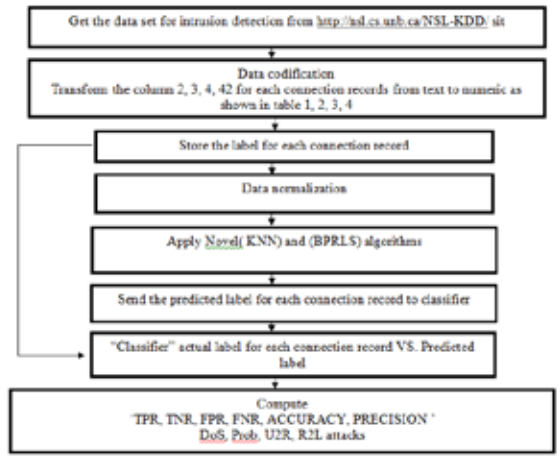

Figure 1: Proposed (BPRLS, Novel K-NN) algorithm for intrusion detection.

Novel K-Nearest Neighbor

The classifier of K-nearest neighbor (K-NN) type is considered as learning method that is based on instance; it is mainly depend n the similarity function or distance, including; the Euclidean and Cosine. During this research ;(K- NN) of a training data is calculated at the beginning. One sample similarities from testing data are tested for the classifiers exactness (Jivani, 2013). Na'mh (2012) summarizes an ordinary (K-NN) algorithm as following steps:

- The NSL-KDD99 training dataset are stored with its equivalent label.
- The distance is then calculated between each one of the connections within testing data set and those connection used for training.
- The calculated distances are then arranged in ascending order. The firs "Minimum Nearest Neighbor" is then selected considering k=1.
- The label for nearest neighbor is then selected; which is considered the test example prediction.
- The above procedure is repeated for all the connections within the considered testing dataset.

Jivani (2013) mentioned that if the number of training records for some classes is much more than the rest then there are chances that these records may get selected in the k nearest neighbors, and the test records would automatically get classified to the majority class instead of the actual class it belongs to. In this proposed algorithm the selection of k nearest records depends on the parameter n. This parameter is taken as input from the user and it depends on the size of the smallest class.

Novel k- nearest algorithm

1. First select the n nearest neighbors of connection record NSLKDD99 from each class founded in the training set, the value of n should not be greater than the size of the smallest class.
2. Sort these n nearest neighbors in descending order of the similarity-sim(r_i,x_(j)) .

$$Sim(r_i, r_j) = \frac{\vec{r}(r_i) \cdot \vec{r}(r_j)}{||\vec{r}(r_i)|| \cdot ||\vec{r}(r_j)||} \quad \ldots\ldots 1$$

3. Select now the top k nearest neighbors from the list prepared in step 2. These are the final k nearest neighbors of the records .

4. Using the decision rules given in (5.2) or (5.3), now find the class to which record r_i is most similar to .

$$Y(r_i) = max_k \sum x_j \in KNN\ y(x_j, c_k) \quad \ldots\ldots 2$$

$$Y(r_i) = max_k \sum x_j \in KNN\ sim(r_i, x_j)\ y(x_j, c_k) \quad \ldots\ldots 3$$

Volume : 4 | Issue : 3 | March 2015 • ISSN No 2277 - 8179

Where r_i is the record to be classified x_j is one of the neighbors of r_i and $y(x_j, c_k) \in \{0, 1\}$ indicates whether Record belongs to class c_k or not, $sim(r_i, x_j)$ is the similarity measure between the test record and its neighboring record . This is generally the cosine similarity In (1).

Back Propagation Recursive Least Square algorithm

According to the (MLP) structure with linear combiner that was presented by (Ham and Kostanic, 2000) ; linear combiner in the layer of the (MLP NN) , when the qth input pattern is presented to the network ,the output of combiner is calculated is calculated as an inner product between the combiner weights and the input vector to the particular layer x_(out.q)^((s-1)) that is,

$$v_{i(s)} = W_{i(s)}t \cdot X_{out.q^{(s-1)}} \quad \ldots\ldots (1)$$

Suppose the desired output for the particular combiner is is known for every pattern in the training set ,training the(MLP NN) effectively assumes training all its linear combiners, the goal of the learning algorithm is to minimize the squared error cost function, given by

$$j_i^{(s)} = \frac{1}{2} \sum_{q=1}^{m} \left(d_{i,q}^{(s)} - v_{i(s)}\right)^2 \quad \ldots\ldots (2)$$

Where m represents the total number of vectors in the training set; Eq. (2) can be written;

$$j_i^{(s)} = \frac{1}{2} \sum_{q=1}^{m} \left(d_{i,q}^{(s)} - W_{i(s)}t \cdot X_{out.q^{(s-1)}}\right)^2 \quad \ldots\ldots (3)$$

To find the weight vector which minimizes the cost function given in Eq. (3), we take the partial derivative with respect to and equate it to zero, that is

$$\frac{\partial j_i^{(s)}}{\partial w_{i(s)}} = \sum_{q=1}^{m} \left(-d_{i,q}^{(s)} x_{out.q^{(s-1)}} + x_{out.q^{(s-1)}} x_{out.q^{(s-1)}}t \, w_{i(s)}\right) = 0 \quad \ldots (4)$$

Defining

$$c_i^{(s)} = \sum_{q=1}^{m} x_{out.q^{(s-1)}} x_{out.q^{(s-1)}} t \quad \ldots\ldots (5)$$

And

$$p_i^{(s)} = d_{i,q}^{(s)} x_{out.q^{(s-1)}} \quad \ldots\ldots (6)$$

Equation (3) can be rearranged in vector matrix form as

$$c_i^{(s)} w_{i(s)} = p_i^{(s)} \quad \ldots\ldots (7)$$

Where can be inter predated as an estimate of the covariance matrix of the input to the layer , is the cross- correlation vector between input to the sth layer and the desired output of the ith linear combiner in the sth layer, is the weight vector to the liner combiner in the layer, Eq. (7) is referred to as deterministic normal equation in the context of adaptive filtering , if the covariance matrix and cross-correlation vector are known's the appropriate weight vector can be solved by using one of the standard technique for solving a system of linear equation

$$w_{i(s)} = [c^{(s)}]^{-1} p_i^{(s)} \quad \ldots\ldots (8)$$

The desired output of the particular nodes in the output layer is known, the desired summation for those nodes in the output layers can be calculated according to the inverse function in the following formula;

$$v\hat{}i^{(s)} = f^{(-1)}\left(d_{i,q}^{(s)}\right) \quad \ldots\ldots (9)$$

We have the desired summation for the nodes in the output layers as;

$$v\hat{}i^{(s)} = \frac{1}{a} \ln \frac{1 + d_{i,q}^{(s)}}{1 - d_{i,q}^{(s)}} \quad \ldots\ldots (10)$$

Return to equation (7) we do not have explicate knowledge of either the covariance matrices or cross correlation vector, and over the course of training of the net work they have to be estimated, the estimate of the correlation matrix for the layer can be written as

$$c_i^{(s)}(k+1) = b c_i^{(s)}(k) + x_{out.q^{(s-1)}} x_{out.q^{(s-1)}} t \quad \ldots\ldots (11)$$

The cross correlation vector for each the linear combiner can be estimated as

$$p_i^{(s)}(k+1) = p_i^{(s)}(k) + v\hat{}i^{(s)} x_{out.q^{(s-1)}} \quad \ldots\ldots (12)$$

Where; b coefficient is called the forgetting factor within the range. Eq. (11) is in recursive formula and it is required within recursive equation for the matrix of inverse autocorrelation [; this can be accomplished through employing the "Matrix Inversion Lemma" or through using Kalman filter.

$$k^{(s)}(k) = \frac{[c^{(s)}(k-1)]^{-1} x_{out^{(s-1)}}(k)}{b + x_{out^{(s-1)}}t(k)\,[c^{(s)}(k-1)]^{-1} x_{out^{(s-1)}}(k)} \quad \ldots\ldots (13)$$

And the update of the inverse matrix;

$$[c^{(s)}(k)]^{-1} = b^{(-1)} \left[[c^{(s)}(k-1)]^{-1} - k^{(s)}(k) x_{out^{(s-1)}}t(k) [c^{(s)}(k-1)]^{-1}\right] \quad (14)$$

Results and Discussion

Novel (K-NN) Algorithm Experiment Results

Table 5 represents the confusion matrix of Novel (K-NN) algorithm

Table 5: confusion matrix of novel (K-NN) algorithm

Confusion matrix	Novel K-NN result					
Target	Normal	DoS	Prob	R2L	U2R	Grand total
Normal	76	52	4	459	336	927
Dos	57	436	32	152	16	693
Prob	29	65	69	0	39	202
R2L	98	0	4	100	63	265
U2R	1	0	0	0	6	7
Grand total	261	553	109	711	460	2094

Table 6: Novel (K-NN) algorithm evaluation result for each class

Attack	TN	TP	FP	FN	Accuracy	Precision	TPR(recall) Detection rate (Sensitivity)	TNR (specificity)	FPR 1-TNR	FNR 1-DR
DoS	76	436	52	57	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
Prob	76	69	4	29	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
R2L	76	100	459	98	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
U2R	76	6	336	1	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]

We conclude from Table 5 and 6; novel (K-NN) produce more accurate result for DoS and prob class in term of "TPR,FPR" and more precision because of small false positive for these classes, novel (K-NN) algorithm produce more accurate result U2R class in term of "FNR" because of small false negative for these classes. Figure 5.11 represent novel (K-NN) algorithm detection rate for each class

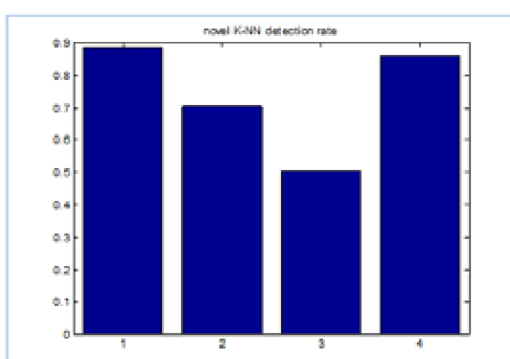

Figure 2: novel (K-NN) algorithm detection rate for each class

Table 7 represents actual class vs. predicted class by Novel (K-NN) algorithm.

Table 7: actual class vs. predicted class by Novel (K-NN) algorithm

	Predictive positive	Predictive negative
Actual positive(attack)	982(TP)	185(FN)
Actual negative(normal)	851(FP)	76(TN)

We can conclude from table 7 Novels (K-NN) algorithm was able to detect (982) attack records from (1167) records, with false negative is equal to (185). Table 8 represent Novel (K-NN) algorithm overall performance.

Table 8: Novel (K-NN) algorithm overall performance

Accuracy	0.505253
Precision	0.535734
TPR(recall) Detection rate (Sensitivity	0.841474
TNR (specificity)	0.0819849
FPR 1-TNR	0.918015
FNR 1-DR	0.505253

BPRLS Experimental Results

BPRLS was tested under several conditions considering tow experiments; at first experiment tow hidden layers have been used, the first hidden layer contain 20 neurons and the second hidden layer contain 10 neurons. On the other hand, the second experiment was performed considering the same number of hidden layers used in the first experiment, but different number of neurons were used; the first hidden layer contain 35 neurons and the second hidden layer contain 25 neurons. This section highlights and investigates the results for the second experiment. Table 9, 10 and 11 represent (BPRLS) algorithm topology and parameters, (BPRLS) algorithm confusion matrix and (BPRLS) algorithm evaluation result for each class for second respectively.

Table 9: (BPRLS) algorithm topology and parameters in second experiment

Parameters	Detail
Learning	Supervised
Layers	[41 35 25 1]
Performance	Mean square error
Mu	40
B	0.99
Transfer function	Sigmoid
Iteration count	30

Table 10: confusion matrix of (BPRLS) algorithm in second experiment

Confusion matrix	BPRLS result					
Target	Normal	DoS	Prob	R2L	U2R	Grand total
Normal	792	104	22	8	1	927
Dos	35	618	40	0	0	693
Prob	2	33	157	10	0	202
R2L	7	9	91	153	5	265
U2R	3	1	1	2	0	7
Grand total	839	765	311	173	6	2094

Table 11: (BPRLS) algorithm evaluation result for each class in second experiment

Attac k	TN	TP	FP	FN	Accuracy	Precisio n	TPR(recall) Detection rate (Sensitivity)	TNR (specificity)	FPR 1-TNR	FNR 1-DR
DoS	792	618	104	35	0.919907	0.855956	0.946401	0.903346	0.0966543	0.0535988
Prob	792	157	22	2	0.979183	0.877095	0.987421	0.97857	0.0222332	0.0125786
R2L	792	153	8	7	0.956842	0.95031	0.95625	0.991337	0.0083632	0.04375
U2R	792	0	1	1	0.991902	0	0	0.998972	0.0010277	1

It can be concluded from Table 10 and 11 that (BPRLS) produces more accurate for all classes in term of 'TPR, FPR, TNR, FPR' and more precision for all classes except the precision for U2R class. Figure below represent (BPRLS) algorithm detection rate for each class in second experiment.

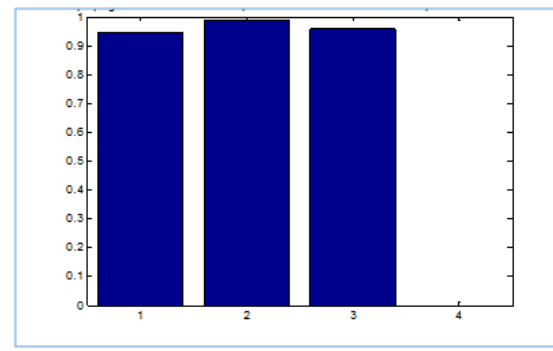

Figure 3: (BPRLS) algorithm detection rate for each class in second experiment

The following table represents the actual class vs. predicted class by (BPRLS) algorithm in second experiment.

Table 12: actual class vs. predicted class by (BPRLS) algorithm in second experiment

	Predictive positive	Predictive negative
Actual positive(attack)	1120 (TP)	47 (FN)
Actual negative(normal)	135(FP)	792 (TN)

It can be concluded from table 12 that (BPRLS) algorithm was able to detect (1120) attack records from (1167) records ,with false negative is equal to (47), (BPRLS) algorithm was able to detect (792) normal records from (927) records, with false positive (135). Table below represents the (BPRLS) algorithm overall performance in second experiment.

Table 13: (BPRLS) algorithm overall performance in second experiment.

Accuracy	0.919965
Precision	0.89243
TPR(recall) Detection rate (Sensitivity	0.959726
TNR (specificity)	0.878049
FPR 1-TNR	0.121951
FNR 1-DR	0.0402742

The Table below represents the Mean Square Error (MSE) of back propagation recursive least squares algorithms in second experiment.

Table 14: (MSE) in second experiment.

Number of iteration	MSE
30	0.2213

Evaluation

Comparison Between The Tow algorithms For Intrusion Detection

Volume : 4 | Issue : 3 | March 2015 • ISSN No 2277 - 8179

The tow algorithms; (Novel (K-NN) and (BPRLS)) were compared in terms of "TPR, TNR, FPR, FNR, accuracy, precision" for each class as illustrated below in table below.

Table 15: results of (Novel (K-NN) and (BPRLS)) algorithms.

Novel(K-NN) algorithm										
DoS	76	436	52	57	0.824477	0.893443	0.884381	0.59375	0.40625	0.115619
Prob	76	69	4	29	0.814607	0.945205	0.704082	0.95	0.05	0.295918
R2L	76	100	459	98	0.240109	0.178891	0.505051	0.142056	0.857944	0.494949
U2R	76	6	336	1	0.195704	0.0175439	0.857143	0.184466	0.815534	0.142857
Back propagation recursive least squares (BPRLS) algorithm										
DoS	792	618	104	55	0.919607	0.855956	0.946401	0.903346	0.0966543	0.0535988
Prob	792	157	22	2	0.979185	0.877095	0.987421	0.97867	0.0221328	0.0125786
R2L	792	159	8	7	0.986842	0.950311	0.95625	0.991837	0.00816327	0.04375
U2R	792	0	1	3	0.999902	0	0	0.999872	0.00102775	1

As shown in table 15; Back propagation recursive least squares algorithm produce more accurate result in term "(TNR,FNR)" than Novel K-nearest neighbor for all classes, expect the "(FNR)" for (U2R) class, However Novel K-nearest neighbor algorithm produce more accurate result in term "(FNR)" for (U2R) class than Back propagation recursive least squares algorithm because of small false negative generated by Novel K-nearest neighbor algorithm for that class, Back propagation recursive least squares algorithm produce more accurate result in term"(TPR)" for all classes than Novel K-nearest neighbors expect the "(TPR)" for (U2R) class, Back propagation recursive least squares algorithm produce more precision for all classes than novel K-nearest neighbors algorithm expect the precision for (U2R) class, Back propagation recursive least squares algorithm produce more accurate result in term"(FPR)" for (R2L,U2R) classes than Novel K-nearest neighbor, However Novel K-nearest neighbor algorithm produce more accurate result in term "(FPR)" for (DoS, prob) classes than BPRLS algorithm.

5-3 Comparison between Our System and Others Systems

The proposed system based on back propagation recursive least squares algorithm performance is compared with three modules in table 16. It is clear that our system produces more accurate result in terms of "FNR, FPR" for all class and detection rate about 0.95% for all classes.

Table 16: comparison the proposed system with other intrusion detection systems

Evaluation	IDs based on BP RLS			Namh,(2012)Hybrid-NN-ERBP		
	DR	FNR	FPR	DR	FNR	FPR
DoS	0.946401	0.0535988	0.0966543	97.9%	0.8%	0.9%
Prob	0.987421	0.0125786	0.0221328	0.99.8%	0%	1.64%
R2L	0.95625	0.04375	0.00816327	96.6%	0.9%	0.6%
U2R	0	1	0.00102775	0.59.5%	4.3%	0.6%
Evaluation	IDs based on BP RLS			Al-Rashdan,(2011)/ SOM+Conscience		
	DR	FNR	FPR	DR	FNR	FPR
DoS	0.946401	0.0535988	0.0966543	100%	0%	0%
Prob	0.987421	0.0125786	0.0221328	94.1%	5.9%	0%
R2L	0.95625	0.04375	0.00816327	88.2%	0	4.9
U2R	0	1	0.00102775	0.80%	20%	0%
Evaluation	IDs based on BP RLS			Alrubaye,(2014) ART		
	DR	FNR	FPR	DR	FNR	FPR
DoS	0.946401	0.0535988	0.0966543	99.70%	0.234%	0.43%
Prob	0.987421	0.0125786	0.0221328	99.03%	0.745%	0.16%
R2L	0.95625	0.04375	0.00816327	96.13%	2.149%	0.51%
U2R	0	1	0.00102775	93.62%	3.33%	3.33%

Conclusion

In this research paper, Two algorithms was tested to detect intrusion (Novel (K-NN), (BPRLS)), it can be concluded that each algorithm used to detect intrusion has it advantage and disadvantage depend on(TPR,TNR,FPR,FNR, accuracy, precision) for each class.

Also, it can be concluded that if we increase the number of hidden layers in back propagation recursive least squares(BPRLS) algorithm the auto correlation matrix Approaching singular matrix. The proposed system based on back propagation recursive least squares algorithm performance is compared with three modules in table 16,we see that our system produce more accurate result in term "FNR,FPR" for all class , and detection rate about 0.95% for all classes.

REFERENCE

[1] Chakraborty, N.2013. "Intrusion Detection System and Intrusion Prevention System: A Comparative Study," IJCBR, ISSN,PP 2229-6166. | [2] Poojitha, G., Kumar, K. N., & Reddy, P. J. 2010. "Intrusion Detection using Artificial Neural Network," In Computing Communication and Networking Technologies (ICCCNT), International Conference, PP. 1-7. | [3] Yichun, P., Yi, N., & Qiwei, H. 2012. "Research on Intrusion Detection System Based on IRBF," In Computational Intelligence and Security (CIS), Eighth International Conference, PP. 544-548. | [4] Norouzian ,M.R.,& Merati, S. 2011. "Classifying attacks in a network intrusion detection system based on artificial neural networks," Advanced Communication Technology (ICACT), 13th International Conference ,PP. 868-873. | [5] Riad, A. M., Elhenawy, I., Hassan, A., & Awadallah, N. 2013. "VISUALIZE NETWORK ANOMALY DETECTION BY USING K-MEANS CLUSTERING ALGORITHM," International Journal of Computer Networks & Communications, 5(5) PP.12-66. | [6] Pervez, S., Ahmad, I., Akram, A., & Swati, S. 2007. "A comparative analysis of artificial neural network technologies in intrusion detection systems," WSEAS Transactions on Computers, 6(1),PP 175-180. | [7] Wang, H., & Ma, R. 2009. "Optimization of neural networks for network intrusion detection," In Education Technology and Computer Science. ETCS'09. First International Workshop, 1, PP. 418-420. | [8] Bi, J., Zhang, K., & Cheng, X. 2009. "Intrusion detection based on RBF neural network," In Information Engineering and Electronic Commerce. IEEC'09. International Symposium PP. 357-360. | [9]Mukhopadhyay, I., Chakraborty, M., Chakrabarti, S., & Chatterjee, T. 2011. "Back propagation neural network approach to Intrusion Detection System," In Recent Trends in Information Systems (ReTIS), International Conference, PP. 303-308. | [10] Norouzian, M. R., & Merati, S. 2011. "Classifying attacks in a network intrusion detection system based on artificial neural networks," In Advanced Communication Technology (ICACT), 13th International Conference, PP. 868-873. | [11] AL-Rashdan, W, Naoum, R, Al_Sharafat, W & Al-Khazaaleh, M. 2010. "Novel network intrusion detection system using hybrid neural network (Hopfield and Kohonen SOM with conscience function)," IJCSNS International Journal of Computer Science and Network Security, 10(11). PP.11-55. | [12] Naoum, R. S., & Al Sultani, Z. N. 2013. "Hybrid system of learning vector quantization and enhanced resilient back propagation artificial neural network for intrusion classification," Int. J. Res. Rev. Appl. Sci, 14(2),PP.23-44. | [13] Naoum, R. S., Abid, N. A., & Al-Sultani, Z. N. 2012. "An Enhanced Resilient Back propagation Artificial Neural Network for Intrusion Detection System," IJCSNS, 12(3), PP 2-56. | [14] Kayacik, H. G., Zincir-Heywood, A. N., & Heywood, M. I. 2005." Selecting features for intrusion detection: A feature relevance analysis on KDD 99 intrusion detection datasets," In Proceedings of the third annual conference on privacy, security and trust, PP.12-44. | [15] Na'mh, Z. 2012. "An Enhanced resilient backpropagation artificial neural network for intrusion detection system," (MSc thesis). MEU: Amman, Jordan, PP 1-99. | [16] Jivani, A. G. 2013. "The Novel k Nearest Neighbor Algorithm," In Computer Communication and Informatics (ICCCI), International ConferencE, PP. 1-4. | [17]Ham, F. M., & Kostanic, I. 2000." Principles of neurocomputing for science and engineering," McGraw-Hill Higher Education, PP.1-219. | [18] Al-Rashdan, W. 2011. "A Hybrid artificial neural network model (Hopfield-SOM with Conscience) for effective network intrusion detection system," (PhD thesis) The Arab Academy for Banking and Financial Sciences, Amman: Jordan, PP 1-230.