Neural Network Techniques for Intrusion Detection

Vladimir Golovko[1)], Leanid Vaitsekhovich[2)]
Brest State Technical University, Moskovskaja str. 267, 224017 Brest, Belarus
1) gva@bstu.by
2) vspika@rambler.ru

Abstract: *This paper presents the neural network approaches for building of intrusion detection system (IDS). Existing intrusion detection approaches have same limitations, namely low detection time and recognition accuracy. In order to overcome these limitations we propose several neural network systems for intrusion detection.*

Keywords: neural networks, computer security, intrusion detection, principal component analysis, multilayer perseptron.

1. INTRODUCTION

At present time one of the form of world space globalization is cyber space globalization, because of increasing number of computers connected to the Internet. As a result the security of computer networks becomes more and more important.

There exist the different defense techniques, in order to protect the computer networks. Many Intrusion detection systems (IDS) are based on hand-crafted signatures or data mining techniques [1-3]. The other IDS use neural network approaches. The major problem of existing models is recognition of new attacks, low accuracy, and detection time and system adaptability [4].

This paper explores the different neural network techniques for construction of intrusion detection systems. We use limited data set for training of neural networks. This data set contains as normal and abnormal learning samples. The generalization capability of IDS is investigated. The KDD-99 dataset [5] is used for training and testing of proposed IDS. This dataset contains about 5 million network connection records with normal and abnormal states. Every record includes 41 independent features. All attacks can be divided into four main classes: DoS, U2R, R2L and Probe.

DoS – denial of service attack. This attack led to overloading or crashing of networks;

U2R – unauthorized access to local super user privileges;

R2L – unauthorized access from remote user;

Probe – scanning and probing for getting confidential data.

Every class consists of different attack types.

This paper considers the recognition as attack types and classes. The experimental results are discussed in Section 4.

2. IDS ARCHITECTURES

Let's examine the different neural network approaches for construction of intrusion detection systems. As for input data it will be used the 41 features from KDD-99 dataset, which contain the TCP-connection information. The main goal of IDS is detection and recognition type of attack. Therefore it will be used as for output data the m-dimensional vector, where m is number of attack plus normal connection. The significant question concerning design of IDS is the following: which features are really important? We propose to use principal component analysis (PCA) neural network for important data extraction and dimensionality reduction.

The second stage construction of IDS is to detect and to recognize attacks. In this paper is proposed to apply multilayer perceptron (MLP) for this purpose. Combining two different neural networks we can obtain the various IDS architectures.

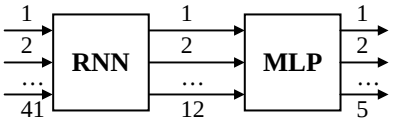

Fig. 1 – The first variant of IDS.

As shown in Fig. 1 the first variant of IDS architecture consists of PCA and MLP neural networks, which are connected consequently. The PCA network, which is also called a recirculation network (RNN), transforms 41-dimentional input vector into 12-dimensional output vector. The MLP performs the processing of compressed data for recognition one type of attack or normal state.

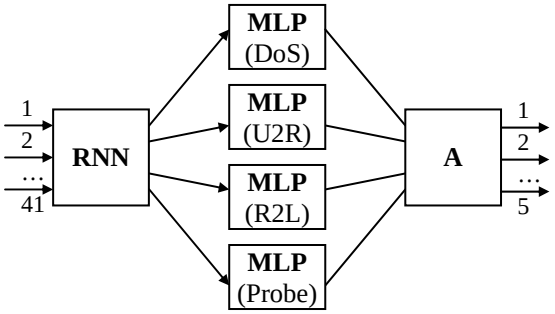

Fig. 2 – The second variant of IDS.

The second variant of IDS structure is shown in Fig. 2. It consists of four MLP networks. As can be seen every MLP network is intended for recognition one type of attack: DoS, U2R, R2L and Probe. The output data from 4 multilayer perceptrons enter to Arbiter, which accept the final decision concerning type of attack. The one layer perceptron can be used as Arbiter. The training of the Arbiter is performed after leaning of PCA and MLP neural networks. Such an approach permits to fulfill the hierarchical classification attacks. In this case Arbiter can define one of 5 attack types and corresponding MLP – class of attack.

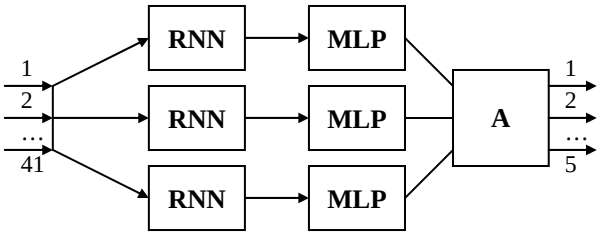

Fig. 3 – The third variant of IDS.

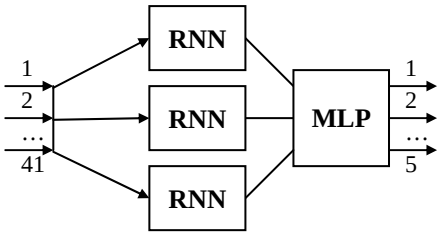

Fig. 4 – The forth variant of IDS.

The next variants of IDS structure are shown in the Fig. 3, 4. As can be seen from the Figures, the initial 41-dimensional vector here is divided on 3 parts, each of these contain the homogeneous data. Every PCA network is intended for processing of corresponding subvector. The MLP defines the type of attack and Arbiter accepts the final decision. Main difference between these two models is common MLP module in the variant 4.

3. NEURAL NETWORKS

As it is mentioned above we use PCA and MLP neural networks in order to construct IDS.

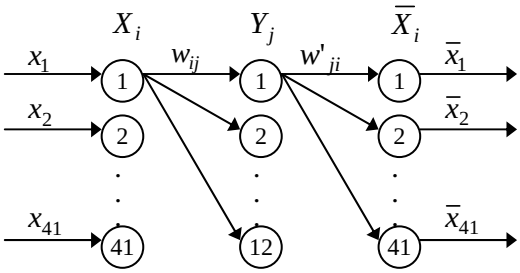

Fig. 5 – RNN architecture.

Let's consider an autoencoder, which is also called a recirculation network is shown in Fig. 5. It is represented by multilayer perceptron, which performs the linear compression of the data set through a bottleneck in the hidden layer. As can be seen the nodes are partitioned in three layers. The hidden units perform the compression of the input data set. The j-th hidden unit output is given by

$$y_j = \sum_{i=1}^{41} w_{ij} \cdot x_i , \qquad (1)$$

where w_{ij} is the weight from the i-th unit to the hidden j-th unit.

The output units are meant for decompression of the hidden data set. The i-th output unit is given by

$$\bar{x}_i = \sum_{j=1}^{12} w'_{ji} \cdot y_j . \qquad (2)$$

The weights of this network are updated iteratively in accordance with the Oja rule:

$$w'_{ji}(t+1) = w'_{ji}(t) + \alpha \cdot y_j \cdot (x_i - \bar{x}_i) , \qquad (3)$$

$$w_{ij} = w'_{ji} . \qquad (4)$$

As it is known [6] such a RNN performs a linear dimensionality reduction. In this procedure the input space is rotated in such a way that the output values are so uncorrelated as possible and the energy or variances of the data is mainly concentrated in a few first principal components.

The preprocessing of input data is performed before entering it to RNN:

$$x_i^k = \frac{x_i^k - \mu(x_i)}{\sigma(x_i^k)} , \qquad (5)$$

where

$$\mu(x_i) = \frac{1}{L} \sum_{k=1}^{L} x_i^k , \qquad (6)$$

$$\sigma(x_i^k) = \frac{1}{L} \sum_{k=1}^{L} (x_i^k - \mu(x_i))^2 . \qquad (7)$$

Here L is the number of training samples. The KDD-99 data set are used for RNN training. The mean square error makes 0.01. The training set contains 20% samples.

Let's consider the mapping of input space data for normal state and Neptune type of attack on the plane two principal components. As can be seen from the Fig. 6 the data, which belong one type of attack can be located in different areas. As a result is not possible the classification of such a data using only linear RNN because of complex relationships between features. One way to decide this problem is to use the nonlinear PCA network.

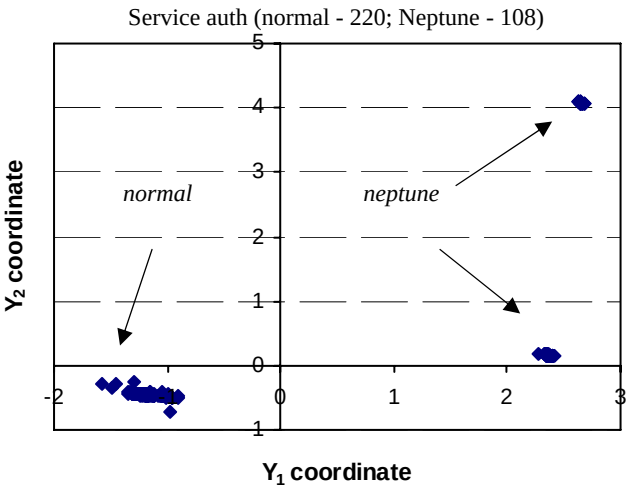

Fig. 6 – Data processed with RNN (service auth).

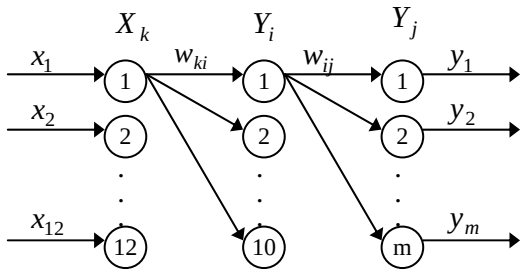

Fig. 7 – MLP architecture.

As it is mentioned before the MLP is intended for attack classification on the basis of principal components (Fig. 7). The number of output units depends on determination of type or class attack. The backpropagation algorithm is used for training MLP. The mean square error makes 0.01. After training of neural networks they are combined in an intrusion detection system.

4. EXPERIMENTAL RESULTS

In this paper the KDD-99 data set is used for training and testing different neural network models. The experiments were performed separately for each service. The learning models were trained with 20% selections from data sets for each service. After training a neural network is ready to be used. Some evaluation metrics were calculated during the testing process such as detection and recognition rates, true attack alarms, false attack alarms, etc.

Let's examine the recognition of attacks with the Model 1 (see Section 2). Table 1 shows statistics of recognition attacks for some services depending on attack class. Total data for almost 30 services are given in Table 2.

Table 2. Identification and recognition statistics depending on attack class for the Model 1 (almost 30 services)

class	count	detected	recognized
DoS	286369	286334(99,9%)	286087(99,9%)
U2R	49	41(83,7%)	40(97,6%)
R2L	1119	1000(89,4%)	906(90,6%)
Probe	1320	1312(99,4%)	1308(99,7%)
normal state			
normal	83281	---	82943(99,6%)

From the above results, the best detection and recognition rates were achieved for DoS and Probe connections. U2R and R2L attack instances were detected slightly worse (83.7% and 89.4% respectively). Besides, the bottom row shows that some normal instances were (incorrectly) classified as intrusions.

Next results (Table 3, 4) are associated with testing in the mode of attack type recognition. Experiments were performed with different count of output neurons. The first case is 23 output units that represent every type of attack and a normal state. The second case is when the number of output units varies dynamically. It means that the program automatically calculates a number of different states of network connections depending on their count in the training set.

Table 3. Identification and recognition statistics depending on attack type for the Model 1 (dynamic count of output units)

service	true attack alarms	false attack alarms	recognized correctly
auth	108(100%)	0	108(100%)
domain	113(100%)	0	113(100%)
eco_I	1252(99,9%)	7(1,8%)	1238(98,9%)
ecr_I	281033(99,9%)	12(3,4%)	281033(100%)
finger	202(100%)	11(2,3%)	200(99,0%)
ftp	283(66,5%)	14(3,8%)	283(100%)
ftp_data	864(93,7%)	44(1,2%)	777(89,9%)
http	2399(99,7%)	96(0,16%)	2396(99,9%)
IRC	1(100%)	1(2,38%)	1(100%)
pop_3	123(100%)	0	123(100%)
smtp	123(97,7%)	44(0,4%)	121(98,4%)
telnet	284(96,6%)	16(7,31%)	280(98,6%)

Table 4. Identification and recognition statistics depending on attack type for the Model 1 (23 output units)

service	true attack alarms	false attack alarms	recognized correctly
auth	108(100%)	0	108(100%)
domain	113(100%)	0	113(100%)
eco_I	1252(99,9%)	0	1239(99,0%)
ecr_I	281034(99,9%)	13(3,77%)	281034(100%)
finger	186(92,1%)	10(2,14%)	185(99,5%)
ftp	418(98,3%)	26(6,97%)	418(100%)
ftp_data	856(92,7%)	31(0,82%)	636(74,3%)
http	2400(99,7%)	96(0,16%)	2400(100%)
IRC	1(100%)	1(2,38%)	1(100%)
pop_3	123(100%)	0	123(100%)
smtp	122(97,6%)	35(0,36%)	119(97,5%)
telnet	284(96,6%)	15(6,85%)	272(95,7%)

The results of testing (see Table 3, Table 4) are very comparative between the two modes.

It is interesting to discuss other models proposed in Section 2. In the case with the Model 4 parameters of input vector were partitioned into the tree groups of whole numbers, keys (0/1) and numbers taking values from the range [0..1]. Each group is processed with the corresponding RNN. The purpose of this is to increase algorithm accuracy owing to the fact that each RNN works with homogeneous data. Though the Model 4 resulted in slightly lower detection rates for U2R and R2L in comparison with the Model 1, it is still quite sensitive to the widely distributed in the KDD-99 DoS attacks. The main advantage of the Model 4 is that it allows to reduce training time. Due to smaller number of links in the module, that calculates principal components, it needs less computational requirement during the training process.

Table 5. Identification and recognition statistics depending on attack class for the Model 4 (almost 30 services)

class	count	detected	recognized
DoS	286369	286369(100%)	286295(99,9%)
U2R	49	33(67,3%)	32(97,0%)
R2L	1119	442(39,5%)	427(96,6%)
Probe	1320	1311(99,3%)	1288(98,2%)
normal state			
normal	83281	---	77673(93,2%)

We found that there was often situation when detection rates for some attack classes were considerably lower than for others. It was necessary to repeat training process from the very beginning to achieve desired results. We have applied the Model 2 to solve the problem. The goal in using this neural network architecture is to be able to get more accurate result for definite attack class. It is also possible to retrain each module MLP taken separately after general training circle has taken place. Table 6 summarizes the performance of this kind of neural network.

Table 6. Identification and recognition statistics depending on attack class for the Model 2 (almost 30 services)

class	count	detected	recognized
DoS	286369	286032(99,9%)	286022(100%)
U2R	49	41(83,7%)	37(90,2%)
R2L	1119	1063 (95,0%)	1049(98,7%)
Probe	1320	1306(98,9%)	1306(100%)
normal state			
normal	83281	---	83009(99,7%)

Table 1. Detailed identification and recognition statistics depending on attack class for the Model 1

service	normal		DoS			U2R		
	count	recognized	count	detected	recognized	count	detected	recognized
auth	220	220(100%)	108	108(100%)	108(100%)			
domain	3	3(100%)	112	112(100%)	112(100%)			
eco_I	389	387(99,5%)						
ecr_I	345	327(94,8%)	281049	281031 (100%)	281031 (100%)			
finger	468	456(97,4%)	197	189(95,9%)	85(45,0%)			
ftp	373	359(96,2%)	104	104(100%)	104(100%)	3	3(100%)	3(100%)
ftp_ data	3798	3752(98.8%)	170	168(98,8%)	26(15,5%)	12	12(100%)	11 (91,7%)
http	61885	61787(99,8%)						
IRC	42	41(97,6%)						
pop_3	79	79(100%)	118	118(100%)	118(100%)	34	26(76,5%)	26(100%)
smtp	9598	9472(98,7%)	120	120(100%)	120(100%)			
telnet	219	204(93,2%)	198	198(100%)	198(100%)	34	26(76,5%)	26(100%)

Table 1. Detailed identification and recognition statistics depending on attack class for the Model 1 (continuation)

service	R2L			Probe		
	count	detected	recognized	count	detected	recognized
auth						
domain				1	1(100%)	1(100%)
eco_I				1253	1251(99,8%)	1251(100%)
ecr_I				6	0(0,0%)	0(0,0%)
finger				5	5(100%)	4(80,0%)
ftp	313	245(78,3%)	244(99,6%)	5	5(100%)	5(100%)
ftp_ data	733	683(93,2%)	595(87,1%)	8	8(100%)	7(87,5%)
http	4	4(100%)	4(100%)	8	8(100%)	8(100%)
IRC				1	1(100%)	1(100%)
pop_3				5	5(100%)	5(100%)
smtp				5	5(100%)	3(60,0%)
telnet	57	56(98,2%)	53(94,6%)	5	5(100%)	5(100%)

5. CONCLUSION

In this paper the neural network architectures for intrusion detection have been addressed. The propose approach is based on integration of the recirculation network and multilayer perceptron. The KDD-99 dataset was used for experiments performing. Combining two different neuron networks (RNN and MLP) it is possible to produce efficient performance in terms of detection and recognition attacks on computer networks. The main advantages of using neural network techniques are ability to recognize novel attack instances and quickness of work, what is especially important in real time mode.

6. REFERENCES

[1] E. Eskin. Anomaly detection over noisy data using learned probability distributions. In Proceedings of the Seventeenth International Conference on Machine Learning (ICML-2000), 2000

[2] A. Ghosh and A. Schwartzbard. A study in using neural networks for anomaly and misuse detection. In Proceedings of the Eighth USENIX Security Symposium, 1999

[3] W. Lee, S. J. Stolfo and K. Mok. Data mining in work flow environments: Experiences in intrusion detection. In Proceedings of the 1999 Conference on Knowledge Discovery and Data Mining (KDD-99), 1999

[4] E. Eskin, A. Arnold, M. Prerau, L. Portnoy, S. Stolfo. A Geometric Framework for Unsupervised Anomaly Detection: Detecting Intrusions in Unlabeled Data, In D. Barbara and S. Jajodia (editors), Applications of Data Mining in Computer Security, Kluwer, 2002

[5] 1999 KDD Cup Competition. http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html

[6] E. Oja. Principal components, minor components and linear networks. Neural Networks, vol.5, pp.927-935, 1992