Journal of Computer Sciences and Applications, 2013, Vol. 1, No. 3, 33-38
Available online at http://pubs.sciepub.com/jcsa/1/3/1
© Science and Education Publishing
DOI:10.12691/jcsa-1-3-1

Improving the Intrusion Detection Systems' Performance by Correlation as a Sample Selection Method

Rahimeh Rouhi[1,*], Farshid Keynia[2], Mehran Amiri[3]

[1]Department of Computer Engineering, Islamic Azad University, science and research branch, Kerman, Iran
[2]Graduate University of Advanced Technology, Kerman, Iran
[3]Department of Computer Engineering, Islamic Azad University, science and research branch, Kerman, Iran
*Corresponding author: r.rouhi.srb@iauk.ac.ir

Received December 20, 2012; Revised April 27, 2013; Accepted May 02, 2013

Abstract Due to a growing number of the computer networks in recent years, there has been an increasing interest in the intrusion detection systems (IDSs). In this paper we have proposed a method applied to the instance selection from KDD CUP 99 dataset which is used for evaluating the anomaly detection techniques. In order to determine the performance of proposed method in the dataset reduction, a feed forward neural network was trained by a reduced dataset to classify normal or attack records in the dataset. The most obvious finding resulted from this study is a considerable increase in the accuracy rate obtained from the neural network.

Keywords: *intrusion detection system (IDS), instance selection, anomaly detection, neural network*

1. Introduction

Intrusion is referred to a sequence of activities interfering in a computer network and threatening its security [1]. Due to a growth of computer networks, creation of a system to provide protection for these networks is of great importance, therefore intrusion detection systems have appeared to establish protection for computer networks. It is a classification process which has attracted the attention of researchers. Classification task can detect normality or abnormality behavior in a set of records. Two points should be taken into account with regard to the classification: distinguishing between normal and abnormal behavior is referred to a two-class problem, in contrast if the number of classes are more than two, it is referred to a multi-class problem [2].

The aim of IDSs is to form a classifier to correctly recognize types of normality or abnormality records. In general, intrusion detection techniques are divided into two categories; anomaly detection: it initially models the normal behaviors then recognizes behaviors which have exceeded certain standard measure, misuse detection: it models the attack behaviors in a system, using patterns of well-known attacks or vulnerable spots to distinguish them [3].

There exist some techniques reported for feature selection in datasets used in intrusion detection domain. In [4] an automatic feature selection procedure is reported based on a correlation measure. Also in [5] a correlation-based feature selection algorithm presented to keep useful features that resulted in increasing the accuracy of classifier. Although there are some attempts using correlation as a feature selection measure, correlation has not been used as an instance selection method. In this paper we propose a novel instance selection method based on correlation. The results show good performance and look promising.

2. Related Works

Neural networks are a major part of machine learning algorithms. Basically a neural network is composed of highly connected processing elements which are called *neuron*. A specific kind of function is used to limit the outputs of neurons to a pre-specified interval. The neurons are connected to each other by different topologies. There are some research papers which have used neural networks as intrusion detection systems. This is due to the promising results which are generated by them and also their generalization ability which helps them find *day-0* attack [1]. This marvelous generalization ability also helps NN find new attacks. Some of papers on IDSs which are based on neural networks are outlined in the followings. Debar et al. [6] introduce an IDS which is based on combining an ordinary expert system and a neural network trying to improve the accuracy. Generated results present a good performance of this combination. In [7], Lin et al. describe an IDS called NNID which is an abbreviation for Neural Network Intrusion Detector. NNID uses neural network with back propagation algorithm. This model is used to monitor user’s behavior. Ryan et al. [8] introduce an IDS based on MLP neural network and back propagation algorithm. This model is also used for tracking users` profile and behavior. However this model is offline. Ghosh et al. [9] show that neural networks can be used in both anomaly and misuse detectors. They predict types of TCP connections based on previous trait of users. Cannady [10] uses a three layer neural network for predicting the type of TCP connections using 10,000 connection records including 1000 simulated attacks. Mukkamala [11] uses three and four layer networks for intrusion detection. The results show about 99 % correct classification. In [1] Beghdad applies five different neural networks to detect intrusions in network. They are MLP, GFF, RBF, SOM and PCA. Based on this research, GFF leads to a better confusion matrix and RBF generates better results. Tan [12] uses some information, such as command sets, CPU usage, login host addresses, to distinguish between normal and abnormal behavior, while Ryan et al. [13] considered the patterns of commands and their frequency. Research, such as [14,15], employed RBFs to learn multiple local clusters for well-known attacks and for normal events. Other than being a classifier, the RBF network was also used to fuse results from multiple classifiers [16]. Jiang et al. [17] reported a novel approach which integrates both misuse and anomaly detections in a hierarchical RBF network.

The instance selection is a common issue in studies in the machine learning and pattern recognition problems from the statistical viewpoints. Its application in different phases such as classification is significantly important because in such an application there might be a lot of similar or repeated instances (records or connections) which can cause disturbance in the neural network training leading to the retention of the redundant data by the classifier. Such a deletion of the redundant data not only does not cause any informational drawbacks but also it decreases the computational costs like, speed and spatial storage memory.

There are various methods for running the instance selection. Each method tries to find the best subset based on the criterion from 2^n candidate subsets of the instances for a dataset with *n* members. In all methods attempt has been made to select a subset as the output records based on the type of the problem. Finding an optimal set of records is difficult and costly in the medium and the large *N*s. Instance selection methods can be divided into the following methods: Filter and Wrapper methods. Langley [18] states that in the wrapper methods, the records which do not play any role in the classification accuracy are removed from the main dataset, while filter methods are independent from the inference algorithms and the selection criterion is not based on the classifier.

On the other hand, there are other classification views for the instance selection methods such as incremental and decremental methods [19]. Incremental method starts with $S=\emptyset$ and then the records which are supposed to be in the reduced dataset are added to set *S*. While, decremental method starts with the main dataset *T* and gradually the records are examined. In this method if a record does not satisfy the criterion it is removed from *T*.

Most proposed wrapper methods are based on K-NN classifier [20]. One of the earliest wrapper methods is Condensed Nearest Neighbor (CNN) method [21] .Since this method is an incremental one, it initially inserts the instances belonging to each class to S randomly. Then it classifies each instance in *T* based on. If an instance *p* is misclassified, it inserts that instance to S and ensures that all instances similar to *p* are classified correctly. As a result, noisy instances can be kept because they are commonly misclassified by their neighboring instances.

Another type of CNN is Generalized Condensed Nearest Neighbor (GCNN) method [22], it is similar to CNN method differing in the fact that the related method inserts the instances to S satisfying an absorption criterion based on a threshold. For each instance, the absorption is calculated according to the nearest neighbors and enemies. Another method is Edited Nearest Neighbor (ENN) [23]. Such a method discards the noise instances from the training set *T*. For the instance, if the class of an instance is different from the majority class of its k nearest neighbors, the instance p is discarded (in ENN, k = 3). There is another variant of ENN, Repeated ENN (RENN). The method applies ENN repeatedly until all instances in S have the same class with their k nearest neighbors. All k-NN is another type of ENN, all instances which are misclassified are labeled by their k nearest neighbors and then all of the labeled instances are discarded [24].

Each instance in the training set can be either a border instance or an interior one. Instance p_j is defined as a border instance for the class c_i if $p_j \in c_i$ and the p_j is the nearest neighbor to any instance in the class $c_i \sim= c_k$ Contrarily, an instance which is not a border type is called an interior one. Border instances have useful information about the class discrimination region [25]. Patterns by Ordered Projections (POP) method is a filter method, in this method interior instances are discarded from the training set and some border instances are selected [26]. The related method is performed based on a concept called *weakness*(*p*) being defined as the number of times that *p* is not a border instance in a class with respect to its attribute values. Some filter methods such as Object Selection by Clustering (OSC) select both the border and interior instances [27].

Clustering approach for the instance selection has been stated by some researchers [28,29]. Clustering is done by turning *T* into *n* clusters, and then the selected instances are set as the clusters' centers. In GCM (Generalized-Modified Chang Algorithm) method same-class nearest clusters are merged and it selects the centers related to the newly merged clusters [30]. According to Venmann and Reinders, in Nearest Sub-class Classifier (NSB) method the selection of the different numbers of the instances (clusters) in each class is done by Maximum Variance Cluster Algorithm [31].

Filter methods compared to the wrapper ones are more efficient [19]. They have obtained a good accuracy and retention. The main characteristic of such methods is that the runtime is shorter than that of the wrapper methods [32,33]. Moreover, because their selection criterion is not based on the classifier, the resulted subset will obtain an acceptable accuracy when the different classifiers are used [26,27].

In this study by computing the correlation coefficient between each two the records in KDD CUP 99 dataset, we proposed a new method for the instance selection to delete the redundant records. By applying the proposed method in KDD CUP 99 dataset, the elapsed time for creating reduced dataset dramatically decreased. Furthermore, after the neural network training performed by the reduced dataset, a considerable increase was obtained in the accuracy of the record classification based on the neural network. To provide a better understanding of the proposed algorithm the rest of the present article is organized in the following order: In section 2, KDD CUP 99 dataset is described, in section 3 the function of the proposed algorithm is explained, following the explanation of the neural network in section 4, numerical results are presented in section 5, and finally the conclusion is drawn.

3. KDD CUP 99 Dataset

KDD CUP 99 dataset is taken from DARPA (Defense Advanced Research Projects Agency) which has been used widely to asses anomaly detection methods since 1999. This dataset has been criticized by McHugh [34], mainly because of the characteristics of the synthetic data. As a result, some of the existing problems in DARPA remain in KDD. KDD CUP 99 training set and test set contain respectively 4898431 and 311027 intrusion and normal records. Intrusion types in this dataset are divided into four groups:

Denial of Service Attack (DoS): is an attack in which the attacker makes some computing or memory resource too busy or too full to handle legitimate requests, or denies legitimate users access to a machine.

User to Root Attack (U2R): is a class of exploit in which the attacker starts out with access to a normal user account on the system (perhaps gained by sniffing passwords, a dictionary attack, or social engineering) and is able to exploit some vulnerability to gain root access to the system.

Remote to Local Attack (R2L): occurs when an attacker who has the ability to send packets to a machine over a network but who does not have an account on that machine exploits some vulnerability to gain local access as a user of that machine.

Probing Attack: is an attempt to gather information about a network of computers for the apparent purpose of circumventing its security controls.

The number of redundant records in the KDD training set and test set in Tavallaee et al. [35] are shown in Table 1 and Table 2.

Table 1. Statistics of redundant records in the KDD train set

	Original Records	Distinct Records
attacks	3,925,650	262,178
Normal	972,781	812,814
Total	4,898,431	1,074,992

Table 2. Statistics of redundant records in the KDD test set

	Original Records	Distinct Records
attacks	250,436	29,378
Normal	60,591	47,911
Total	311,027	77,289

Each record in the dataset has 42 features determining records' being normal or intrusion, 22 of this features explain the connection and 19 of them describe their connections' properties to the same host within in the last two seconds [1]. It should be pointed out that the last feature in each record determines the normality or abnormality of the related record. The features of each record can be classified into three groups:

Basis features: this category encapsulates all the attributes that can be extracted from a TCP/IP connection. Most of these features leading to an implicit delay in detection.

Traffic features: this category includes features that are computed with respect to a window interval and is divided into two groups:

"same host" features: examine only the connections in the past 2 seconds that have the same destination host as the current connection, and calculate statistics related to protocol behavior, service, etc.

"Same service" features: examine only the connections in the past 2 seconds that have the same service as the current connections.

Content features: unlike most of the DOS and probing attacks, the R2L and U2R attacks don't have any intrusion frequent sequential patterns.

4. Proposed Method

Correlation coefficient is a statistical scale between 2 variables x, y. Considering the degree of the statistical correlation between the 2 variables, it can obtain values in the range of [-1,1]. A correlation with a value of 0 indicates that there is no relationship between the two variables, the values of -1 and +1 show respectively a perfect negative and a perfect positive correlation.

To compute the correlation value three types of sum squares, the sum of square value x, the sum of square value y, and the sum of cross-product x, y, are needed, see Equations (1), (2), (3) and (4) as the following.

$$SS_{XX} = \sum (x_i - \overline{x})^2 \tag{1}$$

$$SS_{YY} = \sum (y_i - \overline{y})^2 \tag{2}$$

$$SS_{XY} = \sum (x_i - \overline{x})(y_i - \overline{y}) \tag{3}$$

$$Corr(X,Y) = \frac{SS_{xy}}{\sqrt{(SS_{xx})(SS_{yy})}} \tag{4}$$

As each record in the dataset has 42 different features, therefore, each record is shown with a 42-feature vector of different values. Four features in each record i.e. features 2, 3, 4 and 42, have different nominal values and other features of the record obtain numerical values. To calculate the correlation between each two records, nominal values of the features 2, 3, 4 and 42 in each record should be equal to the nominal values of the related features in the second one. Once the equality is ensured, the correlation among the numerical values related to the numerical features of the two records (not the ones related to features 2, 3, 4 and 42) is calculated. In order to better compare the records, nominal value features are replaced by numerically individual values.

If the correlation value of the two records exceeds a predetermined threshold, it indicates that these two records are statistically similar to each other to a great extent. Hence, to prevent any disturbance in the neural network training, one of the two records must be discarded. Contrarily, if the related value is less than or equal to the threshold, none of the two compared records are discarded.

To calculate the correlation value between each two records in the whole dataset, we initially divided the dataset into sets of 2000 records. Then the correlation value of both records in each partition was calculated. The same process was conducted on the remaining records so that a reduced dataset was formed containing records which were not repeated or did not have a strong similarity. For example, for the two records A and B taken from the dataset, the calculated correlation value is 0.9368 indicating that both records should be retained because their similarity degree is less than 0.99.

A=[0,1,1,1,233,504,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,7,7,0, 0,0,0,1,0,0,64,199,1,0,0.02,0.03,0,0,0,0,1]

B=[0,1,1,1,297,2000,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,6,6, 0,0,0,0,1,0,0,221,255,1,0,0,0.1,0,0,0,0,1]

Since there is not an agreement upon criterion for determining correlation coefficient strong, we examined five different thresholds and at the end, threshold 0.99 produced the best results.

5. Neural Network

To assess the proposed method in the classifier training function, we applied a feed forward 3-layer neural network which was trained by the reduced dataset to classify intrusion and normal records. The value related to the first 41 features of each record, except the last one, was given to the neural network as an input vector. As a result, 41 neurons were located on the input layer of the related neural network. To perform better, the number of neurons in the hidden layers of the neural network was respectively 2, 6. Using a 2-class categorization to distinguish the intrusion records from the normal ones, we put one neuron in the output layer of the neural network so it would produce two outputs 0, 1 to detect respectively the normal and intrusion records.

Transformation functions related to the three layers of the neural network were selected logsig, logsig and tansig respectively. We applied Error Back Propagation Algorithm (EBP) to training of the neural network. The test error of the neural network was calculated by MSE (Mean Square Error) and MAPE (Mean Absolute Percentage Error), based on Equations (5) and (6).

$$MAPE = 100 \times \frac{1}{n}\sum_{i=1}^{n}\frac{\left|O_{(i)} - F_{(i)}\right|}{O_{(i)}} \tag{5}$$

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(O_{(i)} - F_{(i)})^{2} \tag{6}$$

where O *and* F are respectively the target matrix and output matrix, i is the i^{th} element in O or F and n is the number of the records in dataset.

There are two approaches in applying the KDD CUP 99 dataset in the training and testing of the classifiers [21]. In the first approach, the KDD CUP 99 training set is applied to the record selection of not only the test set but also the classifiers' training set.

In the second approach, the KDDCUP 99 training set and its test set are used to respectively select the records of the training set and classifier testing separately.

In this article we applied the second approach in a way that the set which was used for the neural network training was the same dataset which had been produced by our proposed algorithm. The related algorithm selected the records from the KDD training set having 2,742 normal and intrusion records altogether. The redundant records were removed from the KDD test set so that the dataset reduced to 77,289 normal and intrusion records. Therefore, the related reduced dataset was used for the neural network testing.

6. Numerical Results

Applying the proposed algorithm to select the records from the KDD CUP 99 training set and also setting the threshold of 99% as a band for a strong correlation between the two compared records led into a decrease in the elapsed time of approximately 1.44 hour. The proposed algorithm was run in MATLAB environment on a PC with the following characteristics: Intel Pentium 4 (2.93GHz) and 4 GB of RAM with windows operating system. The final number of the remaining records was 2,768. The runtime of the reduction algorithm and the number of the remaining records with different thresholds are displayed in Table 3. Also the calculated test error of the neural network by MAPE and MSE is presented in Table 4.

Table 3. Runtime of the algorithm and remaining records

Threshold	Runtime of reduction algorithm(hour)	Number of remaining records	Reduction Rate(%)
0.99	1.44	2768	99.943
0.95	1.36	1214	99.975
0.90	1.33	935	99.980
0.85	1.29	835	99.982
0.80	1.27	771	99.984

Table 4. Extracted test errors based on the neural network

Threshold	MAPE(%)	MSE(%)
0.99	4.12	0.047
0.95	5.62	0.054
0.90	6.25	0.056
0.85	6.74	0.059
0.80	7.02	0.068

With examining different thresholds for the reduction algorithm and subsequently training of NN at the end the MAPE minimum value of 4.12% in test step was generated indicating that the neural network succeeded in accurately recognizing the test set records' normality or abnormality of about 95.88% of the records. It is the best obtained accuracy rate in classifying the records of reduced dataset with respect to the threshold value of 0.99. The produced accuracy rate of the neural network for the different values of threshold has been shown in Figure 1.

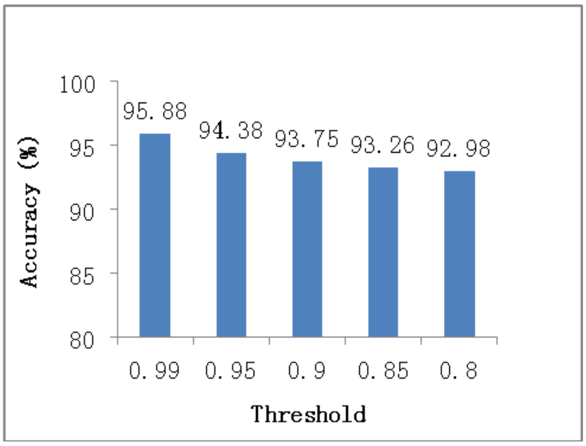


Figure 1. Performance of reduced dataset in recognizing records by neural network

To evaluate the proposed method we used the metric of accuracy as the following Equation (7).

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \qquad (7)$$

TP: true positive, the classification result is positive in presence of abnormality.

TN: true negative, the classification result is negative in absence of abnormality

FP: false positive, the classification result is positive in absence of abnormality.

FN: false negative, the classification result is negative in presence of abnormality.

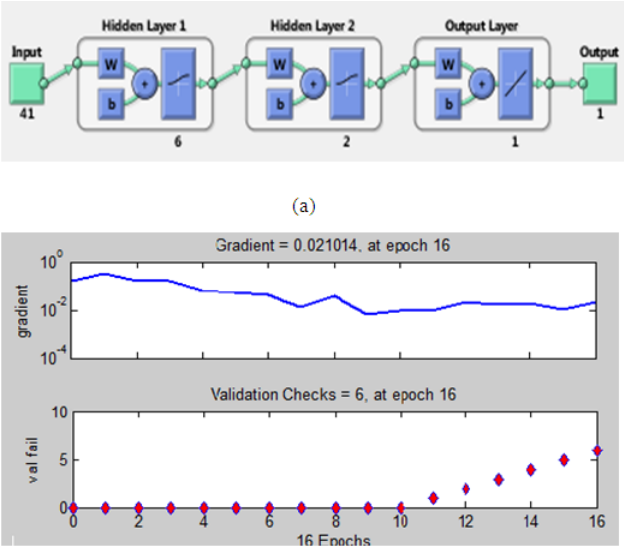
It should be pointed out that although setting a threshold higher than 0.99 to determine the presence or absence of the records in the dataset increases the comparing time by the reduction algorithm, it causes some records to remain in the dataset leading to an improvement in the neural network training and subsequently a decrease in the network test error. Figure 2 and Figure 3 show the training results using the reduced dataset with threshold 0.99 as a training data set to train the neural network.

7. Conclusion

In the supervised learning method, the redundant and similar records in the training set which are applied to the neural network training, cause data retention by the classifier. It can incline the function of the classifier recognition in the network test phase toward selecting the records with more repeated numbers or towards those which are more similar to the other records. Moreover, the existence of a wide number of the records in the dataset declines the speed rate of the neural network training. To overcome such drawbacks, in the present article the statistical correlation coefficient between the 2 records in KDD CUP 99 dataset was computed and an instance selection method to remove redundant records of the aforementioned dataset was proposed. Findings of the present article are as follows:

Due to a parallel performance of the proposed algorithm the elapsed time for formation of the reduced dataset dramatically decreased to about 1.44 hour with respect to threshold value of 0.99. Also because of the presence of a small number of the efficient and applicable records in the reduced dataset, classifier training time decreased.

At the end, following the network training by the reduced dataset a considerable increase in the accuracy rate of the neural network was produced.



(a)

(b)

Figure 2. (a) Neural network structure. (b) Training state gradient

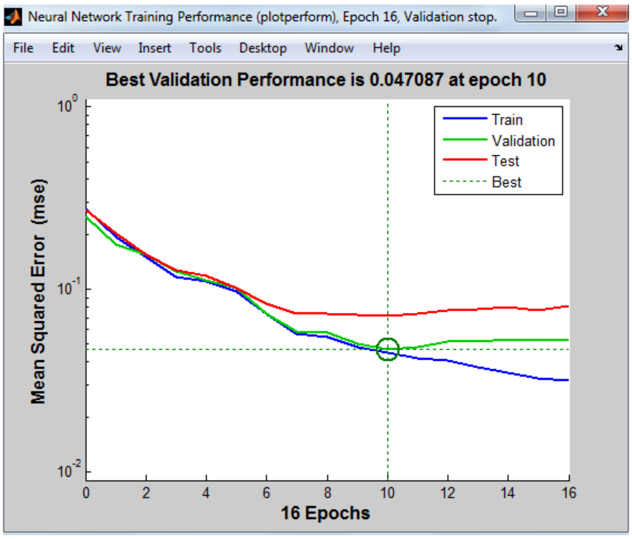


Figure 3. Performance measure for neural network trained to detect normal or abnormal records

Acknowledgement

Thanks to Mahdieh Rouhi for improving the language. Thanks also to the support of computer engineering group.

References

[1] Beghdad, R.,"Critical study of neural networks in detecting intrusions," Computers & Security 27. 168-175. 2008.

[2] Bolón-Canedo, V., Sánchez-Maroño, N., Alonso-Betanzos, A.,"Feature selection and classification in multiple class datasets: An application to KDD CUP 99 dataset," Expert System with Application, 38. 5947-5957. 2011.

[3] Chimphlee, W., HananAbdullah, A., Md Sap, M.N., Chimphlee, S., Srinoy, S., "A Rough-Fuzzy Hybrid Algorithm for computer

intrusion detection," The International Arab Journal of Information Technology, 4(3): 247-253. 2007.

[4] Nguyen, H., Franke, K., Petrović, S., "Improving effectives of intrusion detection by correlation feature selection," International Conference on Availability, Reliability and Security 10. 17-24.2010.

[5] Chon, T.S., Kang, K.Y., Luo, J., "Correlation-based feature selection for intrusion detection design," Military Communication Conference .1-7.2007.

[6] Debar, H., Becker, M., Siboni, D., "A neural network component for an intrusion detection system," The proceedings of the 1992 IEEE symposium on research in computer security and privacy, Oakland, CA. 240-250.1992.

[7] Lin, M., Miikkulainen, R., Ryan, J., "Intrusion detection with neural networks," Advances in Neural Information Processing Systems. 943-949.1998.

[8] Ryan, J., Lin, M., Miikkulainen, R., "Intrusion detection with neural networks," AI approaches to fraud detection and risk management: papers from the 1997 AAAI workshop, Providence, RI. 72-79.1997.

[9] Ghosh, A.K., Schwartzbard, A., "A study in using neural networks for anomaly and misuse detection," The proceeding on the 8th USENIX security symposium. 1999.

[10] Cannady, J., "Artificial neural networks for misuse detection," The proceedings of the 1998 national information systems security conference (NISSC’98).1998.

[11] Mukkamala, S., "Intrusion detection using neural networks and support vector machine," The proceedings of the 2002 IEEE international joint conference on neural networks. 2002.

[12] Tan, K., "The application of neural networks to UNIX computer security," Proceedings of IEEE International Conference on Neural Networks, vol. 1. 476-481.1995.

[13] Ryan, J., Lin, M.J., Miikkulainen, R., "Intrusion detection with neural networks," Advances in Neural Information Processing Systems, 10. 943-949.1998.

[14] Hofmann, A., Schmitz, C., Sick, B., "Rule extraction from neural networks for intrusion detection in computer networks," IEEE International Conference on Systems, Man and Cybernetics, vol. 2. 1259-1265.2003.

[15] Liu, Z., Florez, G., Bridges, S.M., "A comparison of input representations in neural networks: a case study in intrusion detection," Proceedings of the International Joint Conference on Neural Networks (IJCNN’02), vol. 2, Honolulu, HI, USA.1708-1713.2002.

[16] Chan, A.P.F., Ng, W.W.Y., Yeung, D.S., Tsang, E.C.C., "Comparison of different fusion approaches for network intrusion detection using ensemble of RBFNN," Proceedings of 2005 International Conference on Machine Learning and Cybernetics, vol. 6. 3846-3851.2005.

[17] Jiang, J., Zhang, C., Kame, M., "RBF-based real-time hierarchical intrusion detection systems," Proceedings of the International Joint Conference on Neural Networks (IJCNN’03), vol. 2. 1512-1516.2003.

[18] Langley, P., "Selection of relevant features in machine learning. Institute for the Study of Learning and Expertise," Technical Report, 94-3.1994.

[19] Olvera-Lopez, J.A., Carrasco-Ochoa, J.A., Martinez, J.F., Kittler, J., "A review of instance selection methods," Artificial Intelligence Rev 34. 133-134.2010.

[20] Cover, T., Hart, P., "Nearest neighbor pattern classification," IEEE Trans Information Theory 13. 21-27.1967.

[21] Hart, P.E., "The condensed nearest neighbor rule," IEEE Trans Information Theory 14.515-516. 1968.

[22] Chien-Hsing, C., Bo-Han, K., and Fu, C., "The generalized condensed nearest neighbor rule as a data reduction method," Proceeding of the 18th International Conference on Pattern Recognition, IEEE Computer Society, Hong-Kong. 556-559.

[23] Wilson, D.L., "Asymptotic properties of nearest neighbor rules using edited data," IEEE Trans System Man Cybern 2.408-421.1972.

[24] Tomek, I., "An experiment with the edited nearest-neighbor rule," IEEE Trans System Man Cybern 6-6. 448-452. 1976.

[25] Wilson, D.R., Martinez, T.R., "Reduction techniques for instance-based learning algorithms," Mach Learn 38. 257-286.2000.

[26] Riquelme, J.C., Aguilar-Ru íz, J.S., Toro, M., "Finding representative patterns with ordered projections," pattern recognition 36.1009-1018.2003.

[27] Olvera-L ópez, J.A., Carraso-Ochoa, J.A., Mart ínez-Trinidad, J.F., Object selection based on clustering and border objects, In: Kurzynski, M. et al. (Eds.), Computer Recognition Systems 2. ASC 45, Wroclaw, Poland.27-34.2007.

[28] Bezdek, J.C., Kuncheva, L.I., "Nearest prototype classifier designs: an experimental study," International Journal Hybrid Intelligence System 16(12).1445-1473. 2001.

[29] Spillmann, B., Neuhaus, M., Bunke, H., Pekalska, E., Duin, R.P.W. Transforming strings to vector spaces using prototype selection. In: Yeung D.Y.et al. (Eds.), SSPR&SPR, LNCS 4109. Hong-Kong. 287-296.2006.

[30] Mollineda, R.A., Ferri, F.G., Vidal, E., "An efficient prototype merging strategy for the condensed 1-NN rule through class-conditional hierarchical clustering," Pattern Recognition 35. 2771-2782.2002.

[31] Venmann, C.J., Reinders, M.J.T., "The nearest sub-class classifier: a compromise between the nearest mean and nearest neighbor classifier," IEEE Trans Pattern Anal Match Intelligence 27(9). 1417-1429.2005.

[32] Richaroen, T., Lursinsap, C., "A divide-and-conquer approach to the pair wise opposite class-nearest neighbor (POC-NN) algorithm," Pattern Recognition Letter 26(10).1554-1567.2005.

[33] Olvera-Lopez, J.A., Carrasco-Ochoa, J.A., Martinez-Trinidad, J.F., Prototype selection via prototype relevance, in: Ruiz-Shuleloper. J., Kropatch, W.G. (Eds.), CIARP. LNCS 5197, Habana, Cuba. 153-160.2008.

[34] McHugh, J.," Testing intrusion detection system: a critique of the 1998 and 1999 darpa intrusion detection system evaluations as performed by Lincoln laboratory, "ACM Transactions on Information and system security, 3(4): 262-294.2000.

[35] Tavallaee, M., Bagheri, E., Wei, L.u., Ghorbani, A., "A detailed analysis of the KDD CUP 99 dataset," Proceedings of IEEE Symposium on Computational Intelligence in Security and Defense Applications (CISDA). 2009.