Intrusion Detection at Packet Level by Unsupervised Architectures

Álvaro Herrero[1], Emilio Corchado[1], Paolo Gastaldo[2], Davide Leoncini[2], Francesco Picasso[2], and Rodolfo Zunino[2]

[1] Department of Civil Engineering, University of Burgos
C/ Francisco de Vitoria s/n, 09006 Burgos, Spain
{ahcosio, escorchado} @ubu.es

[2] Dept. of Biophysical and Electronic Engineering (DIBE), Genoa University
Via Opera Pia 11a, 16145 Genoa, Italy
{paolo.gastaldo, francesco.picasso, rodolfo.zunino} @unige.it

Abstract. Intrusion Detection Systems (IDS's) monitor the traffic in computer networks for detecting suspect activities. Connectionist techniques can support the development of IDS's by modeling 'normal' traffic. This paper presents the application of some unsupervised neural methods to a packet dataset for the first time. This work considers three unsupervised neural methods, namely, Vector Quantization (VQ), Self-Organizing Maps (SOM) and Auto-Associative Back-Propagation (AABP) networks. The former paradigm proves quite powerful in supporting the basic space-spanning mechanism to sift normal traffic from anomalous traffic. The SOM attains quite acceptable results in dealing with some anomalies while it fails in dealing with some others. The AABP model effectively drives a nonlinear compression paradigm and eventually yields a compact visualization of the network traffic progression.

Keywords: Intrusion Detection System, Network Security, Vector Quantization, Self-Organizing Map, Auto Associative Back Propagation.

1 Introduction

Automatic detection of anomalous traffic is one of the crucial topics in the area of network communication. Intrusion Detection Systems (IDS's) [1] are designed to monitor the traffic in computer networks and generate alerts, or trigger defensive actions, when suspect activities are detected. Nowadays, IDS's have become common elements in modern infrastructures to enforce network policies; nonetheless, some scientific issues remain open in IDS's development and run-time operation.

Today's IDS implementations address either misuse intrusion detection (MID) or anomaly intrusion detection (AID) [1]. MID systems recognize known attack patterns, and typically discriminate normal from malicious traffic by using a knowledge base of rules. MID suffers from two basic drawbacks: the set of rules is susceptible to inconsistencies and continuous updating is required to incorporate unseen attack

patterns. From a different perspective, AID systems embed a model of 'normal' traffic and generate alerts when 'abnormal' events are detected. These techniques do not use sets of rules, and can support time-zero detection of novel attack strategies; however, AID systems require consistent modeling of normal traffic. Accuracy in detection proves indeed the major limitation of such approach [1]. To circumvent that issue, data-driven techniques have been applied to IDS's models; in particular, connectionist models (supervised and unsupervised approaches) have been profitably used. Supervised methods [1], [2], [3] tackle intrusion detection as a binary classification problem (i.e., normal vs. abnormal traffic). They can attain quite accurate results; in fact, the need for data labeling in the set-up phase and the continuous evolution of attack types often lead to a very expensive training process. Unsupervised methods [1], [4], [5] first extract features from traffic data and then apply unlabelled learning methods: the goal is to identify the significant portions of the feature space that support the distribution of normal traffic, whereas outliers will mark abnormal traffic activities. Unsurprisingly, supervised methods outperform unsupervised approaches at identifying known patterns [1]; by contrast, the latter ones prove more robust when coping with unknown attacks in a dynamic scenario, and therefore have been chosen as the scientific baseline for the present research.

This paper tackles the anomaly detection task by analyzing the performance of three different unsupervised paradigms: Vector Quantization (VQ) [6], Self-Organizing Maps (SOM) [7] and Auto-Associative Back-Propagation (AABP) neural networks [8]. These unsupervised paradigms have been previously applied to intrusion detection but the novelty of this paper is based on the issue that they deal for the first time with packet datasets. The VQ model represents a powerful technique to support the basic space-spanning mechanism, i.e. normal traffic vs. anomalous traffic. The SOM focus the intrusion detection task from a similar perspective as it generates a 2D mapping that preserves the topological properties of the input space. Finally, the AABP-based approach tackles the anomaly-detection problem in terms of dimensionality reduction, thus supporting a nonlinear compression that eventually leads to a compact visualization of the network traffic evolution.

The experimental domain involves both normal traffic and anomalous traffic ascribed to the Simple Network Management Protocol (SNMP), which represents one of the top 5 most vulnerable services [9]. Empirical tests involved a dataset previously used in literature for unsupervised analysis [10], [11]. As it has been previously mentioned, a great amount of connectionist models (including VQ, SOM and AABP) have been already applied to intrusion detection. They have been applied to the KDD dataset [12], which contains information about TCP connections. On the contrary, in this work, unsupervised learning is applied to a dataset containing information from the packet level. That is, information extracted from the header of network packets, providing a complementary intrusion detection point of view.

2 A Connectionist-based Framework for IDS

The general scheme for the proposed connectionist-based framework for intrusion detection can be summarized as follows (Fig. 1):

- packets traveling through the network are intercepted by a capture device;
- traffic is coded by a set of features spanning a multidimensional vector space;
- the connectionist model operates on feature vectors and yields as output a suitable representation of the network traffic.

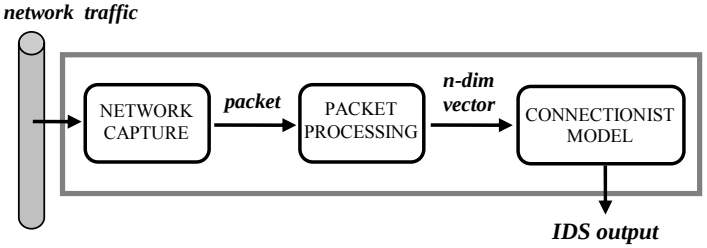


Fig. 1. The connectionist-based IDS framework.

The connectionist model clearly is the actual core of the overall IDS. That module is designed to yield an effective representation of network traffic, thus providing a powerful tool for the automated identification of traffic anomalies. Hence, the effectiveness of the overall approach strictly relates to the successful support to the network supervisor at detecting offending attacks. In the proposed scheme, the connectionist component processes an *n*-dimensional vector that has been previously assembled by a "packet processing" module, which extracts numerical features associated with each network packet. Thus, as a novelty, the proposed IDS operates at the packet level and not at the connection level as other models do [12], [13], [14]. The design of the feature set is indeed a crucial issue [15]. In principle timestamp, source and address port, and protocol can uniquely identify a connection [16]. When dealing with Transmission Control Protocol (TCP) traffic, additional features may be required (e.g. to track connection state [17]); instead, User Datagram Protocol (UDP) traffic can be effectively characterized by a reduced feature set [10].

A connectionist approach appears consistent with the AID problem setting mainly because it allows a system to empirically learn the input-output relationship between raw traffic and subsequent interpretation. The crucial advantage is that the eventual outlier-detection method does not require any *a-priori* analytical formulation of the underlying phenomenon. In principle, any unsupervised method applies to the involved representation process; actually, this work explores the performance of three specific paradigms: Vector Quantization, Self-Organizing Maps and Auto-Associative Back-Propagation.

3 Unsupervised Methods for IDS Implementation

3.1 Vector Quantization

VQ is important in high-dimensional information processing, since represented data and representation codes are expressed in the same domain. The information-

representation paradigm aims to partition the data space into several portions, each one identified by a specific reference codevector, or prototype; hence, VQ is very useful for data representation and compression applications.

The crucial point determining a VQ system performance is represented by the codevector-positioning algorithm. The classical formulation of the VQ training problem can be set as follows. The *n*-dimensional data space is partitioned by a set of prototypes, $V=\{\mathbf{v}_i \in \Re^n,\ i=1,..,N_p\}$, which lie at "significant" positions in the data space; each prototype covers the samples lying within its associate partition. To assign a prototype to each sample, a best-match criterion minimizing a distortion cost is used. Euclidean metrics is usually adopted to measure distortion; hence a data sample, $\mathbf{x} \in \Re^n$, is associated with the prototype, $\mathbf{v}^*(\mathbf{x}) \in V$, that satisfies

$$\mathbf{v}^*(\mathbf{x}) = \arg\min_{\mathbf{v} \in V} \left\{ \|\mathbf{x} - \mathbf{v}\|^2 \right\} \tag{1}$$

The VQ-representation problem implies finding the optimal codebook, V^+, that minimizes the overall distortion:

$$\min E(P) = \int_{\Re^n} \left\| \mathbf{x} - \mathbf{v}^*(\mathbf{x}) \right\|^2 p(\mathbf{x}) d\mathbf{x} \tag{2}$$

The actual sample distribution $p(\mathbf{x})$ is not known *a priori*, hence the integral cannot be computed analytically in any but very peculiar cases. Therefore, one usually resorts to an empirical estimation of the involved distortion: a set of training samples, $X=\{\mathbf{x}_l \in \Re^n,\ l=1,..,N_d\}$, drives vector positioning to minimize the empirical cost:

$$\min\ \hat{E}(V) = \frac{1}{N_p} \sum_{i=1}^{N_p} \left\| \mathbf{x}_i - \mathbf{v}^*(\mathbf{x}_i) \right\|^2 \tag{3}$$

In this research, the VQ training algorithm is based on the "Plastic Neural Gas" (PGAS) [6] model for neural network training. The PGAS algorithm is an adaptive version of the conventional 'neural gas' framework [18], to which it added the ability of dynamically creating and deleting prototypes. The PGAS method is guaranteed to converge in a finite number of steps and, as compared with the neural gas approach, does not suffer from the problem of "dead vectors."

3.2 Self-Organizing Maps

The well known Self-Organizing Map (SOM) [7] is composed of a discrete array of *L* nodes arranged on an *N*-dimensional lattice and it maps these nodes into *D*-dimensional data space while maintaining their ordering. The dimensionality, *N*, of the lattice is normally smaller than that of the input data. Thus, the SOM provides low dimensional representations of multi-dimensional datasets while preserving the topological properties of the input space. The SOM is based on a type of unsupervised learning called competitive learning; an adaptive process in which the neurons in a neural network gradually become sensitive to different input categories, sets of samples in a specific domain of the input space [19]. This can be seen in the SOM update of neighbourhood neurons:

$$w_k(t+1) = w_k(t) + \alpha(t)\eta(v,k,t)\big(x(t) - w_v(t)\big) \quad (4)$$

where w_v is the winning neuron, α the learning rate of the algorithm, $\eta(v,k,t)$ is the neighbourhood function where *v* represents the position of the winning neuron in the lattice and *k* the positions of the neurons in the neighbourhood of this one, x is the input to the network,

To evaluate the adaptation quality of the map to the dataset, two different measures have been used: topographic [20] and quantization [21] error.

3.3 Auto-Associative Back-Propagation Networks

Auto-Associative Back-Propagation (AABP) networks constitute an unsupervised variant of the general Multi-Layer Perceptron (MLP) model, which belongs to the feedforward class of neural networks [23]. A conventional MLP includes three layers (input, 'hidden' and output), and associates an input vector, $\boldsymbol{x} \in \Re^D$, with an output vector, $\boldsymbol{y} \in \Re^Q$, computed as:

$$y_q(\mathbf{x}) = w'_{q,0} + \sum_{u=1}^{N_h}\left[w'_{u,q} \cdot \sigma\left(w_{u,0} + \sum_{k=1}^{D} w_{u,k} x_k \right)\right]; q = 1,\ldots,Q \quad (5)$$

where $\sigma()$ is the sigmoidal function, N_h is the depth of the sigmoid series expansion, and *W* represents the coefficients of the weights for the interconnections between the layers. The empirical Back-Propagation (BP) algorithm [23] drives the weights, *W*, so that the network best reproduces the desired mapping over a given training set.

The AABP model forces target outputs to coincide with the network inputs: $\mathbf{t} \equiv \mathbf{x}$., hence the network should replicate the training sample distribution Since the hidden layer is typically smaller than the input/output ones, the goal is to reduce data dimensionality. At run-time, an AABP network associates each input vector with the 'coding' values computed by the hidden neurons, and therefore supports a (lossy) compression of input data into a lower-dimensional space. The sophisticated AABP model proposed in [8] involves a Non-Linear Principal Component Analysis (NLPCA) architecture.

The crucial difference from classical AABP is that the mapping and reconstruction sections include an additional layer of neurons. The NLPCA architecture retains the universal approximation ability of BP networks [8]. At run-time, the five-layer resulting network operates in the same way as a three-layer AABP: the outputs of the 'coding' layer yield the low-dimensional representation of each input vector.

The increased power of representation conveyed by the NLPCA augmentation is remarkable. The main advantage is that the compressed representation does not relate to any linear model (as Principal Component Analysis [22]). Non-Linear techniques fit those domains in which a non-linear representation best encompasses the observed empirical phenomenon.

4 Unsupervised Connectionist Methods for IDS's

The main demonstration domain of the present research involves traffic anomalies within SNMP, which is a part of the Transmission Control Protocol/Internet Protocol (TCP/IP) protocol suite. SNMP supports the exchange of management information between network devices at the application layer. SNMP data are liable to attacks that may compromise system security [9]. This paper addresses two types of attacks relying on the SNMP protocol:

- SNMP port scan/sweep: a port scan attempts to count the services running on a group of machines by probing ports for response. A port sweep provides information on security weaknesses.
- MIB information transfer: the Management Information Base (MIB) is a collection of information concerning managed devices, including sensitive data. SNMP is used to access MIB objects.

4.1 The Feature Set

The eventual network-based IDS for the detection of SNMP anomalous traffic is structured as shown in Figure 1. The "feature extraction" component (Section 2) generates feature vectors by working out information contained in the packet header. In the present research, network packets are characterized by using the set of features that already proved to be effective for detection of anomalous SNMP traffic [10]. These features can be listed as follows:

- Protocol ID: an integer number that identifies the protocol of the packet.
- Source port: the port number of the device that sent the packet.
- Destination port: the port number of the host to which the packet is sent.
- Size: the packet size (in Bytes).

The four-dimensional space feeds the connectionist component, which is entitled to embed the intrusion-detection task. In the following, the set-up of the VQ-based system and the AABP-based system are introduced.

4.2 Set-up of the Unsupervised Connectionist Models

The VQ paradigm represents a powerful tool to tackle the crucial task that characterizes an AID model: the definition of the 'normal' traffic profile. Thus, the VQ-based IDS is designed to operate as a 'smart' traffic analyzer, which automatically identifies anomalous traffic, i.e. network packets that do not belong to the normal profile. The training phase uses empirical data to configure the prototype set V^+, which should single out the normal profile in the 4-dimensional space that characterizes the network packets. At run-time, the VQ-based component classifies the incoming network packets according to the normal profile defined in the training phase; thus, apposite alarms are generated when a network packet lies in a sector of the 4-dim space that is not associated to the normal profile.

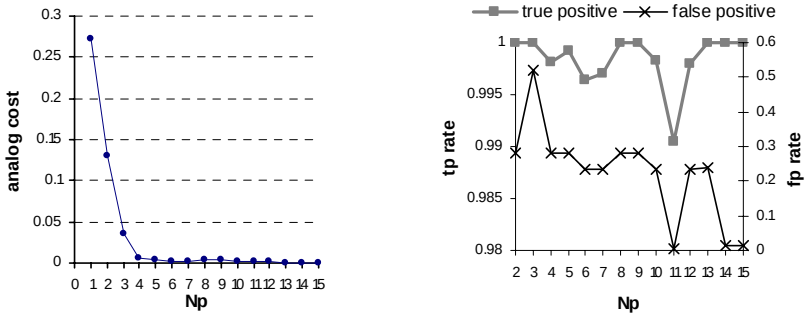
As compared with the VQ paradigm, AABP neural networks represent an intriguing alternative for unsupervised learning, especially when considering its non-linear formulation [8]. The present research exploits the NLPCA architecture to generate a two-dimensional representation of the network traffic by starting from the four-dimensional space defined by the feature set. Hence, the AABP neural network supports the mapping of raw data extracted from traffic sources into an intuitive visual representation.

5 Experimental Results

The three unsupervised approaches to IDS development were tested by using the data set used in [10]. This data set contained network packets captured from UDP traffic, as SNMP uses UDP as the transport protocol for passing data between managers and agents. Hence, the data set included only packets using UDP as transport layer and IP as network layer, and a total of 5866 samples (i.e. network packets) spanned a four-dimensional feature space.

5.1 VQ Paradigm

The PGAS training algorithm drove the unsupervised partitioning of the four-dimensional space. The present research exploited the plasticity feature of PGAS to properly size the cardinality of the prototype set; to this purpose, Figure 2.a reports the results obtained by the training phase. The graph gives on the *x*-axis the cardinality, N_p, of the prototype set, which dynamically grows; the corresponding analog cost is plotted on the *y*-axis. Figure 2.a shows that the analog cost dramatically decreases in the range $1 \leq N_p \leq 4$; then, for $N_p > 4$ the curve is characterized by an asymptotic behavior. $N_p = 14$ appears to be the suitable cardinality of the eventual prototype set V^+, as the curve exhibits small oscillations for $6 \leq N_p \leq 12$.



(a) (b)

Fig. 2. Results obtained by the PGAS algorithm: (a) training phase; (b) test results.

To provide a qualitative assessment, Figure 2.b presents the results obtained on the test set by displaying the true-positive (tp) rate and the false-positive (fp) rate for the

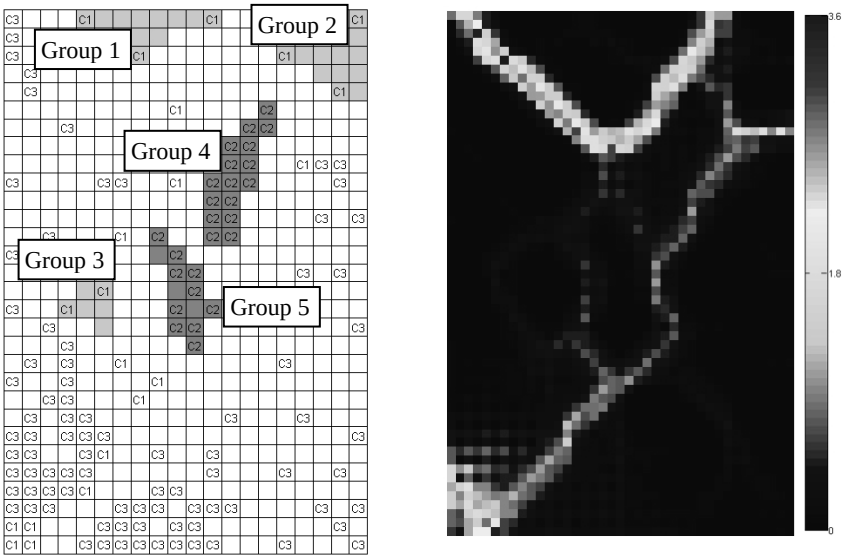
different cardinalities of the prototype set. The graph clearly shows that the best performance (tp = 100% and fp = 0.01%) is attained with $N_p \geq 14$. Thus, the PGAS algorithm proved to be a powerful tool to tackle anomalous traffic detection.

5.2 Self-Organizing Map

The SOM was applied to the previously described data set. A 30x20 neuron lattice was generated to perform the dimensionality reduction. The mapping obtained after training is shown in Figure 3. For a better understanding of results, the neurons have been labeled (See Fig. 3.a) with class information. Each instance in the dataset includes the class value according to: C1 - normal traffic, C2 - port sweeps and C3 - MIB transfer.

As can be seen in Figure 3.a, there are 3 groups of neurons (Groups 1, 2 and 3) modeling most of the normal traffic (C1). All the packets related to the port sweeps (C2) are identified by neurons constituting Groups 4 and 5. The rest of the neurons identify packets related to normal traffic as well as packets related to the MIB information transfer (C3). Figure 3.b shows the associated U-matrix.

The quality measures associated to the results shown in Figure 3 are: quantization error= 0.011 and topographic error= 0.2.



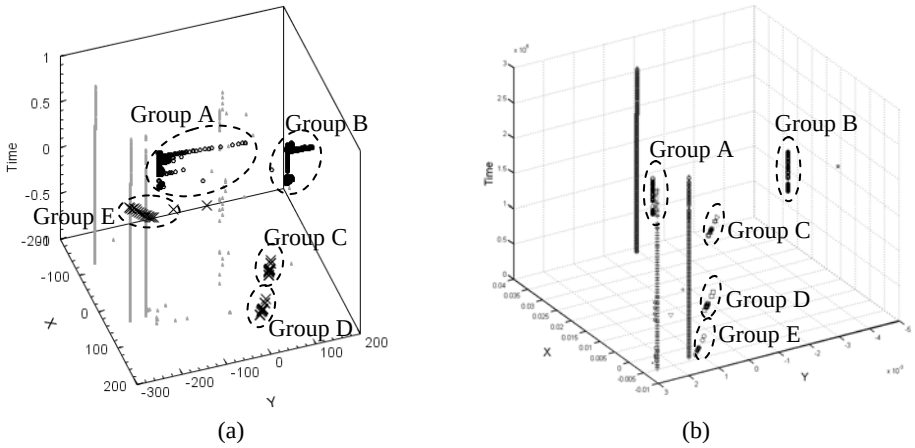
(a) (b)

Fig. 3. Best results obtained by the SOM: (a) labeled map and (b) associated U-matrix.

5.3 AABP Paradigm

According to the set-up introduced in Section 4, the eventual AABP-based IDS was trained to map the original four-dimensional space into a two-dimensional space for

an intuitive visualization of the traffic progress. In the experiments presented here, the configuration of the AABP network included a number of 30 nodes in the hidden layers (coding and reconstruction), while of course the number of neurons in the middle layer was N_h=2. Although theoretical studies did not succeed in providing any established design criterion to set the number of a network's hidden nodes, the literature provides practical criteria [24] for dimensioning a network size, in order to ensure prediction accuracy while minimizing the risk of overfitting training data. In summary, the architecture of the overall AABP network was set as follows: 4 nodes in the input layer, 30 nodes in the compression layer, 2 nodes in the coding layer, 30 nodes in the decompression layer, and 4 nodes in the output layer.



(a) (b)

Fig. 4. Traffic visualization: (a) NLPCA projection; (b) PCA projection.

Figure 4 shows the projection obtained by NLPCA (Fig. 4.a) and PCA (Fig. 4.b). The graph gives on the x-axis and y-axis the outputs of the two neurons in the middle layer, i.e. the compressed representation of the signal; besides, the z-axis gives the time evolution of the packets. In Figure 4.a, gray markers characterize the normal traffic, while black crosses are used to mark port sweeps and black circles are used to mark the MIB information transfer. One can easily note that the proposed NLPCA-based system manages to identify the two anomalous situations contained in the data set: the MIB information transfer (Groups A and B in Figure 4.a) is identified due to its orthogonal direction with respect to the normal traffic and to the high density of packets; the sweeps (Groups C, D and E in Figure 4.a) are identified due to their non-parallel direction to the normal one. In this type of visualization, normal traffic is associated to straight lines evolving in parallel directions.

For comparison purposes, PCA was also applied to the same problem. As can be seen in Figure 4.b, PCA was only able to identify the port sweeps (Groups C, D and E), while it failed to detect the MIB information transfer (Groups A and B) because the packets in this anomalous situation evolve in the same way (a parallel direction) as the "normal" one.

Acknowledgments. This research has been partially supported by the MCyT project TIN2004-07033.

References

1. Laskov, P., Dussel, P., Schafer, C., Rieck, K.: Learning Intrusion Detection: Supervised or Unsupervised?. Proc. ICIAP 2005, Cagliari, Italy (2005) 50-57
2. Liao, Y., Vemuri, V.R.: Use of K-nearest Neighbor Classifier for Intrusion Detection. Comput. Security 21(5) (2002) 439-448
3. Sarasamma, S.T., Qiuming, A.Z., Huff, J.: Hierarchical Kohonen Net for Anomaly Detection in Network Security. IEEE Trans. on SMC – part B 35(2) (2005)
4. Zanero, S.: Analyzing TCP Traffic Patterns Using Self Organizing Maps. Proc. Int. Conf. on Image Analysis and Processing, ICIAP 2005, Cagliari, Italy (2005) 83-90
5. Zheng, J., Hu, M.: An Anomaly Intrusion Detection System Based on Vector Quantization. ICIE Trans. on Inf. & Syst. E89-D(1) (2006)
6. Ridella, S., Rovetta, S., Zunino, R.: Plastic Algorithm for Adaptive Vector Quantization. Neural Computing & Applications 7 (1998) 37-51
7. Kohonen, T.: The Self-Organizing Map. Proceedings of the IEEE 78(9), 1464-1480 (1990)
8. Kramer, M.A.: Nonlinear Principal Component Analysis using Autoassociative Neural Networks. AIChE Journal 37(2) (1991)
9. Cisco Secure Consulting: Vulnerability Statistics Report. 2000
10. Corchado, E., Herrero, A., Saiz, J.M.: Detecting Compounded Anomalous SNMP Situations using Unsupervised Pattern Recognition. Proc. ICANN 2005, Springer - LNCS 3697(2) (2005) 905-910
11. Corchado, E., Herrero, A., Saiz, J.M.: Testing CAB-IDS through Mutations: on the Identification of Network Scans. Proc. KES 2006, Springer LNAI 4252 (2006) 433-441
12. Elkan, M.: Results of the KDD'99 Classifier Learning Contest. (1999) online from: http://www-cse.ucsd.edu/users/elkan/clresults.html
13. Lippmann, R., Haines, J.W., Fried, D.J., Korba, J., Das, K.: Analysis and Results of the 1999 DARPA Off-Line Intrusion Detection Evaluation. Proc. 3rd Int. Workshop on Recent Advances in Intrusion Detection, Springer - LNCS, 1907 (2000) 162 -182
14. Sabhnani, M., Serpen, G.: Application of Machine Learning Algorithms to KDD Intrusion Detection Dataset within Misuse Detection Context. Proc. MLMTA 2003 (2003) 623–630
15. Lee, W., Xiang, D.: Information-Theoretic Measures for Anomaly Detection. Proc. 2001 IEEE Symp. on Security and Privacy (2001) 130-143
16. Lee, W., Stolfo, S.J., Mok, K.W.: Mining in a Data-Flow Environment: Experience in Network Intrusion Detection. Proc. 5th ACM International Conference on Knowledge Discovery and Data Mining (KDD-99) (1999) 114–124
17. Lee, W., Stolfo, S.J., Mok, K.W.: Adaptive Intrusion Detection: A Data Mining Approach. Artificial Intelligence Review 14(6) (2000) 533–567
18. Martinetz, T., Berkovich, S.G., Schulten, K.J.: Neural Gas Network for Vector Quantization and its Application to Time-Series Prediction. IEEE TNN 4(4) (1993) 558-569
19. Kohonen, T., Lehtio, P., Rovamo, J., Hyvarinen, J., Bry, K., Vainio, L.: Principle of Neural Associative Memory. Neuroscience 2(6), 1065- 1076 (1977)
20. Kiviluoto, K.: Topology Preservation in Self-Organizing Maps. In: IEEE International Conference on Neural Networks. vol. 1, pp. 294-299 (1996)
21. Kohonen, T.: Self-Organizing Maps. Springer Series In Information Sciences, Vol. 30 Springer-Verlag New York, Inc. (1997)
22. Pearson, K.: On Lines and Planes of Closest Fit to Systems of Points in Space. Philosophical Magazine, Vol. 2(6) (1901) 559-572
23. Rumelhart, D.E., McClelland, J.L.: Parallel Distributed Processing. MIT Press, Cambridge, MA (1986).
24. Widrow, W., Lehr, M.A.: 30 Years of Adaptive Neural Networks: Perceptron, Madaline and Back Propagation. Proc. IEEE. 78(9) (1990) 1415-42