Journal of Advances in Computer Research
Quarterly ISSN: 2008-6148
Sari Branch, Islamic Azad University, Sari, I.R.Iran
(Vol. 4, No. 4, November 2013), Pages: 51-
www.jacr.iausari.ac.ir

A Lattice based Nearest Neighbor Classifier for Anomaly Intrusion Detection

Yazdan Jamshidi[✉1], Hossein Nezamabadi-pour[2]

1) Department of Computer Engineering, Science and Research, Islamic Azad University, Kermanshah, Iran.

2) Department of electrical engineeering, Shahid Bahonar university of Kerman

y.jamshidi@gmail.com; nezam@uk.ac.ir

Received: 2013/05/23; Accepted: 2013/07/28

Abstract

As networking and communication technology becomes more widespread, the quantity and impact of system attackers have been increased rapidly. The methodology of intrusion detection (IDS) is generally classified into two broad categories according to the detection approaches: misuse detection and anomaly detection. In misuse detection approach, abnormal system behavior is defined at first, and then any other behavior is defined as normal behavior. The main goal of the anomaly detection approach is to construct a model representing normal activities. Then, any deviation from this model can be considered as an anomaly, and recognized to be an attack. Recently much more attention is paid to the application of lattice theory in different fields. In this work we propose a lattice based nearest neighbor classifier capable of distinguishing between bad connections, called attacks, and good normal connections. A new nonlinear valuation function is introduced to tune the performance of the proposed model. The performance of the algorithm was evaluated by using KDD Cup 99 Data Set, the benchmark dataset used by Intrusion detection Systems researchers. Simulation results confirm the effectiveness of the proposed method.

Keywords: *Anomaly detection, Nearest Neighbor, Lattice Theory, Positive Valuation Function, KDD Cup 99*

1. Introduction

As networking and communication technology becomes more widespread, the quantity and impact of spammers, system attackers, and criminal enterprises have been increased rapidly. The research in this area has mainly focused on the development of system security mechanisms like firewalls. However, as complete prevention of computer attacks is not possible, many researchers have concentrated their effort on developing novel detection techniques, capable of promptly identifying network attacks. IDS is used to detect any intruder which might have entered into the computers or networks. The methodology of intrusion detection is generally classified into two broad categories according to the detection approaches: misuse detection and anomaly detection. The misuse detection approach (also called signature based detection) [1],[2],[3] is based on extensive knowledge of patterns associated with known attacks or signatures so that even variations of these attacks can be detected. Based on these

signatures, this approach detects attacks through a large set of rules describing every known attack. Signature based intrusion detection systems rely on human intervention to create, test, and deploy the signatures. Thus, it may take hours or days to generate a new signature for an attack which can be too long when dealing with rapidly moving attacks, such as worm propagation. Some effort has been put into automatic signature generation, which does not require human intervention, but these systems are not yet ready for large scale deployment [4]. Although misuse detection systems have the capability of detecting many or all known attack patterns but they are unable to detect novel and unanticipated attacks. The main goal of the anomaly detection (also called profile based detection) approach [5-8] is to construct a model representing normal activities. Then, any deviation from this model can be considered as an anomaly, and recognized to be an attack. Notice that when this approach is employed, it is possible to recognize unforeseen attacks, although in some cases, this approach can lead to a high false attack rate. Due to this potentiality of detecting unknown attacks there has been a fast growing interest in developing new techniques to build models based on normal traffic behavior in the past years.

Profiles of normal behavior can be built with a variety of techniques including statistical methods [9],[10],[11],[12], association rules [13],[14], neural networks [15], computer immunology [16], and specification based methods [17].

At the early stage, the research focus lies in using rule-based expert systems and statistical approaches. But when encountering larger datasets, the results of rule-based expert systems and statistical approaches become worse. Thus, many data mining techniques have been introduced to solve the problem. Among these techniques, the Artificial Neural Network (ANN) is widely used and has been successful in solving many complex practical problems [18].

ANNs have been proposed as alternatives to the statistical analysis component of anomaly detection systems [19],[20]. Brause et al. [21] used a compound method based on rule-based systems and an ANN for credit card fraud detection. Other ANN based credit card fraud detection systems have been undertaken by Hassibi [22], Dorronsoro et al. [23] and Syeda et al. [24]. Wang et al. [20] proposed a new approach called FC-ANN based on ANN and fuzzy clustering to solve the problem and help IDS to achieve a higher detection rate. In [25],[26] SVM was proposed as an approach for novelty detection with a significant success rate. Unfortunately, as noticed by Eskin et al. [27], SVM for novelty detection works under the assumption that the number of normal traffic instances vastly outnumbers the number of anomalies.

Lattice computing is an interesting topic which has been taken into account by several authors. The term lattice computing was introduced recently by Graña [28]. More specifically, lattice computing was defined as the class of algorithms that uses lattice theory either to achieve pattern recognition or to produce generalizations. Lattices are popular in mathematical morphology including image processing applications [29,30]. Moreover, algebraic lattices have been used for modeling associative memories [31]. One way and bidirectional lattice associative memories [32] have been proposed to overcome capacity limitations [33],[34]. Lattices are used implicitly in some neural networks such as fuzzy-ART and min-max [35],[36]. Petridis and Kaburlasos [37] have found inspiration in lattice theory and versions of the ART model and have devised another successful approach to lattice-based computational

intelligence. Moreover, in [38] a fuzzy interval number k-nearest neighbor classifier has been proposed and was successfully applied to predicting annual sugar production.

This paper presents a novel nearest neighbor classification algorithm for the anomaly intrusion detection based on lattice theory. A practical advantage of lattice theory is the ability to model both uncertain information and disparate types of lattice-ordered data [39]. Indeed, our proposed algorithm is capable of dealing with disparate type of data including real vectors, fuzzy sets, symbols, graphs, images, waves and even any combination of the aforementioned data. It can handle both points and intervals. Learning in the proposed algorithm is carried out fast therefore, in many application, when the data is so massive, and the analysis process so time consuming, the proposed algorithm can be a proper choice.

The layout of this paper is as follows. In Section 2 the mathematical background on lattices is reviewed. Section 3 explains our proposed model. Section 4 provides empirical results that demonstrate the performance of our proposed model. Finally, Section 5 summarizes the results of this work.

2. Mathematical Background

A lattice $(L, \leq)$ is a partially ordered set (or, simply, poset) such that any two of its elements $a, b \in L$ have a greatest lower bound $ab = inf\{a, b\}$and a least upper bound $ab = sup\{a, b\}$.The lattice operations $\wedge$ and $\vee$ are also called meet and join, respectively. A lattice $(L, \leq)$ is called complete when each of its subsets has a least upper bound and a greatest lower bound in L [40]. A non-void complete lattice has a least element and a greatest element denoted by O and I, respectively. The inverse $\geq$ of an order relation $\leq$ is itself an order relation. The order $\geq$ is called the *dual* order of $\leq$, symbolically $\leq^{\partial}$. A lattice $(L, \leq)$ can be Cartesian product of N constituent lattices $L_1, \ldots, L_N$ i.e. $L = L_1 \times \cdots \times L_N$. The lattice operations meet and join of product lattice are defined as below:

$$a \wedge b = (a_1, \ldots, a_N) \wedge (b_1, \ldots, b_N) = (a_1 \wedge b_1, \ldots, a_N \wedge b_N) \quad (1)$$

$$a \vee b = (a_1, \ldots, a_N) \vee (b_1, \ldots, b_N) = (a_1 \vee b_1, \ldots, a_N \vee b_N) \quad (2)$$

A valuation on a crisp lattice L is a real-valued function $v: L \to R$ which satisfies $v(a) + v(b) = v(a \vee b) - v(a \wedge b)$, $a, b \in L$. A valuation is called monotone iff $a \leq b$ in L implies $v(a) \leq v(b)$ and positive iff $a < b$ implies $v(a) < v(b)$.

Consider the set R of real numbers. It turns out that $(\overline{R} = R \cup \{-\infty, +\infty\}, \leq)$ under the inequality relation $\leq$ between $a, b \in R$ is a complete lattice with the least element $-\infty$ and the greatest element $+\infty$ [41]. A lattice $(L, \leq)$ is totally ordered if and only if for any $a, b \in L$ either $a \geq b$ or $a < b$. The lattices $(\overline{R}^N, \leq)$ and $([0,1]^N, \leq)$ under inequality relation are not a totally ordered lattice.

We remark that the goal of positive valuation function v is to deal with lattice elements. Choosing a suitable valuation function is problem dependent. Various positive valuation function have been proposed in the literature [42],[43],[44]. In our experiments the data have been normalized in lattice $L = [0,1]^N$, the unit N dimensional

hypercube, by the function $x_{Norm} = \frac{x - x_{\min}}{x_{\max} - x_{\min}}$ where, $x_{\min}$ and $x_{\max}$ stands for respectively the least and greatest attribute values of the data and N is the dimension of the input data. Thus, we propose the following nonlinear positive valuation function.

For all $A \in ([0,1]^N, \leq)$, $\nu(A) = \sum_{i=1}^{N} \sin(a_i)$ (3)

Based on the principle that for $\forall a \in [0,1]$, $a \leq \sin(a)$, it can be easily proved that $\nu(A \vee B) + \nu(A \wedge B) = \nu(A) + \nu(B)$, $A, B \in [0,1]^N$. Furthermore, Figure 1. Shows that the proposed valuation function is a strictly increasing function in lattice $L = [0,1]^N$. Finally the aforementioned function (3) maps the least element of lattice $([0,1]^N, \leq)$ to zero. Thus, it satisfies the conditions of a positive valuation function.

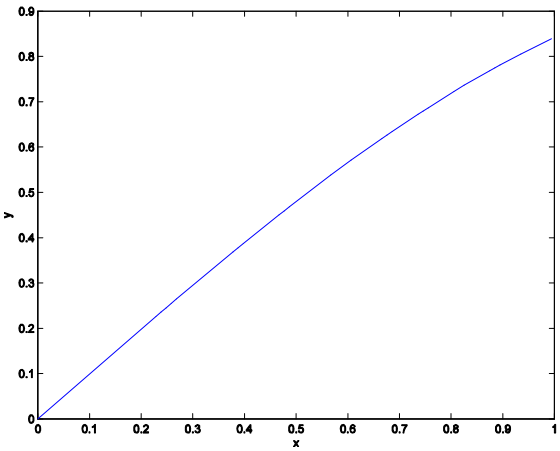


Figure 1. The positive valuation function y=sin(x).

In some applications, we may want to numerically express the differences of two objects by means of the distance of the corresponding sets. It follows that if two sets A and B have a large similarity, then they will have a small distance. A formal definition of metric distance is given in [45]. Here we introduce a useful metric distance function capable to deal with lattice elements. For two N-dimensional vectors $A = (a_1, ..., a_N)$ and $B = (b_1, ..., b_N)$ the following metric distance $d : L \times L \to [0,1]$ in a lattice L is defined as follows:

$$d(A, B) = \nu(A \vee b) - \nu(A \wedge B) = \sum_{i=1}^{N} [\nu(a_i \vee b_i) - \nu(a_i \wedge b_i)].$$

For more details, we refer the reader to [40,42].

3. The proposed model

This section presents a nearest neighbor classifier based on lattice theory capable of detecting large-scale attacks in real-time. The nearest neighbor classifier assigns to a test sample a class label of its closest neighbor using a metric distance such as Euclidean distance typically used in conventional nearest neighbor. Our proposed

Classifier uses the metric distance introduced in the previous section to compute the distance of a given test sample with a predefined normal profile. If the distance is less than a user defined threshold then the test pattern will be labeled as "normal" situation otherwise a network intrusion is reported. One of its important properties is the ability of dealing with disparate type of data including real vectors, fuzzy sets, symbols, graphs, images, waves and even any combination of the aforementioned data and this shows the ability of the algorithm in combining different type of data. Furthermore, the proposed model can cope with both points and intervals. The inherent speed of the algorithm for training, detecting attacks timely and having a simple structure to implement are other benefits of this approach. The algorithm is described in the following:

The proposed algorithm

S0. select a new datum P as a normal profile
(it can be selected randomly or based on some information).

S1. Compute the distance $d_0 = d(P, x)$ where x is a new test pattern.

S2. If $d_0 \leq \delta$,where δ is a user defined threshold, then
assign the label 'normal" to x
otherwise,
report a network intrusion.

4. Experimental results and discussion

4.1 Data set description

The principle interest of this work is to benchmark the performance of the proposed model by using KDD Cup1999 Data Set [46] which is generally recognized and widely used by researchers working on IDS field. The main task for the KDD 99 classifier learning contest was to provide a predictive model able to distinguish between normal and intrusion or attacks connections in a computer network. KDD training dataset includes approximately 4,900,000 single connection records each of which contains 41 features and is labeled as either normal or an attack, with exactly one specific attack type. The simulated attacks fall in one of the following four categories: DOS (denial of service attacks), R2L (remote user unauthorized access to the local), U2R (illegal access to super-privilege attacks) and Probing (surveillance and other probing attacks). In our experiment we have used about 1,000,000 of the data and we have labeled all the data as being either normal or abnormal. The distribution of the data is shown in table 1.

Table 1. Distribution of the KDD dataset.

Data type	Normal	Attack
Number of data	595795	452778

4.2 Accuracy Measure

Estimating classifier accuracy is important in that it allows one to evaluate how accurately a given classifier will label the test data. To evaluate the effectiveness of a classifier for problems with skewed classes, the standard precision, recall, specificity, F_1 measure and total accuracy are used here. The evaluation measures which are used in approach for testing process in our research work could be defined as follows [47]:

True Positive (TP): This states the number of normal connection records correctly classified as normal.

True Negative (TN): This states the number of attack records correctly classified as attack.

False Positive (FP): This states the number of normal records classified as attack.

False Negative (FN): This states the number of attack records classified as normal.

In a classification task, the precision for a class is defined to be the ratio of true positives divided by the total number of elements labeled as belonging to the positive class. Recall is the ratio of true positives divided by the total number of elements that actually belong to the positive class. Specificity is the percentage of negative labeled instances that were predicted as negative. The F_1 measure tries to balance both precision and recall indeed, it is the harmonic mean between precision and recall. Finally accuracy indicates the percentage of predictions that were correct. The precision, recall, specificity, F_1 measure and accuracy are respectively defined in the following form:

$$precision = \frac{TP}{TP + Fp} \tag{4}$$

$$recall = \frac{TP}{TP + FN} \tag{5}$$

$$specificity = \frac{TN}{TN + FP} \tag{6}$$

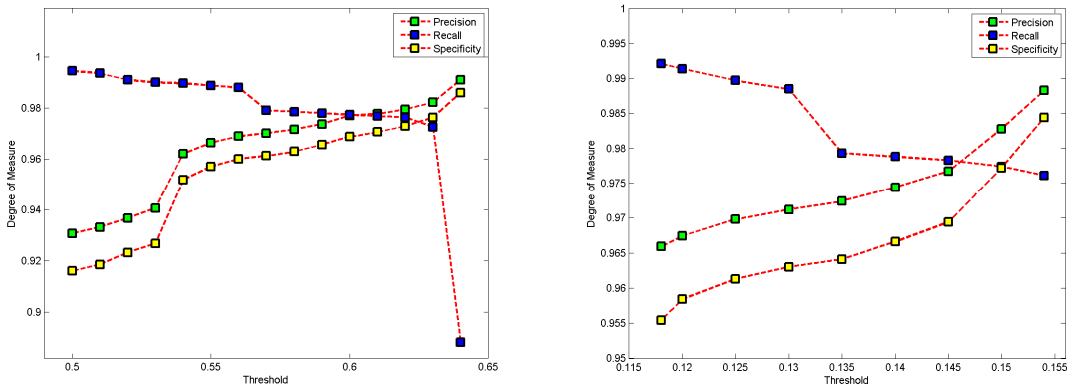
$$F_1 = 2 \times \frac{precision \times recall}{precision + recall} \tag{7}$$

$$accuracy = \frac{Tp + TN}{TP + TN + FP + FN} \tag{8}$$

4.3 Results

In this section we detail the overall results of the proposed framework. In Figure 1(a) and (b), the precision, recall and specificity of the proposed method and conventional *k-NN* are depicted. We can see that both models have the same behavior as the values of threshold parameter δ change. Larger values of δ lead to low recall and higher precision and specificity while smaller values for that result in higher recall and lower precision and specificity.

It should be mentioned that under the anomaly detection perspective, recall should be as high as possible, but precision and specificity should also be high. Indeed for the sake of optimization recall, precision and specificity must be kept in equilibrium. Figure 2 shows the relation between recall, precision and specificity for the conventional *k-NN* and our proposed model.



(a) (b)

Figure 2. Relation between recall, precision and specificity (a) K-NN (b) Proposed Model

In order to evaluate the performance of the proposed method more accurately F1 measure which takes into account both precision and recall and classification accuracy are considered. Tables 2 and 3 show the comparison of the experimental results of the proposed model endowed with different positive valuation functions [42, 43] with those of produced by the conventional *K-NN* and fuzzy *K-NN* algorithms for different values of threshold parameter δ. In this work the proposed method endowed with linear valuation function $\nu(x) = x$, nonlinear valuation functions $\nu(x) = \frac{1}{1+e^{-x}}$, $\nu(x) = \arctan(x)$ and $\nu(x) = \sin(x)$ are denoted respectively by *LBK-NN*$_x$, *LBK-NN*$_e$, *LBK-NN*$_a$ and *LBK-NN*$_s$ where, *LBK-NN* stands for Lattice Based *K-NN*. For each algorithm 10 best results are reported for different values of parameter δ raised in steps of 0.0002

Note that depending on the algorithm the values of parameter δ lies in different intervals.

Table 2. Different measures used to evaluate the performance of the methods.

Conventional *K-NN*			**Fuzzy *K-NN***			***LBK-NN*$_x$**		
Threshold	*F$_1$ measure*	*Accuracy*	*Threshold*	*F$_1$ measure*	*Accuracy*	*Threshold*	*F$_1$ measure*	*Accuracy*
0.0544	0.9767	0.9739	0.9468	0.9775	0.9758	0.1460	0.9795	0.9770
0.0546	0.9773	0.9743	0.9470	0.9784	0.9757	0.1462	0.9795	0.9770
0.0548	0.9773	0.9744	0.9472	0.9782	0.9756	0.1464	0.9795	0.9770
0.0550	0.9775	0.9747	0.9474	0.9780	0.9753	0.1466	0.9796	0.9770
0.0552	0.9779	0.9752	0.9476	0.9780	0.9752	0.1468	0.9796	0.9770
0.0554	0.9780	0.9753	0.9478	0.9775	0.9748	0.1470	0.9796	0.9769
0.0556	0.9782	0.9755	0.9480	0.9784	0.9746	0.1472	0.9796	0.9770
0.0558	0.9783	0.9756	0.9482	0.9772	0.9744	0.1474	0.9795	0.9770
0.0560	0.9784	0.9757	0.9484	0.9768	0.9740	0.1476	0.9795	0.9769
0.0562	0.9785	0.9758	0.9486	0.9763	0.9733	0.1478	0.9795	0.9770

Table 3. Different measures used to evaluate the performance of the methods.

LBK-NN$_e$			***LBK-NN$_a$***			***LBK-NN$_s$***		
Threshold	*F_1 measure*	*Accuracy*	*Threshold*	*F_1 measure*	*Accuracy*	*Threshold*	*F_1 measure*	*Accuracy*
0.0762	0.9794	0.9770	0.0376	0.9798	0.9772	0.1522	0.9815	0.9788
0.0764	0.9795	0.9770	0.0378	0.9796	0.9772	0.1524	0.9816	0.9790
0.0766	0.9795	0.9771	0.0380	0.9798	0.9773	0.1526	0.9819	0.9791
0.0768	0.9799	0.9774	0.0382	0.9800	0.9773	0.1528	0.9818	0.9792
0.0770	0.9798	0.9774	0.0384	0.9799	0.9773	0.1530	0.9819	0.9793
0.0772	0.9799	0.9774	0.0386	0.9799	0.9773	0.1532	0.9820	0.9794
0.0774	0.9801	0.9775	0.0388	0.9799	0.9773	0.1534	0.9820	0.9794
0.0776	0.9799	0.9776	0.0390	0.9798	0.9773	0.1536	0.9822	0.9795
0.0778	0.9799	0.9775	0.0392	0.9798	0.9772	0.1538	0.9822	0.9796
0.0780	0.9796	0.9774	0.0394	0.9789	0.9764	0.1540	0.9822	0.9796

In Table 3, the minimum, maximum and average for the classification accuracy and F_1 measure on the entire experiments have been shown. In other words first, the minimum, maximum and average of each column of the Tables 1 and 2 have been calculated, and then the results for the algorithm with better performance have been shown in bold. As it can be seen, the *LBK-NN$_s$* algorithm in all cases has achieved better results and outperforms other methods.

Table 3. The performance of the models

Algorithm\Measure	**F_1 Measure**			**Accuracy**		
	Min	Average	Max	Min	Average	Max
Conventional *K-NN*	0.9767	0.9778	0.9785	0.9739	0.9750	0.9758
Fuzzy *K-NN*	0.9763	0.9776	0.9784	0.9733	0.9749	0.9758
LBK-NN$_x$	0.9795	0.9795	0.9796	0.9769	0.9770	0.9770
LBK-NN$_e$	0.9794	0.9798	0.9801	0.9770	0.9773	0.9776
LBK-NN$_a$	0.9789	0.9797	0.9800	0.9764	0.9772	0.9773
LBK-NN$_s$	**0.9815**	**0.9819**	**0.9822**	**0.9788**	**0.9793**	**0.9796**

5. Conclusion

This study proposed a nearest neighbor classifier based on lattice theory capable of detecting large-scale attacks in real-time. Experimental results on the well-known KDD Cup 1999 data set demonstrate that the proposed method can effectively detect anomalies with high detection rate. Furthermore the performance of the proposed model can be improved by tuning valuation function ν. The inherent speed of the algorithm for training is another advantage of the algorithm.

6. Refrences

[1] T. F. Lunt, A. Tamaru, F. Gilham, R. Jagannathan, C. Jalili, P. G. Neumann, H. S. Javitz (1992) A Valdes and T. D. Garvey "A Real Time Intrusion Detection Expert System (IDES)- Final Technical Report," Computer Science Laboratory, SRI International, Menlo Park, California

[2] K. Illgun, R. Kemmerer, P. A. Porras, State Transition Analysis : A rule-based intrusion detection approach, IEEE Transaction on Software Engineering pp 181-199 (1995)
[3] K. Illgun, USTAT: A Real-Time Intrusion Detection System for UNIX, in Proc. Of the 1993 Symposium Security and Privacy, pp. 16-28, May 24-26 (1993)
[4] T. Shon, J. A. Moon, hybrid machine learning approach to network anomaly detection, Information Sciences 177: 3799–3821(2007)
[5] H. S. Javitz, A. Valdes, The NIDES Statistical Component Description and Justification, Annual report, SRI International, 333 Ravenwood Avenue, Menlo Park, CA 94025 (1994)
[6] P. A. Porras, P. G. Neumann, EMERALD: Event Monitoring Enabling Responses to Anomalous Live Disturbances," 20th NISSC (1997)
[7] K. Sequeira, M. Zaki, ADMIT: Anomaly based Data Mining for Intrusions", SIGKDD, Edmonton, Alberta, Canada (2002)
[8] W. Lee, S. Stolfo Data Mining Approaches for Intrusion Detection," In Proc. of the 7th USENIX Security Symposium, San Antonio, Texas, January 26-29 (1998)
[9] D. Anderson, T. F. Lunt, H. Javits, A. Tamaru, A. Valdes Detecting Unusual Program Behavior Using the Statistical Component of the Next-generation Intrusion Detection System (NIDES), Technical Report SRI-CSL-95-06, SRI International (1995)
[10] J. B. D. Cabrera, B. Ravichandran, R. K. Mehra, Statistical traffic modeling for network intrusion detection, 8th International Symposium on Modeling Analysis and Simulation of Computer and Telecommunication Systems (2000)
[11] P. Neumann, P. Porras, Experience with emerald to date, Proceedings of 1st USENIX Workshop on Intrusion Detection and Network Monitoring (1999)
[12] P. Porras, P. G. Neumann, Emerald: event monitoring enabling responses to anomalous live disturbances, Proceedings of 1^{9th} National Information Systems Security Conference (1997)
[13] D. Barbara, N. Wu, S. Jajodia, Detecting novel network intrusions using bayes estimators, Proceedings of the ^{1st} SIAM Conference on Data Mining.
[14] D. Barbara, J. Couto, N. Wu, S. Jajodia, ADAM: detecting intrusion by data mining, Proceedings of the 2nd Annual IEEE Information Assurance Workshop (2001)
[15] A. K. Ghosh, A. Schartzbard, A study in using neural networks for anomaly and misuse detection, Proceedings of USENIX Security Symposium (2002)
[16] S. Forrest, S. Hofmeyr, A. Somayaji, T. Longstaff, A sense of self for Unix processes, Proceedings of IEEE Symposium on Security and Privacy (1996)
[17] R. Sekar, A. Gupta, J. Frullo, T. Shanbhag, A. Tiwari, H. Yang, S. Zhou, Specification-based anomaly detection: a new approach for detecting network intrusions, Proceedings of ACM Conference on Computer and Communication Security (2002)
[18] G. Wang, et al., A new approach to intrusion detection using artificial neural networks and fuzzy clustering'', Original Research Article Expert Systems with Applications, 37(9), pp. 6225–6232 (2010)
[19] H. Debar, B. Dorizzi An Application of a Recurrent Network to an Intrusion Detection System. In *Proceedings of the International Joint Conference on Neural Networks.* pp. (II)478-483 (1992)
[20] J. Ryan, M. Lin, R. Miikkulainen, Intrusion Detection with Neural Networks. *AI Approaches to Fraud Detection and Risk Management: Papers from the 1997 AAAI, Workshop (Providence, Rhode Island)*, pp. 72-79. Menlo Park, CA: AAAI (1997).
[21] R. Brause, T. Langsdorf, M. Hepp, Credit card fraud detection by adaptive neural data mining'', Proceedings of the 11th IEEE International Conference on Tools with Artificial Intelligence, pp. 103–106 (1999)
[22] K. Hassibi, Detecting payment card fraud with neural networks, In Business Applications of Neural Networks, P.J.G. Lisboa, A. Vellido and B. Edisbury, Eds., World Scientific, Singapore (2000)
[23] J. R. Dorronsoro, F. Ginel, C. Sanchez, C. S. Cruz, Neural fraud detection in credit card operations'', IEEE Transactions on Neural Networks, 8: pp. 827–834 (1997)
[24] M. Syeda, Y. Q. Zhang, Y. Pan, Parallel granular neural networks forfast credit card fraud detection'', Proceedings of the 2002 IEEE International Conference, 1: pp. 572–577 (2002)
[25] B, Schölkopf, J. C. Platt, J. Shawe-taylor, A. J. Smola, R. C. Williamson, Estimating the support of a high-dimensional distribution. Neural Computation, 13, 1443–1471 (2001)

[26] D. Tax, R, Duin, Data domain description using support vectors. In Proceedings of the european symposium on artificial neural networks pp. 251–256 (1999)
[27] E. Eskin, A. Arnold, M. Prerau, L. Portnoy, S. Stolfo, A geometric framework for unsupervised anomaly detection: Detecting intrusions in unlabeled data. In Applications of Data Mining in Computer Security. Kluwer (2002)
[28] M. Graña, A brief review of lattice computing, in: Proceedings *of the World Congress on Computational Intelligence*, Hong Kong, China, June 1–6, pp. 1777–1781 (2008)
[29] G. X. Ritter, J. N. Wilson, Handbook of Computer Vision Algorithms in Image Algebra. CRC Press, Boca Raton (1996)
[30] M. Graña, A. M. Savio, M. García-Sebastián, E. Fernádez, A lattice computing approach for on-line fMRI analysis, Image and Vision Computing. 28 (7): 1155—1161 (2010)
[31] G. X. Ritter, G. Urcid, Learning in Lattice Neural Networks that Eemploy Dendritic Computing , Studies in Computatonal Intelligence (SCI), 67: 25-44 (2007)
[32] T. Kohonen Correlation matrix memory, *IEEE Trans. Computers* C-21 353-359 (1972)
[33] J. J. Hopfield, Neurons with graded response have collective computational properties like those of two state neurons, Proc. of the National Academy of Sciences, 81: 3088—3092 (1984)
[34] B, Kosko, Bidirectional associative memories, IEEE Trans. Systems, Man, and Cybernetics 18 (1) 49—60 (1988)
[35] G. A. Carpenter, S. Grossberg, D. B. Rosen, Fuzzy ART: Fast stable learning and categorization of analog patterns by an adaptive resonance system. Neural Networks 4(6), pp. 759—771 (1991)
[36] P. K. Simpson, Fuzzy min-max neural networks - part1: Classification. IEEE Trans Neural Networks 3(5), pp. 776—786 (1992)
[37] V. Petridis, V. G. Kaburlasos, Fuzzy lattice neural network (FLNN): A hybrid model for learning. IEEE Trans Neural Networks 9(5), pp. 877—890 (1998)
[38] V. Petridis, V. G. Kaburlasos, FINkNN: A Fuzzy Interval Number k-Nearest Neighbor Classifier for Prediction of Sugar Production from Populations of Samples, Journal of Machine Learning Research 3: 1479-1499 (2003)
[39] V. G. Kaburlasos, V. Petridis, Fuzzy Lattice Neurocomputing (FLN): a novel connectionist scheme for versatile learning and decision making by clustering, International Journal of Computers and Their Applications 4 (3), pp. 31—43 (1997)
[40] G. Birkhoff, Lattice Theory, third ed., American Mathematical Society Providence, RL (1967)
[41] V. G. Kaburlasos, T. Pachidis, A Lattice-Computing ensemble for reasoning based on formal fusion of disparate data types, and an industrial dispensing application, Information Fusion (In Press) (2011)
[42] V. G. Kaburlasos, I. N. Athanasiadis, P. A. Mitkas, Fuzzy Lattice Reasoning (FLR) classifier and its application for ambient ozone estimation, International Journal of Approximate Reasoning 45: 152-188 (2007)
[43] H. Liu, S. Xiong, Z. Fang, FL-GrCCA: A granular computing classification algorithm based on fuzzy lattices. Computers and Mathematics with Applications 61: 138–147 (2011)
[44] Y. Jamshidi-Khezeli, H. Nezamabadi-pour, Fuzzy Lattice Reasoning for Pattern Classification Using a New Positive Valuation Function, Advances in Fuzzy Systems (2012)
[45] W. Rudin, principles of mathematical analysis, third ed., McGraw-Hill, New York (1976)
[46] KDD Cup 1999. Available on: http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html, Ocotber 2007.
[47] T. Zhang, R. Ramakrishnan, M. Livny, BIRCH: An Efficient Data Clustering Method For Very Large Databases, Technical Report, Computer Sciences Dept., Univ. of Wisconsin-Madison (1996)