Discriminating and Visualizing Anomalies Using Negative Selection and Self-Organizing Maps

Fabio A. González
fagonzalezo@unal.edu.co

Juan Carlos Galeano
jcgaleanoh@unal.edu.co

Diego Alexander Rojas
darojaspa@unal.edu.co

Angélica Veloza-Suan
avelozas@unal.edu.co

Lab. de Investigación en Sistemas Inteligentes
Dept. de Ingeniería de Sistemas e Industrial
Universidad Nacional de Colombia
Bogotá, Colombia

ABSTRACT

An immune inspired model that can detect anomalies, even when trained only with normal samples, and can learn from encounters with new anomalies is presented. The model combines a negative selection algorithm and a self-organizing map (SOM) in an immune inspired architecture. The proposed system is able to produce a visual representation of the self/non-self feature space, thanks to the topological 2-dimensional map produced by the SOM. Some experiments were performed on classification data; the results are presented and discussed.

Categories and Subject Descriptors

I.2.8 [**Artificial Intelligence**]: Problem Solving, Control Methods, and Search—*Heuristic methods,Artificial immune systems*
; H.5.m [**Information Systems**]: Miscellaneous—*Data visualization,Anomaly visualization*
; I.2.6 [**Artificial Intelligence**]: Learning—*Connectionism and neural nets,SOM*
; I.5.2 [**Pattern Recognition**]: Design Methodology—*Classifier design and evaluation*

General Terms

Algorithms

Keywords

artificial immune systems, negative selection, self-organizing maps, anomaly detection, anomaly visualization

GECCO'05, June 25–29, 2005, Washington, DC, USA.

1. INTRODUCTION

From a simplistic point of view, the anomaly detection problem could be seen as a classification problem: a set of samples has to be classified into normal and abnormal, so it is possible to train a classifier, using a conventional classification algorithm, which will do the discrimination task. However, the problem is much more complex than that. First, in many real world problems, only normal samples are available at the training phase, therefore, it is impossible to train a conventional classifier, since the training algorithm needs both normal and abnormal samples to build a model that could discriminate them. Second, the set of possible anomalies could be potentially infinite. Let us consider, for instance, the computer virus problem: at a given time, we know a set of viruses that have attacked computer systems previously, every time a new unknown virus attacks, it becomes part of the set of known viruses, but as far as new viruses continue to be built, this process will never end. In this context, it is necessary a system capable of detecting known and unknown anomalies and able to learn from encounters with new anomalies.

This paper presents a model that exhibits some of the characteristics discussed before: it can detect anomalies, even when trained only with normal samples, and can learn from encounters with new anomalies. The model combines an immune inspired algorithm, negative selection, and a self organizing map to produce a system that is able to detect and visualize anomalies, and can learn in a dynamic fashion. The architecture of the model is clearly immune inspired involving mechanisms such as self/non-self discrimination and immune learning.

One of the main advantages of the proposed model is that it generates a visual representation of the self/non-self space, which is generated by feeding synthetic anomalies, produced by a negative selection algorithm, along with normal and abnormal samples to a self-organizing map. This representation allows the visual discrimination of normal, known abnormal, and unknown abnormal regions, helping the understanding of the structure of a complex self/non-self feature space.

The rest of this paper is organized as follows: in Section 2, related work is presented; Section 3 presents a detailed description of the proposed model; Section 4 discusses the experiments that were performed, including the experimental setup, the results and their discussion; finally, Section 5 presents some conclusions and suggestions for future work.

2. BACKGROUND WORK

2.1 Negative Selection Algorithm

The Negative Selection (NS) algorithm [12] is based on the principles of self/non-self discrimination in the immune system. It uses as input a set of strings that represents the normal data (self set) in order to generate detectors in the non-self space. The negative detectors are chosen by matching them to the self strings: if a detector matches a self string, it is discarded, otherwise, it is kept. There exist efficient implementations of the algorithm (for binary strings) that run in linear time with the size of self [10, 12, 19]. Other versions of the algorithm, which work with alternative representations, such as real-valued vectors, have been proposed [14, 16].

There are different variations of the algorithm, which have been applied to solve anomaly detection problems [8, 19], fault detection problems [6, 26], to detect novelties in time series [7, 14], and to function optimization [3].

2.2 Self/non-self discrimination and immune learning

Artificial immune systems (AIS) which model the self/non-self discrimination function of the natural immune system (NIS) are mainly based on the NS algorithm discussed before, nevertheless, new models have been proposed, which are based on danger theory [1, 24].

Other AIS models, which are based on the idiotypic immune network theory [20] and clonal selection principle, emulate the learning capability of the NIS, which is able to learn from its interaction with the environment.

Both categories of AIS models, self/non-self discrimination based and immune learning based, have been extensively and independently investigated since the beginning of AIS research [4, 9]; however, there is not much work on combining these two approaches in one model. The exception are the security system architectures inspired by the NIS such as those proposed by Kephart [21], Dasgupta [5], Williams et al. [17, 27], and Hofmeyr et al.[19], which include different features exhibited by the NIS including self/non-self discrimination and immune learning. However, it is important to highlight that these are high level architectures designed with a specific application in mind: computer security.

2.3 Anomaly visualization

Usually, the anomaly detection problem arises in context where the monitored system is very complex in structure and function (computer systems, computer networks, manufacturing processes, etc). Monitoring this kind of systems is really challenging, even for humans, because of the high number of variables involved and the complex dynamic. Information visualization techniques could help to deal with this complexity, since human perception could detect unexpected features in visual displays and recall related images to detect anomalies [25].

Most of the work done in anomaly visualization has been restricted to the area of computer security. Published work includes: systems for security log visualization [2, 13, 25], network traffic visualization [23], and network structure and activity visualization[11].

2.4 Self-Organizing Maps

A self-organizing map (SOM) is a type of neural network that uses competitive learning [18, 22]. A SOM is able to capture the important features contained in the input space and provides a structural representation that preserves a topological structure. The output neurons of a SOM are organized in a one- or two-dimensional lattice. The weight vectors of these neurons represent prototypes of the input data that can be interpreted as the cluster centroids of samples with similar features.

3. NS-SOM ANOMALY DISCRIMINATION MODEL

The proposed model combines a NS algorithm and a SOM network to produce a 2-dimensional map that represents the feature (self/non-self) space. The trained map is composed of nodes (SOM neurons) that are able to recognize different type of inputs. This allows to classify new samples as normal or abnormal and to visualize them as points in a 2-dimensional representation of the feature (self/non-self) space.

The NS-SOM anomaly discrimination model consist of three phases (see Figure 1) with different goals:

- **Phase 1: self tolerization**. Normal samples are used to train a SOM that is able to discriminate between normal and abnormal samples.
- **Phase 2: primary response (affinity maturation)**. Normal and abnormal samples are presented to the map. If the samples are unlabeled, the map classifies them as normal or abnormal, using labels assigned to each SOM node during the self tolerization training phase. If the samples are labeled, this information is used to update the label of the cells to reflect the class of input they recognize (normal, unknown anomaly, known anomaly).
- **Phase 3: secondary response**. Unlabeled samples are presented to the map. The map can classify the input in different categories including normal and different types of anomalies.

The self tolerization phase uses an NS algorithm to produce artificial anomalies from the normal samples. The idea of using an NS algorithm to produce artificial anomalies instead of non-self detectors was proposed by Gonzalez et al. [14, 15]. In that work, NS-generated artificial anomalies were used to feed a classifier training algorithm in order to produce an anomaly classifier. Here, a similar approach is followed, but a SOM training algorithm is used instead of a classifier training algorithm. The SOM training algorithm produces a network whose node weights represent points in the feature space reflecting the structure of the normal and artificial abnormal samples used for training. Some nodes represent normal samples, and the others represent abnormal samples. The nodes are labeled accordingly with the

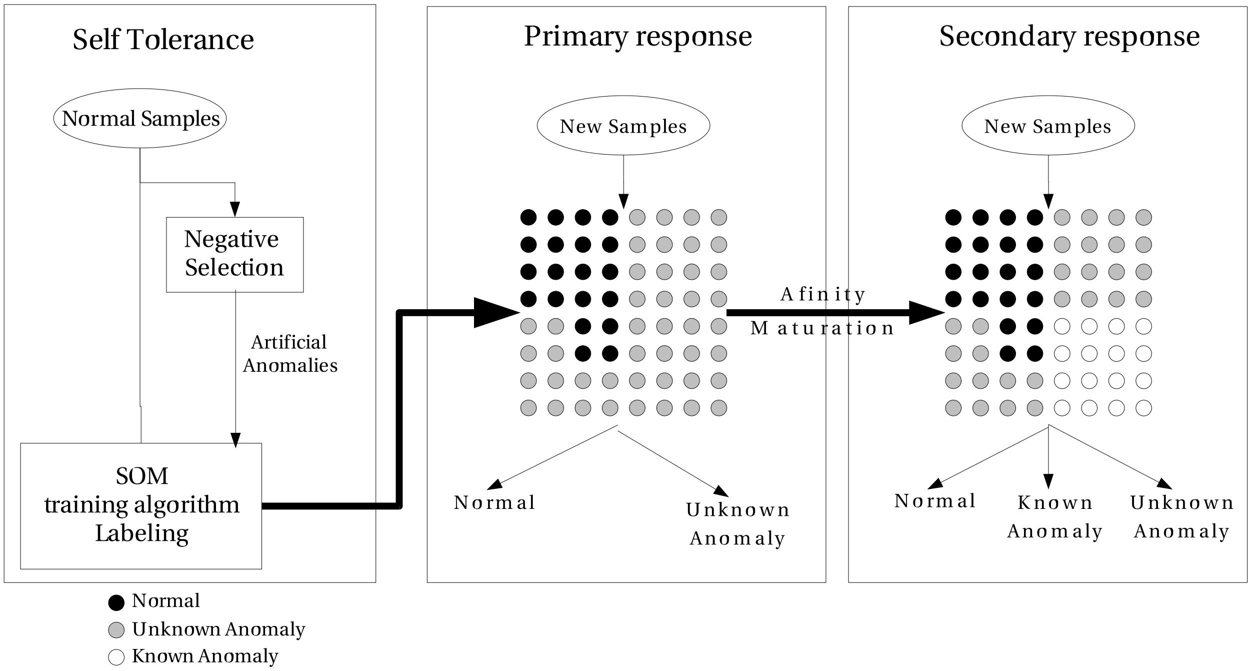


Figure 1: NS-SOM model structure. The model consist of three phases: self tolerization, primary response (affinity maturation), and secondary response. The squared arrange of nodes corresponds to a self-organizing map, where black, gray, and white labels represent normal, unknown anomaly, and known anomaly respectively.

category they represent. Notice that the use of a NS algorithm to generate artificial anomalies is very important since, in many real world problems, only normal samples are available, and training a SOM with only normal samples will produce a map that only reflects the structure of the self space ignoring the non-self space.

In this work, the feature (self/non-self) space corresponds to an n-dimensional real valued space; therefore, it is necessary to use an NS algorithm that could deal with this real valued representation. For this work, the randomized real-valued negative selection (RRNS) algorithm is used [16].

During the second phase, if unlabeled samples are presented, the network can classify them as normal or abnormal; this constitutes the primary response of the system. The classification is done by finding the node that is closest to the input (called the winner node), and classifying the input depending on the label of this node (normal or abnormal). If the input samples are labeled, they could be used to improve the accuracy of the classification (affinity maturation) by changing node labels[1]. Node labels are assigned by finding the closest labeled sample for each node and assigning the sample's label to the corresponding node. This strategy could be generalized to use k-nearest neighbor classification; in this case, the k closest samples are used and the label of the majority is assigned.

In the third phase, secondary response, new unlabeled samples are presented and they are classified accordingly with the label of the winner node. The network is expected to produce a more specific response that could identify the kind of input more precisely as normal or as a specific kind of anomaly.

Notice that the first phase is executed just once, but the second and third phases could be executed as many times as sets of new samples are available. If the samples belong to an anomaly class that is presented for the first time, the system generates a primary response, otherwise it produces a secondary response.

A visual representation of the feature (self/non-self) space could be generated by drawing the 2-dimensional grid corresponding to the network, and assigning each node a different color depending on the category it represents (normal, unknown anomaly, or known anomaly). A new input could be visualized by highlighting the node that gets activated when the sample is presented. An assessment of the type of input presented could be easily done be identifying the color of the highlighted node.

4. EXPERIMENTATION

In order to test the proposed approach with real data, we used two data sets composed of different classes. One of the classes was considered as normal and the remaining classes were considered as abnormal. The data set was divided in training and test subsets. Normal samples from the training subset were used as input for the first phase. Normal and abnormal samples from the training subset were used as labeled samples during the second phase. Normal and ab-

[1] Notice that the weights of the node could be changed, and, probably, this will improve the classification accuracy. However, we chose not to modify the weights to keep the initial version of the model simple. Additional experimentation is required to determine the impact of modifying weights during the second and third phases.

normal samples from the test subset were used as unlabeled samples to test the generalization capability of the model during the second and third phases.

The results are evaluated by calculating confusion matrices, which account the predicted class and the real class for every test sample presented. The purpose of these exploratory experiments is to evaluate the capability of the model to represent the structure of the feature (self/non-self) space and to differentiate diverse kind of anomalies.

4.1 Experimental setup

The data sets used were the Iris data set and the Wisconsin Breast Cancer data set[2]. Both data sets have been widely used to test classification algorithms. In both data sets, the attributes were normalized to fit the $[0, 1]$ real interval. Training and test sets were generated using 10-folding cross validation. Five experiments were performed for each different pair of training and test subsets totaling 50 experiments per data set.

The RRNS algorithm [16] was run using as input the normal data from the training subset to generate a number of detectors (100 for Iris data set, 300 for Breast Cancer data set). The algorithm parameters were: self radius, $r_{self} = 0.1$; minimum accepted transitions, $\eta_{min} = 0.3$; temperature decay rate, $\alpha = 0.9$; neighborhood radius decay rate, $\alpha_{pert} = 0.95$; self covering importance coefficient, $\beta = 1$; and the number of iterations = 600. Two different SOM topologies were used with a rectangular output layer of 8×8 and 16×16 nodes. The SOM network was trained using the SOM-PAK package[3]. The parameters of the algorithm were: random weight initialisation; training rates, $\alpha_1 = 0.05$ and $\alpha_2 = 0.02$; neighborhood radius, $r_1 = 10$ and $r_2 = 3$; number of iterations for first and second training, 1000 and 10000 respectively; and bubble neighborhood.

4.2 Iris data set experimental results

The Iris data set contains 3 classes of 50 4-dimensional instances each, where each class refers to a type of iris plant. One class is linearly separable from the other two, the latter are not linearly separable from each other. The data set is adapted to be used for anomaly detection testing by considering one class as normal class and the others as abnormal classes; therefore, experiments were made in such a way that each class was considered as normal in different independent executions. A total of 150 experiments were performed (50 experiments per class).

Figure 2 shows a visual representation of the network (SOM) evolving through the different phases of the model. The black zone represents the self (normal) region, the gray and white zones represent nodes that detect unknown and known anomalies respectively. In this case, class 1 was taken as normal. The first phase, self tolerization, produces an initial configuration of the network that is able to discriminate between normal and abnormal samples, this is shown in Figure 2(a), where the feature space is divided in a normal (self) region and unknown anomalies (non-self) region. At this point, the network performs an affinity maturation process (second phase), so that some nodes become more specific. Figures 2(b), 2(c) and 2(d) show the results of the primary response phase (second phase) when new labeled samples are presented, including class 2 and 3 data. Notice that the white zones correspond to nodes that represent known anomalies, which make the network able to remember previous encounters discriminating between known (white zones) and unknown (gray zones) anomalies.

Table 1: Average detection rates (ADR) and average false alarm rates (AFAR) over 50 experiments (per class) for the primary response phase. Each column represents the results produced when a specific class was used as normal. The standard deviation is reported between parentheses.

	Normal Class		
	1	**2**	**3**
ADR	92.0%(10.8%)	51.6% (5.8%)	47.4% (20.1%)
AFAR	0.0% (0.0%)	0.0% (0.0%)	0.0% (0.0%)

Table 2: Average detection rates (ADR) and average false alarm rates (AFAR) over 50 experiments (per class) in secondary response. Each column represents the results produced when a specific class was used as normal. The standard deviation is reported between parentheses.

	Normal Class		
	1	**2**	**3**
ADR	92.2% (10.9%)	81.4% (15.5%)	80.0% (15.1%)
AFAR	0.0% (0.0%)	2.4% (7.7%)	0.8% (3.9%)

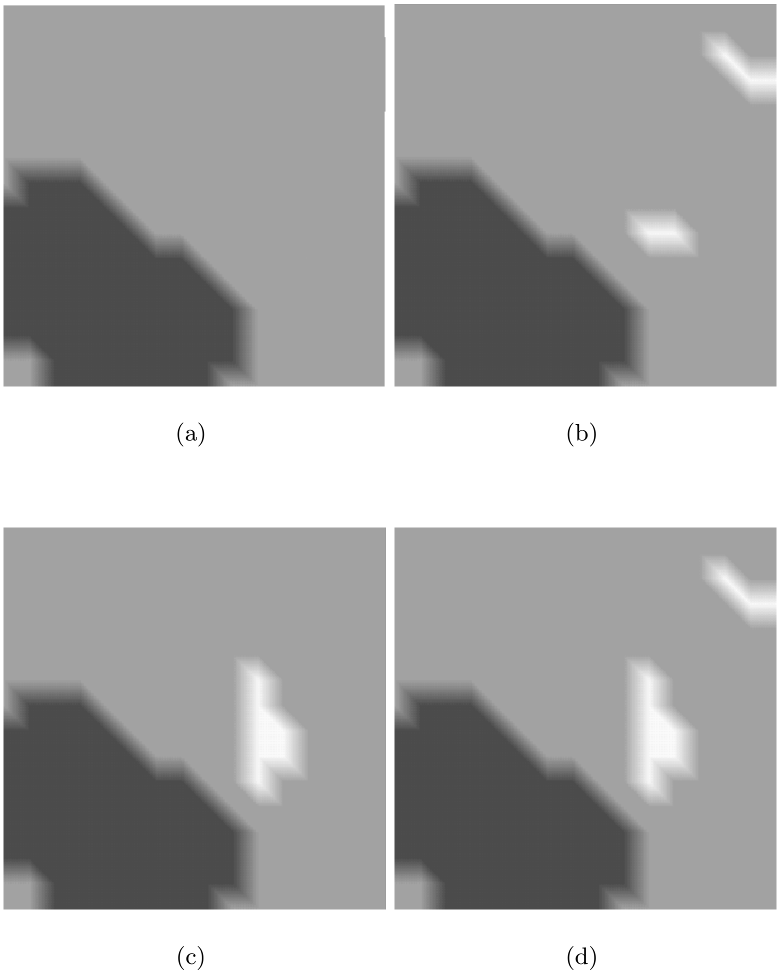
Table 1 summarizes the results of the primary response phase for different normal classes. Each value corresponds to the average over 50 experiments, where test samples, different to the ones used for training, are presented to the NS-SOM model. The detection rate is the percentage of well classified anomalies and the false alarm rate is the percentage of the normal samples erroneously classified. The results are consistent with the structure of the data set, i.e. class 1 is linearly separable from the other two classes, so it is easy to discriminate, but classes 2 and 3 overlap.

Table 2 summarizes the secondary response results. The detection rates were improved, showing the effect of the affinity maturation process. However, the false alarm rate increases for classes 2 and 3. It has to do with the fact that classes 2 and 3 overlap. The affinity maturation improves the detection rate by reducing the self region area, but this increases the false alarm rate, since normal samples could lie outside the identified self region.

Table 3 shows a more detailed view of the results of the secondary response phase. The rows represent the real class of the samples input to the model. The columns represent the class predicted by the model grouped by the class used as normal. U.A. means unknown anomaly. As before, the table entries correspond to the average of 50 different experiments.

[2] UCI repository of machine learning databases and domain theories `ftp://ftp.ics.uci.edu/pub/machine-learning-databases`

[3] Available at the Laboratory of Computer and Information Science of the Helsinki University of Technology home page: `http://www.cis.hut.fi/`


(a) (b)

(c) (d)

Figure 2: Network evolution through the different phases of the model. Black, white and gray zones represent normal class, known anomalies and unknown anomalies respectively. After the first phase, self-tolerization, a gray region appears representing unknown anomalies (a). At the end of the second phase, primary response, white regions appear in the map representing known anomalies. (b), (c) and (d) show the state of the network after new samples from class 2 (b), class 3 (c), and classes 2 and 3 (c) were presented.

Table 3: Confusion matrices for the secondary response phase for the Iris data set. The values correspond to the mean of 50 experiments. The rows represent the real class of the input samples. The columns represent the class predicted by the model. U.A. means unknown anomaly. The data subsets used for test have 5 samples per class, so the diagonal elements of an ideal confusion matrix must be equal to 5.

	Normal Class 1				**Normal Class 2**				**Normal Class 3**			
	Predicted Class											
Real class	**1**	**2**	**3**	**U. A.**	**1**	**2**	**3**	**U. A.**	**1**	**2**	**3**	**U. A.**
1	5.0	0.0	0.0	0.0	3.6	0.0	0.0	1.4	4.0	0.0	0.6	0.4
2	0.0	4.4	0.1	0.5	0.0	4.9	0.1	0.0	0.0	3.7	1.3	0.0
3	1.2	1.1	0.9	1.8	0.1	2.9	2.0	0.0	0.0	0.1	4.9	0.0

Table 4: Average detection rates (ADR) and average false alarm rates (AFAR) over 50 experiments in primary and secondary responses for the Breast Cancer Data Set. The standard deviation is reported between parentheses.

	Primary Response	Secondary Response
ADR	93.8% (5,7%)	95.3% (4.9%)
AFAR	3.7% (2.4%)	4.1% (2.6%)

Table 5: Confusion matrices of the primary and secondary response phases for the Breast Cancer data set. The values correspond to the mean of 50 experiments. The standard deviation is reported between parentheses. The data subsets used for test have between 44 and 45 normal samples and between 23 and 24 abnormal samples.

	Primary Response		Secondary Response	
Real Class	**Predicted Class**			
	Normal	**Abnor.**	**Normal**	**Abnor.**
Normal	42.7 (1.3)	1.7 (1.1)	42.6 (1.4)	1.8 (1.1)
Abnor.	1.5 (1.3)	22.4 (2.0)	1.1 (1.2)	22.8 (1.2)

4.3 Breast cancer data set experimental results

Each record in this data set is conformed by nine numerical attributes and the label (benign or malign). The data is composed by 699 records, but 16 of them have missing values (we did not use these records).

The results of the primary and secondary response phases are shown in Table 4 and Table 5. Interestingly, the results of the primary response phase are good, despite the model were trained only with normal samples. This indicates that the self subspace has a simple structure and it does not overlap with the non-self subspace. The results of the secondary response phase are slightly better than the results of the primary response phase. In this case, the affinity maturation process did not have much effect on the accuracy of the model.

Figure 3 shows a typical 2-dimensional representation of the self/non-self space for this data set generated by the NS-SOM model.

4.4 Discussion

The overall performance of the NS-SOM model over these two data set is good. It produces an acceptable self/non-self discrimination, which is improved by the affinity maturation process carried on during the primary response. The secondary response is more specific and it is able to discriminate the different types of anomalies (in the case of the Iris data set).

These results are not as good as other results reported for the same data sets, but it is important to highlight that the particular experimental setup used in this set of experiments is different to the typical experimental setup used to test conventional classifications algorithms. In this case, the data set is gradually shown to the NS-SOM model: first, the normal class is presented, allowing the model to create an internal representation of the self/non-self space; next, the remaining (abnormal) classes are presented, allowing the model to update the labels, not the weights, of the network nodes.

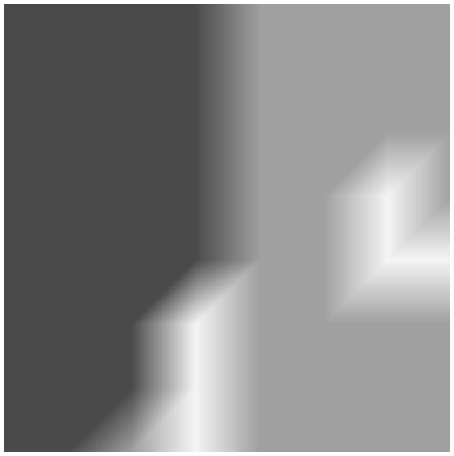


Figure 3: Visual representation of a typical SOM map generated by NS-SOM for the Breast Cancer data set. Black, white, and gray zones represent normal class, known anomalies, and unknown anomalies respectively.

In the visual representation of the self/non-self space produced by the NS-SOM model, it is possible to distinguish clearly defined areas of normal, known abnormal and unknown abnormal regions of the feature space. This helps to usnderstand the structure of a possibly complex self/non-self space, and may give insights on the subjacent system function or problem structure.

5. CONCLUSIONS

An AIS model that exhibits self/non-self discrimination and immune learning capabilities is presented. The model combines an NS algorithm and a SOM training algorithm to produce a network that can discriminate normal samples from abnormal samples and can learn from its encounters with antigens to improve the specificity of its response.

One remarkable characteristic of the model is its ability to generate a 2-dimensional visual representation of the feature space. This representation facilitates the understanding of the structure of the self/non-self space by producing a visual discrimination of the normal, known abnormal and unknown abnormal regions. This feature could be useful for building interactive visualization tools, such as a graphical anomaly monitoring system.

The NS-SOM model could give some insight on the nature of the problem of building a practical model that combines self/non-self discrimination and immune learning. Also, it might serve as a basis to build a more complex model that could be applied to solve real problems.

Experiments using two well known classification data sets were carried on. Based on the results it is possible to make the following remarks:

- The model could produce a network that captures the structure of the normal samples used for training. This could be observed in the 2D topological map associated with the network, which assigns a set of neighbor nodes to the normal and abnormal subspaces respectively.
- During the second phase, primary response, the model was able to perform an acceptable discrimination between normal and abnormal samples when trained only with normal samples.
- In the third phase, secondary response, this discrimination was improved by the affinity maturation process carried on during the second phase of the model, mainly for the Iris data set. Also, the model was able to distinguish the two different kinds of anomalies with a good accuracy.

The results are encouraging and suggest that it is worthy to pursue further research to improve the model. Some of the possible research paths to be explored are:

- Performing a more systematic experimentation with other data sets that could assess the real strength of the model and its sensitivity to the parameters.
- Allowing a better adaptation of the model during the second and third phases, by performing a better affinity maturation. This could be accomplished by applying the SOM training algorithm (or a Learning Vector Quantization algorithm) to adjust the vectors representing the nodes to get and improved matching of the input samples.
- Using an AIS immune network model instead of a SOM network.

6. ACKNOWLEDGMENTS

The authors would like to thank Luis F. Nino, for his valuable corrections and suggestions, and the members of LISI (Intelligent Systems Research Lab at National University of Colombia) for their encouragment and support.

7. REFERENCES

[1] U. Aickelin, P. Bentley, S. Cayzer, J. Kim, and J. McLeod.Danger theory: The link between AIS and IDS?In J. Timmis, P. Bentley, and E. Hart, editors, *Proceedings of the 2nd International Conference on Artificial Immune Systems*, volume 2787 of *Lecture Notes in Computer Science*, pages 147–155. Springer-Verlag, September 2003.

[2] S. Axelsson.Visualising intrusions: Watching the webserver,.In *proceedings of the 19th IFIP International Information Security Conference (SEC2004)*, Tolouse, France, Aug 2004.

[3] C. A. Coello Coello and N. Cruz Cortés.A parallel implementation of the artificial immune system to handle constraints in genetic algorithms: preliminary results.In D. B. Fogel, M. A. El-Sharkawi, X. Yao, G. Greenwood, H. Iba, P. Marrow, and M. Shackleton, editors, *Proceedings of the 2002 Congress on Evolutionary Computation CEC2002*, pages 819–824, USA, 2002.

[4] D. Dasgupta.*Artificial immune systems and their applications.*Springer-Verlag, New York, 1999.

[5] D. Dasgupta.Immunity-based intrusion detection system: a general framework.In *Proceedings of the 22nd national information systems security conference (NISSC)*, pages 147–160, Oct. 1999.

[6] D. Dasgupta and S. Forrest.Tool breakage detection in milling operations using a negative-selection algorithm.Technical Report CS95-5, Department of Computer Science, University of New Mexico, 1995.

[7] D. Dasgupta and S. Forrest.Novelty detection in time series data using ideas from immunology.In J. F. C. Harris, editor, *Proceedings of the 5th International Conference on Intelligent Systems*, pages 82–87, Cary, NC, June 1996. ISCA.

[8] D. Dasgupta and S. Forrest.An anomaly detection algorithm inspired by the immune system.In D. Dasgupta, editor, *Artificial immune systems and their applications,*, pages 262–277. Springer-Verlag, New York, 1999.

[9] L. N. de Castro and J. Timmis.*Artificial Immune Systems: A New Computational Approach.*Springer-Verlag, London, UK, 2002.

[10] P. D'haeseleer, S. Forrest, and P. Helman.An immunological approach to change detection: algorithms, analysis and implications.In J. McHugh and G. Dinolt, editors, *Proceedings of the 1996 IEEE Symposium on Computer Security and Privacy*, pages 110–119, USA, 1996. IEEE Press.

[11] R. F. Erbacher.Glyph-based generic network visualization.In *Proceedings of the SPIE '2002 Conference on Visualization and Data Analysis*, pages 228–237, San Jose, CA, January 2002.

[12] S. Forrest, A. Perelson, L. Allen, and R. Cherukuri.Self-nonself discrimination in a computer.In *Proceedings IEEE Symposium on Research in Security and Privacy*, pages 202–212, Los Alamitos, CA, 1994. IEEE Computer Society Press.

[13] L. Girardin and D. Brodbeck.A visual approach for monitoring logs.*Proceedings of the Twelth Systems Administration Conference (LISA XII) (USENIX Association: Berkeley, CA)*, page 299, 1998.

[14] F. González and D. Dasgupta.Anomaly detection using real-valued negative selection.*Genetic Programming and Evolvable Machines*, 2003.to be published.

[15] F. González, D. Dasgupta, and R. Kozma.Combining negative selection and classification techniques for anomaly detection.In D. B. Fogel, M. A. El-Sharkawi, X. Yao, G. Greenwood, H. Iba, P. Marrow, and M. Shackleton, editors, *Proceedings of the 2002 Congress on Evolutionary Computation CEC2002*, pages 705–710, USA, May 2002. IEEE Press.

[16] F. González, D. Dasgupta, and F. Niño.A randomized real-valued negative selection algorithm.In J. Timmis, P. Bentley, and E. Hart, editors, *Proceedings of the 2nd International Conference on Artificial Immune Systems*, volume 2787 of *Lecture Notes in Computer Science*, pages 261–272. Springer, September 2003.

[17] P. Harmer, G. Williams, P.D.and Gnusch, and G. Lamont.An Artificial Immune System Architecture for Computer Security Applications.*IEEE*

Transactions on Evolutionary Computation, 6(3):252–280, June 2002.

[18] S. Haykin.*Neural networks : a comprehensive foundation*.Macmillan, New York, 1994.

[19] S. Hofmeyr and S. Forrest.Architecture for an artificial immune system.*Evolutionary Computation*, 8(4):443–473, 2000.

[20] N. K. Jerne.Towards a network theory of the immune system.*Ann. Immunol. (Inst. Pasteur)*, 125C:373–389, 1974.

[21] J. O. Kephart.A biologically inspired immune system for computers.In R. A. Brooks and P. Maes, editors, *Proceedings of the 4th International Workshop on the Synthesis and Simulation of Living Systems ArtificialLifeIV*, pages 130–139, Cambridge, MA, USA, July 1994. MIT Press.

[22] T. Kohonen.*Self-Organizing Maps*, volume 30 of *Springer Series in Information Sciences*.Springer, Berlin, Heidelberg, 1995.(Second Extended Edition 1997).

[23] I.-V. Onut, B. Zhu, and A. A. Ghorbani.
A novel visualization technique for network anomaly detection.In *Proceedings of the Second Annual Conference on Privacy, Security and Trust (PST'04)*, New Brunswick, Canada, oct 2004.

[24] A. Secker, A. Freitas, and J. Timmis.A danger theory approach to web mining.In J. Timmis, P. Bentley, and E. Hart, editors, *Proceedings of the 2nd International Conference on Artificial Immune Systems*, volume 2787 of *Lecture Notes in Computer Science*, pages 156–167. Springer-Verlag, September 2003.

[25] S. T. Teoh, T. J. Jankun-Kelly, K.-L. Ma, and S. F. Wu.Visual data analysis for detecting flaws and intruders in computer network systems.*IEEE Computer Graphics and Applications*, 24(5), Sep/Oct 2004.

[26] A. Tyrrell.Computer know thy self! : a biological way to look at fault tolerance.In *Proceedings of the 2nd Euromicro/Ieee workshop on Dependable Computing Systems*, pages 129–135, Milan, 1999.

[27] P. D. Williams, K. P. Anchor, J. L. Bebo, G. H. Gunsch, and G. D. Lamont.CDIS: Towards a computer immune system for detecting network intrusions.*Lecture Notes in Computer Science*, 2212:117–133, 2001.