ESANN'2007 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 25-27 April 2007, d-side publi., ISBN 2-930307-07-2.

Classification of computer intrusions using functional networks. A comparative study

A. Alonso-Betanzos, N. Sánchez-Maroño, F.M. Carballal-Fortes,
J. Suárez-Romero and B. Pérez-Sánchez [*]

Department of Computer Science - University of A Coruña
Campus de Elviña s/n - 15071 A Coruña. Spain.

Abstract. Intrusion detection is a problem that has attracted a great deal of attention from computer scientists recently, due to the exponential increase in computer attacks in recent years. DARPA KDD Cup 99 is a standard dataset for classifying computer attacks, to which several machine learning techniques have been applied. In this paper, we describe the results obtained using functional networks – a paradigm that extends feedforward neural networks – and compare these to the results obtained for other techniques applied to the same dataset. Of particular interest is the capacity for generalization of the approach used.

1 Introduction

A computer intrusion is a set of actions that violate the security of a system. Such a situation must be detected and corrected in order to guarantee the integrity, confidentiality and/or availability of computing resources. Intrusion detection systems (IDS) have been designed that complement other security measures based on attack prevention (firewalls, antivirus, etc.). The aim of an IDS is to inform the system administrator of any suspicious activities and to recommend specific actions to prevent or stop the intrusion (for example, close network ports, kill doubtful processes, etc.). In order to be able to implement these actions, the IDS must, among other tasks, analyze network traffic data in order to determine whether there is evidence of an attack, or whether the data are anomalous with respect to normal traffic. Ideally, the system should be sufficiently generalized to be able to detect any type of attack yet maintain a low false positive rate. The false positive measure is of great importance in determining the quality of an IDS system [11].

There are two basic intrusion detection systems: misuse detection and anomaly detection ([7] [1]). *Misuse detection* systems attempt to match computer activities with previously known attacks in their database. An important drawback of this type of system is that it can only detect known attacks, so attacks that are not stored or variants of stored attacks will not be detected. *Anomaly detection* systems learn the normal activity of the system and attempt to detect any computer activity that deviates from normal patterns. The problem here

[*]This work has been funded by projects TIN2006-02402 of the Ministerio de Educación y Ciencia, Spain (partially supported by the European Union ERDF), PGIDIT03TIC0501PR and PGIDT05TIC10502PR of the Xunta de Galicia. The authors acknowledge the Supercomputer Center of Galicia (CESGA) for the use of its high–performance computer servers.

is that too broad or too narrow training can result, respectively, in high false negative or false positive rates. In our approach, we implemented an anomaly detection method with a view to raising the efficiency of the systems already reported in the literature.

There are two main ways to retrieve data for constructing an IDS. The first one consists of implementing a network simulation and recovering relevant data. This approach has drawbacks in regard to simulating a real network with hundreds of computers, comparing the results obtained with other authors´ results, and emulating real traffic networks. An alternative is to employ standard datasets available for this area. For this work we used the KDD Cup 99 dataset [5].

Several approaches to the IDS problem have been reported in the literature, such as artificial neural networks [7]; kernel-based methods, Support Vector Machines (SVM), and variants based on each [3] [4] [8], etc. Functional networks are a generalization of neural networks that have also been successfully applied to a range of different problems [10]. In this work, this is the paradigm applied to the KDD Cup 99 dataset problem. We describe our results and compare them to the results obtained by other authors and by the KDD Cup 99 competition winner.

2 Material and Methods

2.1 The KDD Cup 99 dataset

The KDD Cup 99 dataset, which derives from the DARPA dataset [6], was used for the 1999 KDD (Knowledge Discovery and Data Mining Tools Conference) Cup Competition [5]. Each record representing a TCP/IP connection is composed of 41 features that are both qualitative and quantitative in nature [9]. The dataset used in our study is a smaller subset of the original training set which was unwieldy to work with as it contains almost 5 million input patterns. For the sake of comparison with other authors' results [3], two subsets were extracted from this larger training set: a set of 46093 patterns for training and a set of 447927 patterns for validation. These subsets were selected in such a way that the distribution of attacks were the same as in the original DARPA dataset. Although the model training set was but a small part of the available data, we observed no significant degradation in generalization performance which would indicate inadequate dataset size. For the test set we used the original KDD Cup 99 dataset containing 331029 patterns. Around 20% of the three sets were normal patterns (no attacks). As for attack patterns, four categories were identified:

- Denial of Service (DoS) attacks, which prevent a computer from complying with legitimate requests by consuming its resources.
- Probe attacks, which are scanning and polling activities that gather information on vulnerabilities for future attacks.

Type	% Training and Validation Sets	% Test Set
Normal	19.69	19.48
DoS	79.24	73.90
Probe	0.83	1.34
R2L	0.23	5.21
U2R	0.01	0.07

Table 1: Percentage distribution of normal activities and different kinds of attacks in the KDD Cup 99 training, validation and test datasets.

- Remote-to-local (R2L) attacks, which are local non-authorized access attempts from a remote machine.
- User-to-root (U2R) attacks, which have the goal of obtaining illegal or non-authorized super-user or root privileges.

The training, validation and test set percentages for normal activities and for the four attack types are shown in Table 1.

2.2 Preprocessing

From the KDD Cup 99 intrusion detection dataset, 41 features were derived to summarize each connection information. In order to train an architecture, several data enumeration and normalization operations were necessary. As a first approach, symbolic variables in the dataset were enumerated and all variables were normalized. Thus, each instance of a symbolic feature was first mapped to sequential integer values. The data was then normalized to ensure that no input vector component has an overwhelming influence on the training result. Standard [0..1] normalization was used for this research. However, some numerical features of the connection feature vector (such as connection duration, total bytes set to destination/source host) had dynamic range values. After a detailed analysis normalization was performed for intervals.

2.3 Functional Networks

Functional networks are a generalization of neural networks that combine both knowledge about the structure of the problem and data: the former determines the architecture of the network, and the latter estimates the unknown functional neurons [10]. There are important differences between neural and functional networks, however. One of the most significant differences is that, in functional networks, weights are incorporated into the neural functions (f, g, h and p in Figure 1). These neural functions are unknown functions (from a given family of functions, e.g. polynomial or Fourier) to be estimated during the learning process. For example, the neural function f might be approximated by:

$$f(x_1) = c_0 + \sum_{i=1}^{m_i} c_i x_1^i \qquad \text{or} \qquad f(x_1) = c_0 + \sum_{i=1}^{m_i} (c_{2i-1}) sin(ix) + c_{2i} cos(ix) \quad (1)$$

and the parameters to be learned will be the coefficients c_0 and c_i. As each neural function is learnt, a different function is obtained for each neuron. The architecture of the network, moreover, is derived from the known properties of the data, normally, by applying functional equations [10]. In the KDD Cup 99 dataset, there was not enough information to obtain this architecture. Hence, as a first approximation a simple but powerful model was selected, namely the associativity model depicted in Figure 1.

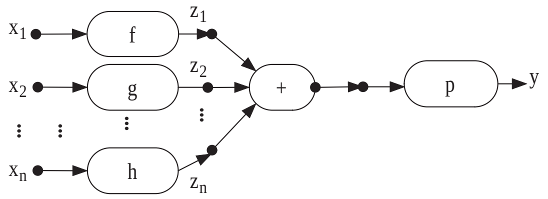

Fig. 1: The generalized associativity functional network.

3 Results and Discussion

There are a number of function families that can be used for the neural functions in a functional network model. For our research we considered polynomial, Fourier and exponential functions. After several trials it was found that the best approach was one in which five functional networks were developed (in which $m_i = 3$ in equation 1), one for each of the classes in Table 1. The overall output of the system corresponded to the functional network with the highest output.

Tables 2 and 3, respectively, show the results obtained for the validation and test sets. For the sake of comparison, the metrics used by other authors [3], were used. The first three columns in each table correspond to binary classification results (obtained by grouping the four attack types in one class and the normal patterns in another class). As for the columns Error shows the overall percentage error rate for the five categories (normal patterns plus the four attack types), Det shows the overall percentage of attacks detected, and FP shows the false positive rate, defined as the proportion of normal patterns erroneously classified as attacks. The last four columns in each table show correct detection (Det) classification percentages for each of the four attack types. The first three lines in each table show results for our proposed method, while the remaining lines show results obtained by other authors (see [3]), specifically, for different SVM models, ANOVA (ANOVA ens.), and linear perceptrons (Pocket 2cl. and Pocket mcl.). Finally, the results obtained for the KDD Cup 99 competition winner are reproduced in the fourth line in Table 3.

As can be seen from Tables 2 and 3, the functional networks had lower error rates and lower false positive rates. In the case of the validation set, the results were poorer or similar to those reported by other authors. However, for the test

Method	**Total**			**DoS**	**Probe**	**R2L**	**U2R**
	Error	*Det*	*FP*	*Det*	*Det*	*Det*	*Det*
5FNs_poly	0.27	99.83	0.51	99.86	94.94	92.36	61.54
5FNs_fourier	0.28	99.83	0.54	99.86	95.33	92.98	73.08
5FNs_exp	0.28	99.83	0.55	99.86	95.37	92.90	73.08
SVM Linear	0.17	99.89	0.39	99.98	98.21	79.11	10.00
SVM 2poly	0.08	99.93	0.11	99.99	98.40	88.27	2.00
SVM 3poly	0.08	99.94	0.15	99.99	98.43	91.35	6.00
SVM RBF	0.07	99.94	0.11	99.99	99.06	90.02	20.00
ANOVA ens.	1.07	98.67	0.02	99.18	77.65	0.93	2.00
Pocket 2cl.	1.31	98.40	0.06	99.33	35.71	3.81	2.00
Pocket mcl.	0.20	99.89	0.56	99.96	98.54	83.95	20.00

Table 2: An overview of the results (in %) obtained for the validation dataset.

Method	**Total**			**DoS**	**Probe**	**R2L**	**U2R**
	Error	*Det*	*FP*	*Det*	*Det*	*Det*	*Det*
5FNs_poly	6.48	92.45	0.86	96.77	85.96	29.43	8.33
5FNs_fourier	6.69	92.72	0.75	96.86	85.74	23.75	10.97
5FNs_exp	6.70	92.75	0.75	96.85	85.60	23.77	13.60
KDD Winner	6.70	91.80	0.55	97.69	87.73	10.26	26.32
SVM Linear	6.89	91.83	1.62	97.38	81.76	16.55	21.93
SVM 2poly	6.95	91.79	1.74	97.41	86.44	14.74	1.75
SVM 3poly	7.10	91.67	1.94	97.62	88.45	9.35	2.63
SVM RBF	6.86	91.83	1.43	97.30	79.26	18.29	25.88
ANOVA ens.	6.88	91.67	0.90	97.64	87.52	8.51	53.94
Pocket 2cl.	6.90	91.80	1.52	97.40	85.84	14.77	29.82
Pocket mcl.	6.93	91.86	1.96	97.65	86.79	11.45	54.38

Table 3: An overview of the results (expressed in %) obtained for the test dataset.

set, the functional networks obtained a lower error rate – lower even than the KDD Cup 99 competition winner error rate. This interesting result highlights the generalization capacity of our system – a fact which is particularly relevant to our problem, given that the test and validation datasets are quite different (see Table 1); leaving aside distribution differences, the test dataset contains new attacks not included in the training and validation sets. Correct classification of R2L and U2R attacks is very difficult, because the model underlying both is very different from that of the other two attack types, as discussed in [12]. This is evident in the results obtained, which show a dramatic drop in quality between the validation and test datasets. Our model attempts to overcome this problem by using five different functional networks, one for each class in the dataset. Consequently, our method exhibits better generalization behavior than the other methods.

4 Conclusions

The results obtained by our new approach to the problem of intrusion detection, based on functional networks, revealed a good capacity for generalization. Generalization is a problem that is difficult to handle in the KDD intrusion detection dataset, due to differences between validation and test datasets. Comparing our results with other approaches – including with the KDD Cup 99 competition winner results– we obtained a better attack detection rate and a lower error rate for the test set. However, our false positive rates were higher than those for the KDD Cup 99 competition winner, although much better than those for the other methods analyzed.

In future research, we plan to consider more complex functional networks models; for example, such as models that take into account relationships between features. We also plan to construct a mixture of experts model, in which the most suitable model would be employed in each classification problem in order to improve the overall performance of the system.

References

[1] A. Chebrolu, A. Abraham, and J. Thomas, Feature Deduction and Ensemble Design of Intrusion Detection Systems. *Computers & Security*, 24:4, 295-307, 2005.

[2] O. Depren, M. Topallar, E. Anarim and M. Kemal, An Intelligent Intrusion Detection System (IDS) for Anomaly and Misuse Detection in Computer Networks, *Expert Systems with Applications*, 29:713-722, 2005.

[3] M. Fugate and J.R. Gattiker, Computer Intrusion Detection with Classification and Anomaly Detection, using SVMs, *International Journal of Pattern Recognition and Artificial Intelligence*, 17: 441-458, 2003.

[4] M. Fugate and J.R. Gatikker, Anomaly Detection Enhanced Classification in Computer Intrusion Detection. Applications of Support Vector Machines, *Int. Conf. Pattern Recognition and Machine Learning*, pp. 186-197, 2002.

[5] KDD Cup 1999 Data. <http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html> [Last access: 20th october 2005].

[6] R. P. Lipmann, D. Fried, I. Graf, J. Haines, K. Kendall, D. McClung, D. Weber, S. Webster, D. Wyschogrod, R. Cunningham, and M. Zissman, Evaluating intrusion detection systems. In the 1998 DARPA off-line intrusion detection evaluation. *Proc. DARPA Information Survivability Conf. & Exposition*, 2000.

[7] S. Mukkamala, A. Sung, and A. Abraham, Intrusion Detection using an ensemble of intelligent paradigms. *Journal of Network and Computer Applications*, 28:167-182, 2005.

[8] S. Mukkamala, A. Sung, Feature Ranking and Selection for Intrusion Detection Systems Using Support Vector Machines. In *Digital Forensic Research Workshop*, 2002.

[9] S. Stolfo, W. Fan, W. Lee, A. Prodomidis and R.K. Chan, Cost-based modeling for fraud and intrusion detection: Results from the JAM project. In *DARPA Information Survivability Conf.*, 2000.

[10] E. Castillo and A. Cobo and J.M. Gutiérrez and E. Pruneda. *Functional Networks with Applications*, Kluwer Academic Publishers, 1998.

[11] S. Axelsson, The base-rate fallacy and its implications for the difficulty of intrusion detection. *Proceedings of the 6th ACM Conference on computer and Communications security* (CCS 1999), pages 1–7, ACM press, 1999.

[12] M. Sabhnani, Why machine learning algorithms fall in misuse detection on KDD Intrusion detection data set. *Journal of Intelligent Data Analysis*, 8:403-445, 2004.