Effectiveness evaluation of data mining based IDS

Agustín Orfila, Javier Carbó, and Arturo Ribagorda

Carlos III University of Madrid,
Computer Science Department,
Leganés 28911, Spain,
{adiaz,jcarbo,arturo}@inf.uc3m.es

Abstract. Data mining has been widely applied to the problem of Intrusion Detection in computer networks. However, the misconception of the underlying problem has led to out of context results. This paper shows that factors such as the probability of intrusion and the costs of responding to detected intrusions must be taken into account in order to compare the effectiveness of machine learning algorithms over the intrusion detection domain. Furthermore, we show the advantages of combining different detection techniques. Results regarding the well known 1999 KDD dataset are shown.

1 Introduction

According to ISO/IEC TR 15947 [8] intrusion detection is the process of identifying that an intrusion has been attempted, is occurring or has occurred. Thus, Intrusion Detection Systems (IDS) are technical systems that are used to identify and respond to intrusions in IT systems. Consequently, IDS attempt to identify actions that does not conform to security policy.

IDS analysis of the data sources can be done through different methods such as expert systems, statistical techniques, signature analysis, neural networks, artificial immune systems or data mining among others. Data mining is defined as the process of discovering patterns in data automatically. This technique deals well with big amounts of information what makes it appropriate for the intrusion detection task. In fact, the appliance of data mining to the intrusion detection field is an active research topic [15, 23, 6]. After the process of extracting the interesting characteristics from data sources, either supervised or unsupervised learning can be applied over the processed data. Supervised algorithms need a training set in order to build the model that will be used in operating conditions. At the training phase, the IDS can model either the normal behaviour of the system, the abnormal or both. The main advantages of IDS based on supervised learning are their ability to detect known attacks and minor variants of them. Weak points deal with the necessity of building proper training datasets and with the time consuming phase for building the models. On the other hand, unsupervised learning does not require a training dataset.

Evaluation of IDS effectiveness did not become an active topic until 1998. MIT Lincoln Laboratories (MIT/LL) led an ambitious evaluation of IDS [12, 13]. A military network was simulated in order to test different proposals. These works and subsequent criticism [16, 14] set down the main difficulties that research community had to face in order to evaluate IDS effectiveness [17]. A processed version of the data generated by MIT/LL was used to evaluate several machine learning algorithms in the 1999 KDD intrusion detection contest [4]. This dataset has been used extensively after the contest although its limitations (derives from a controversial dataset and the probability of intrusion that it shows is far away from what is expectable in a real scenario).

In order to measure IDS effectiveness the *receiver operating characteristic* (ROC) has been widely used. ROC was introduced in signal theory to characterize the trade-off between hit rate and false alarm rate over a noisy channel. In data mining field, it has been used to measure machine learning algorithms performance when the class distribution is not balanced (i.e. accuracy is not a good measure). The main problem of ROC analysis is that it does not consider the costs of misclassification (wrong detections). Normally the cost of a false negative (failing to detect an intrusion) is much higher than the cost of a false positive (stating an event is hostile when it is not) so a different representation of effectiveness that takes this situation into account is necessary. In fact, if the ROC curves of two classifiers (detectors) cross, it could not be stated that one outperforms the other under any circumstances. Because of this, Drummond [3] proposed an alternative representation to the ROC in order to compare classifiers under different class distributions and error costs.

Ulvila works [5, 24, 25] showed that IDS effectiveness depends not only on the hit rate and false alarm rate but also on the hostility of the operating environment and on the cost of detection errors. Orfila [19] showed the need of including the response cost in order to evaluate IDS effectiveness properly.

The remainder of this paper is organized as follows. In section 2 we expose a methodology for comparing IDS effectiveness from a decision analysis perspective. Then in section 3 we propose a simple way to combine the detection capabilities of several machine learning techniques. Section 4 overviews 1999 KDD dataset that is used in section 5 for the experimental work. This experimental work compares different machine learning algorithms with the combined model proposed in section 3 by means of the methodology described in section 2. Finally, this paper ends up with the main conclusions.

2 Decision Model Analysis

Decision theory has been successfully applied in areas such as Psychology [22], Economy [21] or Meteorology [9]. In the computer security field, it has been used to face the intrusion detection task, in order to provide a way to compare different IDS and to determine the best operating point of an IDS under different operating conditions [5].

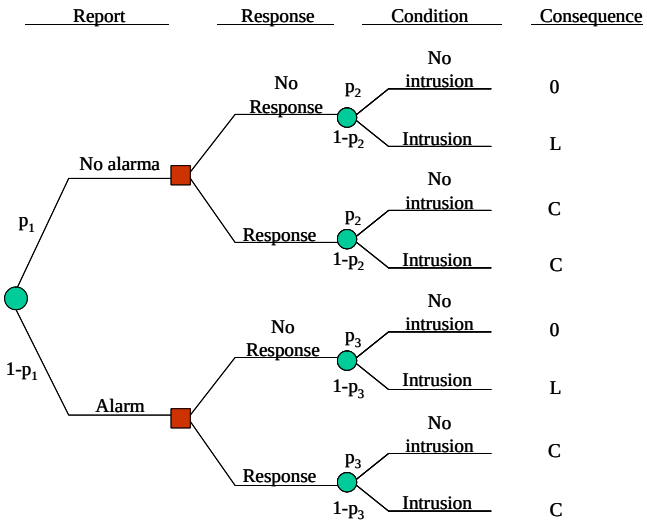

Fig. 1. Decision tree of the detector's expected cost that considers the response cost

In this section we propose a method to measure IDS effectiveness that considers both damage (produced by a successful intrusion) and response cost (the one incurred by taking actions in order to avoid an intrusion [19]). The comparison is made from an utility perspective, that means the best IDS is the one that better helps on minimizing expenses when defending a system.

The system we want to protect can be in two possible states: an intrusive state (I) or a non intrusive state (NI). Similarly an IDS, depending on the analysis of data sources, can report an alarm (A) or not (NA). The ROC of a detector is a plot of the conditional probabilities $P(A|I)$ (hit rate H) vs. $P(A|NI)$ (false alarm rate F) as shown in Table 1.

Table 1. Conditional probabilities that an IDS detects the system state

	System state	
Detector's report	No intrusion (NI)	Intrusion (I)
No alarm (NA)	FVN $(1-F)$	FFN $(1-H)$
Alarm (A)	FFP (F)	FVP (H)

The expected cost of a detector on a certain operating point can be computed analyzing the decision tree of Figure 1.

Decision or action nodes, which are displayed as squares, are under control of the decision maker, who will choose which branch to follow. Conversely, the circles represent event nodes that are subject to uncertainty. A probability dis-

tribution represents the uncertainty about which branch will happen following an event node. Event nodes probabilities are defined as follows:

- p_1: is the probability that the detector reports no alarm.
- p_2: is the conditional probability of no intrusion given that the detector reports no alarm.
- p_3: is the conditional probability of no intrusion given that the detector reports an alarm.

Conditional probabilities $1 - p_2$ and $1 - p_3$ can be expressed in terms of hit and false alarm rates:

$$1 - p_2 = P(I|NA) = \frac{P(NA|I)P(I)}{P(NA)} = \frac{(1-H)p}{p_1} \tag{1}$$

$$1 - p_3 = P(I|A) = \frac{P(A|I)P(I)}{P(A)} = \frac{Hp}{1-p_1} \tag{2}$$

In the model we propose, there is a cost C if the IDS responds, irrespective the intrusion took place or not, that corresponds to the countermeasures taken. L represents the losses if there is a false negative. The decision maker (the own IDS if the response is automatic or the network administrator if it is not) will follow the strategy that minimizes the expected cost. In order to compute this expected cost it is necessary to calculate the expected cost conditional on the detector's report. The four possibilities are summarized in Table 2. The prior probability that an intrusion happens is represented by p (P(I)).

Table 2. Expected cost of responses vs. detector's report

	Response	
Detector's report	No	Yes
No alarm	$L(1-p_2) = \frac{L(1-H)p}{p_1}$	$Cp_2 + C(1-p_2) = C$
Alarm	$L(1-p_3) = \frac{LHp}{1-p_1}$	$Cp_3 + C(1-p_3) = C$

Thus, if the report of the detector is known, the minimal expected cost can be computed. If there is no alarm the expression for the expected cost under this condition is:

$$M_{NA} = \min\{L(1-p_2), C\} = \min\{\frac{L(1-H)p}{p_1}, C\} \tag{3}$$

Similarly, the expected cost given an alarm is:

$$M_A = \min\{L(1-p_3), C\} = \min\{\frac{LHp}{1-p_1}, C\} \tag{4}$$

Finally, the expected cost of operating at a given operating point (a point in the ROC curve), is the sum of the products of the probabilities of the detector's reports and the expected costs of operating conditional on the reports. Then the expression is:

$$\begin{aligned} & p_1 \min\{\tfrac{L(1-H)p}{p_1}, C\} + (1-p_1)\min\{\tfrac{LHp}{1-p_1}, C\} \quad = \\ & = \min\{L(1-H)p, C((1-F)(1-p)+(1-H)p)\} + \\ & + \qquad \min\{LHp, C(F(1-p)+Hp)\} \end{aligned} \tag{5}$$

Consequently, the expected cost by unit loss (M) is:

$$\begin{aligned} M = & \min\{(1-H)p, \frac{C}{L}((1-F)(1-p)+(1-H)p)\} + \\ & + \min\{Hp, \frac{C}{L}(F(1-p)+Hp)\} \end{aligned} \tag{6}$$

It is important to note that this formulation includes the possibility of taking actions against the report of the detector if this action leads to a lower expected cost.

Next, a metric that measures the value of an IDS is introduced. First some concepts need to be defined.

The expected expense of a perfect IDS (the one that achieves H=1 and F=0) by unit loss is (from expression (6))

$$M_{per} = \min\{p, \frac{C}{L}p\} = p\min\{1, \frac{C}{L}\} \tag{7}$$

In addition, an expression is needed for the expected cost when only information about the probability of intrusion is available (no IDS working). In this situation, the decision maker can adopt two strategies: always protect taking some precautionary action (incurring in a cost C) or never protect (incurring in losses pL). Consequently, the decision maker will respond if $C < pL$ and will not if $C > pL$. Then, the expected cost by unit loss is:

$$M_{prob} = \min\{p, \frac{C}{L}\} \tag{8}$$

Accordingly, the value of an IDS is defined as the reduction it gives on the expected cost over the one corresponding to the only knowledge of the probability of intrusion, normalized by the maximum possible reduction.

$$V = \frac{M_{prob} - M}{M_{prob} - M_{per}} \tag{9}$$

As a result, if an IDS is perfect at detecting intrusions its value is 1. Conversely, an IDS that does not improve a predictive system solely based on the probability of intrusion has a value less or equal than 0.

The metric of value is very useful because it includes all the relevant parameters involved in the evaluation of IDS effectiveness. A similar metric was proposed in [18] but it did not manage the possibility that a decision is made contrary to the detector's report.

3 Parametric IDS

Several IDS can not be tuned to work at different operating points. This is a limiting feature. An operating point is defined by a pair (F,H). In order to adapt to different operating conditions, an IDS should be able to work at different operating points [2]. In consequence, we propose a very simple parametric IDS that consists on combining different non parametric detection techniques.

The question we want to answer is in what sense the mere combination of machine learning algorithms outperforms the individual approaches on intrusion detection domain. The way the parametric IDS works is the following. Let us consider an event happens on the monitored system. If the fraction of individual models that state the event is hostile is over a certain probability threshold p_t, then the parametric IDS will assume it as intrusive. p_t can be tuned from 0 to 1 in such a way that different predictions about the event are produced. In other words, the parametric detection depends on how many machine learning models predicted the event as hostile and on the threshold p_t.

Therefore, the hit rate and false alarm rate of the parametric IDS, computed over an event dataset, depend on this threshold p_t [18].

$$H = H(p_t) \quad F = F(p_t) \quad \forall p_t \in [0,1] \tag{10}$$

Consequently, the value of the IDS, as defined in equation (9), also depends on p_t.

$$V = V(p_t) \quad \forall p_t \in [0,1] \tag{11}$$

For a fixed $\frac{C}{L}$ relationship, the optimum value of the IDS is:

$$V_{opt} = \max_{pt} V(p_t) \quad \forall p_t \in [0,1]. \tag{12}$$

4 Experimental Setup

The main problem to test IDS effectiveness is the absence of non controversial benchmarks. It is not an easy task to build such a benchmark because different requirements must be considered to test different IDS [17]. There are also problems in repeating experiments with real data (privacy problems) and simulated data is under suspicion because is hard to establish how close the artificial data is

Table 3. Attack and normal instances in original KDD training and test datasets

	Training	Test
Normal instances	97277	60593
Attack instances	396743	250436
Total	494020	311029
% of normal instances	19.69	19.48
% of attack instances	80.31	80.52

Table 4. Attack and normal instances in original training and test 1999 KDD datasets (after filtering)

	Training	Test
Normal instances	97277	60593
Attack instances	4887	2650
Total	102164	63243
% of normal instances	95.22	95.81
% of attack instances	4.78	4.19

to the real one. The ad-hoc methodology that is prevalent in today's testing and evaluation of network intrusion detection systems makes it difficult to compare different algorithms and approaches [1]. Although the best way to evaluate any intrusion detection algorithm is to use live or recorded real traffic from the site where the algorithm is going to be deployed, there is a need of public datasets in order to evaluate proposals in a repeatable manner.

In order to model the sensor agents of our system , we needed a datFor the experiments we have used the well known 1999 KDD dataset[1] [4]. It derives from from MIT/LL 98 evaluation. Training and testing datasets were created at Columbia University. KDD dataset is the most frequently used dataset to test machine learning algorithms on the intrusion detection domain (e.g. [7, 20, 10]). It was firstly employed for a machine learning competition in order to test different classifiers over the intrusion detection domain. A complete description of the data mining process can be found in [11]. They are currently available at California University website[2]. Next, we are going to review the dataset briefly (a general description can be found in [4]). Each connection record defines a TCP session and is described by 41 attributes (38 numeric and 3 nominal), and the corresponding class that indicates if the record represents normal or hostile activity. The number of normal and attack examples are summarized in Table 3. As it can be seen, the percentage of attacks is extraordinary high both on training and test datasets. This situation is not expectable in a real environment because, normally, the probability of intrusion is very low. However we have shown in previous sections the importance of the probability of intrusion when evaluating IDS effectiveness. Consequently the experiments we carried out have been done over original and filtered data. The filtering consisted on getting rid of the most

[1] In fact the reduced version of the dataset (10% of the complete one)
[2] http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html

common attack types both in training and test datasets. The resulting number of examples is summarized in Table 4. It is important to note that, after filtering, attack frequency remains under 5%.

We have experimented with 10 machine learning algorithms over the complete and the filtered datasets. Two of these algorithms are based on decision trees (*ADTree, J48*), five in rules (*ConjuctiveRule, DecisionStump, DecisionTable, OneR, PART*), one on bayesian learning (*NaïveBayes*) and two in simpler techniques (*HyperPipes, VFI*)[3]. Consequently, 10 models were built from training dataset and tested against the test dataset. The parametric IDS built from the individual models is tested against test dataset as well. The following section shows the results.

5 Experimental Results

Figure 2 shows the operating points of each machine learning model against the ROC points of the parametric IDS over the original dataset. The main conclusions from the analysis of this figure are:

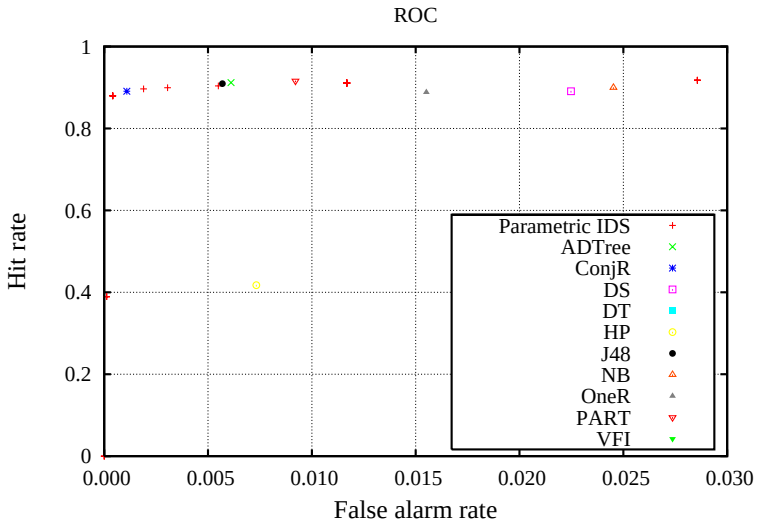

Fig. 2. ROC space of parametric IDS vs. individual models over original KDD test dataset

[3] The names correspond to the implementation name in WEKA software [26]

– Parametric IDS can work on different operating points.
– As p_t is increased, ROC points of the parametric IDS present lower H and F.
– It is difficult to state if the parametric model outperforms the non parametric components.

Then, in order to compare the different models, the metric of value we proposed in section 2 is used. Figure 3a) shows non parametric value curves while Figure 3b) shows those that correspond to the parametric IDS.

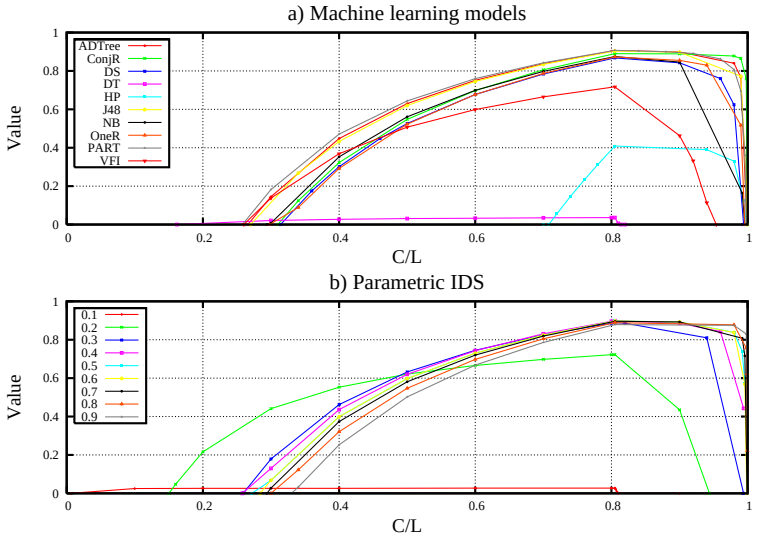

Fig. 3. Value curves over original KDD test dataset. a) Machine learning models and b) Parametric IDS

The main conclusions from these results are:

– *PART* is the machine learning model that achieves better results over a wider $\frac{C}{L}$ range. In addition, the greatest value of the non parametric IDS is also obtained by *PART*. The corresponding operating point is (F, H)=(0.009,0.916) with V=0.907 for $\frac{C}{L}$=0.805. This means that under these cost conditions and with the unrealistic probability of intrusion of the test dataset, *PART* is expected to have a value that is 90.7% of the perfect IDS.
– Parametric IDS obtains results that are similar to *PART* for high $\frac{C}{L}$ relationships. But for lower $\frac{C}{L}$ the parametric system is much better. For instance,

if $\frac{C}{L}$=0.3 the value of the parametric IDS is 0.441 (p_t=0.2). Under the same operating conditions, the best non parametric model ($PART$) has a value of 0.185. This means an increase of 25.6%. In addition, the $\frac{C}{L}$ range where the parametric is valuable is 25% wider than any of the individual models.

As we have stated, the probability of intrusion that KDD dataset shows is not realistic. Next results show the effectiveness of the systems tested after filtering the data as explained in section 4. Figure 4 shows the corresponding points on ROC space and Figure 5 exposes the value curves. From the ROC curves is difficult to say if *J48* is more effective than *NaïveBayes* or contrary. Figure 5a) shows that for $\frac{C}{L}$ >0.41 *J48* is preferred but for lower values *NaïveBayes* is better. It is important to note that normally L is much bigger than C because the losses when an intrusion happens are usually bigger than the cost of taking some action of response. Therefore, it is important to study the behaviour for low $\frac{C}{L}$ relationships.

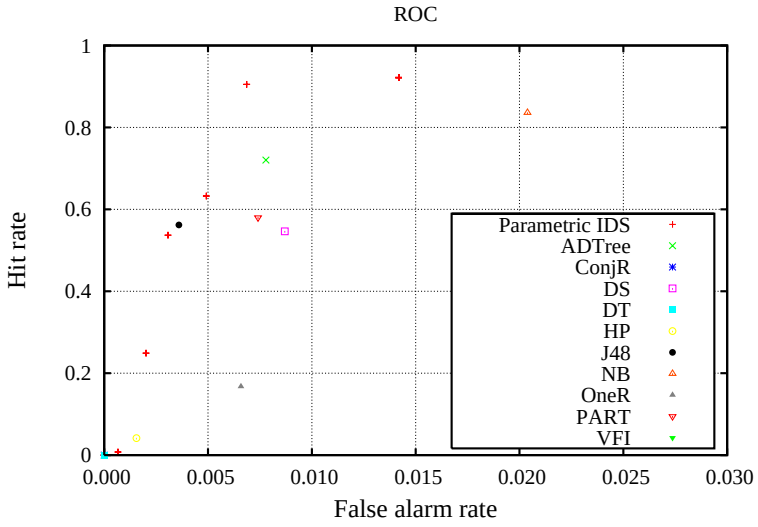

Fig. 4. ROC space of parametric IDS vs. individual models over filtered KDD test dataset

On the other hand, parametric IDS clearly outperforms any of the machine learning models. The envelope of the first includes the envelope of the composition of the second ones. To put it simply, there is no cost relationship where an individual algorithm outperforms the parametric IDS. In fact, the maximum of

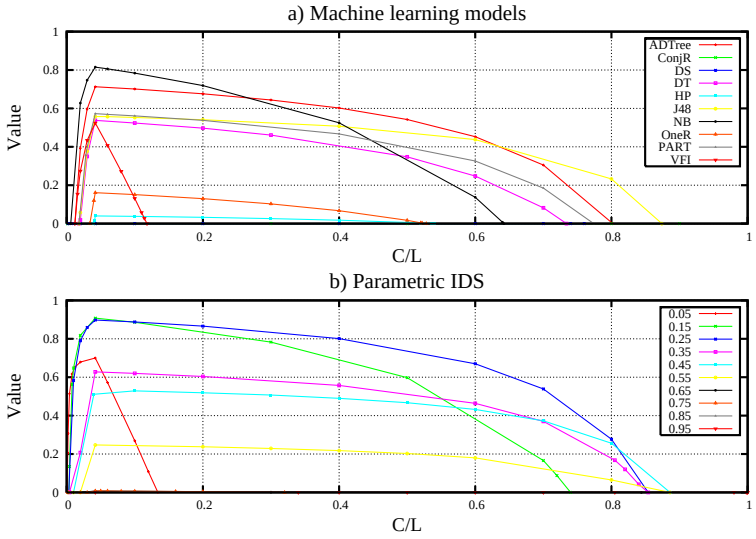

Fig. 5. Value curves computed over KDD filtered test dataset

the parametric envelope is a 9.3% over the best non parametric curve. Furthermore, the range with value is also bigger for the parametric approach and, what is more important, for low $\frac{C}{L}$ relationships ($0.00012 < \frac{C}{L} < 0.018$) the combined model is valuable and none of its components separately has any value.

6 Conclusions

This paper has proposed a method for comparing IDS effectiveness from a perspective of the utility of IDS detections. Experimental work has been done focusing on supervised machine learning algorithms. Results show that, generally, the best classifier highly depends on the operating conditions (summarized in the cost relationship and in the probability of intrusion). ROC curves are a good performance representation if an IDS has greater hit rate and lower false alarm rate than the one it is compared with. Else some alternative representation is needed. In fact, even when ROC curves are useful they do not give quantitative information about the dominance of one machine learning schema over another (this difference depends on the costs and on the probability of intrusion). The metric of value we propose gives a quantitative measure of how better a model is under different operating conditions. Furthermore, value curves state if a classifier is worthless (no better than a predictive system based on uncertainty) and allow to know how far a model is from a perfect one.

Results on the KDD dataset confirm that different classifiers stand out at different operating conditions. So, over the intrusion detection domain, it is very important to compare proposals considering the environment faced. In addition, the proper combination of different machine learning techniques produces a more effective IDS in the sense that it can operate under different scenarios (versatility) getting greater absolute values.

In conclusion, we encourage to adopt this evaluating methodology when evaluating and testing data mining approaches over the intrusion detection domain in order to avoid out of context conclusions.

References

1. N. Athanasiades, R. Abler, J. G. Levine, H. L. Owen, and G. F. Riley. Intrusion detection testing and benchmarking methodologies. In *Proceedings of the International Information Assurance Workshop, IWIA '03*, pages 63–72, Maryland, USA, 2003.
2. S. Axelsson. The base-rate fallacy and the difficulty of intrusion detection. *ACM Transactions on Information and System Security, TISSEC*, 3(3):186–205, 2000.
3. C. Drummond and R. C. Holte. Explicitly representing expected cost: an alternative to roc representation. In *Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '00*, pages 198–207, Nueva York, USA, 2000. ACM Press.
4. C. Elkan. Results of the KDD'99 classifier learning contest. September 1999.
5. J. E. Gaffney and J. W. Ulvila. Evaluation of intrusion detectors: A decision theory approach. In *Proceedings of the IEEE Symposium on Security and Privacy, SP '01*, pages 50–, Washington, DC, USA, 2001. IEEE Computer Society.
6. G. Giacinto, R. Perdisci, and F. Roli. Alarm clustering for intrusion detection systems in computer networks. In *Proceedings of the 4th International Conference on Machine Learning and Data Mining in Pattern Recognition, MLDM 2005*, pages 184–193, Leipzig, Germany, July 2005.
7. G. Giacinto, F. Roli, and L. Didaci. Fusion of multiple classifiers for intrusion detection in computer networks. *Pattern Recognition Letters*, 24(12):1795–1803, 2003.
8. ISO. Information technology - security techniques - it intrusion detection frameworks. Technical report, 2002. ISO/IEC TR 15947.
9. R. W. Katz and A. H. Murphy. *Economic Value of Weather and Climate Forecasts*. Cambridge University Press, UK, 1997.
10. P. Laskov, P. Düssel, C. Schäfer, and K. Rieck. Learning intrusion detection: supervised or unsupervised. In *Proceedings of the Thirteenth International Conference on Image Analysis and Processing, ICIAP 2005*, Cagliari, Italy, 2005.
11. W. Lee and S. J. Stolfo. A framework for constructing features and models for intrusion detection systems. *ACM Transactions on Information and System Security*, 3(4):227–261, 2000.
12. R. Lippmann, 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: The 1998 DARPA off-line intrusion detection evaluation. In *Proceedings of the DARPA Information Survivability Conference and Exposition*, Los Alamitos, California, USA, 2000. IEEE Computer Society Press.

13. R. Lippmann, J. W. Haines, D. J. Fried, J. Korba, and K. Das. The 1999 darpa off-line intrusion detection evaluation. *Computer Networks*, 34(4):579–595, 2000.
14. M. V. Mahoney and P. K. Chan. An analysis of the 1999 darpa/lincoln laboratory evaluation data for network anomaly detection. In *Proceedings of the Sixth International Workshop on Recent Advances in Intrusion Detection*, pages 220–237, Pittsburgh, USA, 2003.
15. M. A. Maloof. *Machine Learning and Data Mining for Computer Security: Methods and Applications (Advanced Information and Knowledge Processing)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2005.
16. J. McHugh. Testing intrusion detection systems: a critique of the 1998 and 1999 darpa intrusion detection system evaluations as performed by lincoln laboratory. *ACM Transactions on Information and System Security*, 3(4):262–294, 2000.
17. P. Mell, V. Hu, R. Lippman, J. Haines, and M. Zissman. An overview of issues in testing intrusion detection, June 2003. National Institute of Standards and Technologies. Internal report 7007.
18. A. Orfila, J. Carbó, and A. Ribagorda. Fuzzy logic on decision model for ids. In *Proceedings of the Twelveth IEEE International Conference on Fuzzy Systems, FUZZ-IEEE '03*, volume 2, pages 1237–1242, St. Louis, Missouri, USA, may 2003.
19. A. Orfila, J. Carbó, and A. Ribagorda. Intrusion detection effectiveness improvement by a multi-agent system. *International Journal of Computer Science & Applications*, 2(1):1–6, January 2005.
20. M. Sabhnani and G. Serpen. Kdd feature set complaint heuristic rules for r2l attack detection. In *Security and Management*, pages 310–316, 2003.
21. A.K. Sen. Choice functions and revealed preferences. *Review of Economic Studies*, 38:307–317, 1971.
22. J. A. Swets, R. Dawes, and J. Monahan. Psychological science can improve diagnostic decisions. *Psychological Science in the Public Interest*, 1(1):1–26, 2000.
23. B. K. Sy. Signature-based approach for intrusion detection. In *Proceedings of the 4th International Conference on Machine Learning and Data Mining in Pattern Recognition, MLDM 2005*, pages 526–536, Leipzig, Germany, July 2005.
24. J. W. Ulvila and J. E. Gaffney. Evaluation of intrusion detection systems. *Journal of Research of the National Institute of Standards and Technology*, 108(6):453–473, nov–dec 2003.
25. J. W. Ulvila and J. E. Gaffney. A decision analysis method for evaluating computer intrusion detection systems. *Decision Analysis*, 1(1):35–50, March 2004.
26. I. H. Witten and E. Frank. *Data mining: practical machine learning tools and techniques*. Morgan Kaufmann Publishers Inc., San Francisco, California, USA, June 2005.