Parzen-Window Network Intrusion Detectors

Dit-Yan Yeung Calvin Chow

Department of Computer Science, Hong Kong University of Science and Technology
Clear Water Bay, Kowloon, Hong Kong

Abstract

Network intrusion detection is the problem of detecting anomalous network connections caused by intrusive activities. Many intrusion detection systems proposed before use both normal and intrusion data to build their classifiers. However, intrusion data are usually scarce and difficult to collect. We propose to solve this problem using a novelty detection approach. In particular, we propose to take a nonparametric density estimation approach based on Parzen-window estimators with Gaussian kernels to build an intrusion detection system using normal data only. To facilitate comparison, we have tested our system on the KDD Cup 1999 dataset. Our system compares favorably with the KDD Cup winner which is based on an ensemble of decision trees with bagged boosting, as our system uses no intrusion data at all and much less normal data for training.

1. Introduction

1.1. Host-Based versus Network-Based

Intrusion detection refers to a certain class of system attack detection problems. Many intrusion detection systems built thus far are based on the general model proposed by Denning [5]. From a high-level view, the goal is to find out whether or not a system is operating normally. Abnormality or anomaly in the system behavior may indicate successful exploitation of system vulnerabilities.

Host-based intrusion detection systems detect possible attacks into individual computers. Such systems typically use information specific to the operating systems of the target computers. On the other hand, *network-based* intrusion detection systems monitor network behavior by examining the content as well as the format of network data packets, which typically are not specific to the exact operating systems used by individual computers as long as these computers can communicate with each other using the same network protocol. For both types of systems, one may take a data mining approach by "mining" through the host-based or network-based data to detect possible attacks from internal or external intruders.

In this paper, our focus is on network-based intrusion detection systems.

1.2. Classification versus Novelty Detection

Typical *classification* problems can be formulated as follows. A discriminative classifier is built using training examples from all c (≥ 2) classes, so that it can classify each presented pattern into one of c classes with as low generalization error as possible.

While many pattern recognition problems fall into this category, some other problems are best formulated differently as *novelty detection* [1, 4, 7] problems. In a probabilistic sense, novelty detection is equivalent to deciding whether an unknown test pattern is produced by the underlying distribution that corresponds to the training set of normal patterns. While novelty detection problems appear to be similar to 2-class classification problems, a major difference is that they typically use only normal patterns as training examples to build a generative model of normal behavior. The novelty detection approach is particularly attractive under situations where novel or abnormal patterns are expensive or difficult to obtain for model construction.

In this paper, the novelty detection approach is adopted.

1.3. Our Research

Over the past few years, several intrusion detection contests, such as DARPA 1998, DARPA 1999, and KDD Cup 1999, were held to evaluate results in intrusion detection research. Many network intrusion detection methods have been proposed in the research community, e.g., [2, 3, 6, 8, 9, 10, 11, 14, 17, 18]. However, almost all of them have to use both normal and intrusion traffic data for classifier training. Thus the problem is essentially a classification problem. In practice, it is not always possible to obtain sufficient intrusion data. Our objective is to take the

novelty detection approach without requiring intrusion data for training.

The rest of this paper is organized as follows. In Section 2, we will review the density estimation approach to novelty detection and present our model based on Parzen-window estimators. In Section 3, the KDD Cup dataset will be described. Experimental results obtained using our model will be presented in Section 4. We will also compare our results with those obtained by the KDD Cup winner. Finally, some concluding remarks will be made in Section 5.

2. Parzen-Window Estimators for Novelty Detection

2.1. Density Estimation Approach

One approach to novelty detection is based on *density estimation*. It assumes a probabilistic generative model for the observed data. Density estimation refers to the process of estimating the underlying density function such that the model can best describe the data. The learned model is then used to detect novel patterns based on some criteria derived from statistical measures, such as likelihood. Some previous novelty detection methods based on this approach include [1, 12, 15, 16].

Since simple parametric density functions such as Gaussian are too restrictive for modeling real-world data distributions, the simple *parametric* density estimation approach is inappropriate for novelty detection. Instead, *semiparametric* or *nonparametric* methods are usually used. The most popular semiparametric method is based on Gaussian mixture models [15, 16]. Another possible method is to use regression trees [12]. Although semiparametric methods can usually give parsimonious representations, they require a parameter estimation process that could be computationally intensive. Nonparametric density estimation has also been used for novelty detection [1]. Although the amount of training data required could be very large and hence testing unknown patterns on the model becomes slow, the advantage of this approach is that essentially no training is required. Provided that sufficient data are available, the nonparametric approach can model arbitrary distributions without being restricted to special functional forms. Moreover, nonparametric models can easily be adapted under situations with time-varying data distributions. For these reasons, we will use a nonparametric method for this work.

2.2. Parzen-Window Density Estimation

Parzen introduced a nonparametric method for estimating density functions [13]. Let $p(\mathbf{x})$ be the density function to be approximated. Given a set $D = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$ of n i.i.d. examples drawn according to $p(\mathbf{x})$, the Parzen-window estimate of $p(\mathbf{x})$ based on the n examples in D is

$$\hat{p}(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^{n} \delta_n(\mathbf{x} - \mathbf{x}_i)$$

where $\delta_n(\cdot)$ is a kernel function with localized support and its exact form depends on n.

We choose to use Gaussian kernel functions for two reasons. First, the Gaussian function is smooth and hence the estimated density function $\hat{p}(\mathbf{x})$ also varies smoothly. Second, if we assume a special form of the Gaussian family in which the function is radially symmetrical, the function can be completely specified by a variance parameter only. Thus $\hat{p}(\mathbf{x})$ can be expressed as a mixture of radially symmetrical Gaussian kernels with common variance σ^2:

$$\hat{p}(\mathbf{x}) = \frac{1}{n(2\pi)^{d/2}\sigma^d}\sum_{i=1}^{n} \exp\left\{-\frac{\|\mathbf{x} - \mathbf{x}_i\|^2}{2\sigma^2}\right\}$$

where d is the dimensionality of the feature space.

2.3. Novelty Detection as Hypothesis Test

Let ω_1 denote the state of nature corresponding to normality and ω_0 denote that for anomaly or novelty. The prior probabilities are denoted as $P(\omega_1)$ and $P(\omega_0)$ and the probability density functions are denoted as $p(\mathbf{x}|\omega_1)$ and $p(\mathbf{x}|\omega_0)$. Decisions are made according to the Bayes decision rule: $\mathbf{x} \in \omega_1$ if and only if $P(\omega_1|\mathbf{x}) > P(\omega_0|\mathbf{x})$ or $p(\mathbf{x}|\omega_1) > p(\mathbf{x}|\omega_0)(P(\omega_0)/P(\omega_1))$. For a given input $\mathbf{x}$, the decision rule above corresponds to comparing the likelihood $p(\mathbf{x}|\omega_1)$ against a threshold. This threshold varies with $\mathbf{x}$ unless $p(\mathbf{x}|\omega_0)$ is uniformly distributed with respect to $\mathbf{x}$. However, although it is possible to estimate the prior probability $P(\omega_1)$ (and hence $P(\omega_0)$) from data, it is very difficult to model the distribution of anomalous or novel events. This calls for a different formulation.

Our alternative formulation can be justified from a hypothesis testing perspective. Suppose $\mathcal{M}$ is the model built from normal data. Given an unknown case $\mathbf{x}$ from the test set, we want to decide whether $\mathbf{x} \in \omega_1$. This problem is formulated as applying a statistical test. Suppose we generate a sufficiently large sample $\mathcal{V}$ from $\mathcal{M}$. For an arbitrary example $\mathbf{y} \in \mathcal{V}$, the log-likelihood of $\mathbf{y}$ with respect to $\mathcal{M}$ is denoted as $L(\mathbf{y}) = \log p(\mathbf{y}|\mathcal{M})$. Similarly, the log-likelihood of $\mathbf{x}$ is $L(\mathbf{x}) = \log p(\mathbf{x}|\mathcal{M})$. Based on the empirical probability distribution of $L(\mathbf{y})$ over $\mathcal{V}$, we test the hypothesis that $L(\mathbf{x})$ is drawn from the distribution of the log-likelihood of the random examples in $\mathcal{V}$, i.e., $P(L(\mathbf{y}) \leq L(\mathbf{x})) > \psi$ for some parameter $0 < \psi < 1$. We reject the null hypothesis if the probability is not greater than ψ, implying that $\mathbf{x} \notin \omega_1$ or $\mathbf{x} \in \omega_0$.

In practice, it is not always possible to use $\mathcal{M}$ as a generative model. Instead of using $\mathcal{M}$ to generate a random sample $\mathcal{V}$, we choose to set aside a separate set of normal data to approximate $\mathcal{V}$. We call this set of data the *threshold determination set*. Note that ψ is equal to the *false detection rate* (FDR) or *false alarm rate* for the threshold determination set. Each chosen value of FDR (and hence ψ) will induce a threshold in log-likelihood against which test cases can be compared and determined as either normal or intrusive.

3. Dataset for Experiments

In our experiments, we use the dataset from KDD Cup 1999. The network traffic data are connection-based, meaning that each data record corresponds to one network connection. A network connection is a sequence of TCP/IP packets sent during a period of time between two IP addresses according to some well-defined network protocol.

There are three symbolic features and 38 numerical features. In addition, a label indicating whether the record is normal or intrusive is provided. The features include some basic features in the packet header, some features suggested by domain knowledge, and some temporal features such as the number of connections to the same host in the past two seconds. We represent each symbolic feature by a group of binary-valued features. The resulting feature vectors have a total of 119 dimensions.

The dataset has four intrusion categories: *probing*, *denial-of-service* (DoS), *user-to-root* (U2R), and *remote-to-local* (R2L). We mainly use two performance measures in our experiments. The *true acceptance rate* (TAR) measures the percentage of normal connections in the test set that are classified as normal, whereas the *true detection rate* (TDR) measures the percentage of intrusive connections in the test set that are detected as intrusions.

4. Experimental Results

4.1. Comparison with KDD Cup Winner

4.1.1 TAR and TDR as Performance Measures

We use 30000 randomly sampled normal connections as training data to estimate the density of a model. Another 30000 randomly sampled normal connections form the threshold determination set, which has no overlap with the training set. To reduce the effect due to random sampling, three trials have been carried out separately with three randomly sampled training sets. The average TAR and TDR values over the three trials are shown in Table 1.

The winning method of KDD Cup, submitted by Pfahringer, uses an ensemble of decision trees with bagged boosting. Since the KDD Cup is concerned with multi-class classification but we are interested only in normal/intrusion discrimination, we have converted the results of the winning method into our format. Specifically, the TDR measures the percentage of intrusive connections in the test set that are detected as intrusions, without considering whether they are classified into the correct intrusion categories. The best results are highlighted by rectangular boxes. Although the KDD winner gives slightly higher TAR for normal connections and slightly higher TDR for DoS intrusions, it gives significantly lower TDR values for other intrusion categories. In general, U2R and R2L attacks are more difficult to detect since they typically involve much fewer connections, but our method can give very high TDR for U2R attacks and can outperform the KDD Cup winner by more than three times for both U2R and R2L attacks.

Table 1. Comparison of our model at 99% confidence interval (i.e., FDR = 1%) and $\sigma = 0.01$ with the KDD Cup winner

Method	TAR	TDR			
	Normal	Probing	DoS	U2R	R2L
Ours	97.38%	99.17%	96.71%	93.57%	31.17%
KDD	99.45%	87.73%	97.69%	26.32%	10.27%

4.1.2 KDD Cup Scoring Scheme and Variant

Let us also compare our method with the KDD winning method based on the scoring scheme used in the KDD Cup. The scoring scheme uses the cost matrix in Table 2. The cost matrix is analogous to a loss function for pattern classification. In the matrix, the rows correspond to actual categories and the columns correspond to predicted categories. Note that the cost of failing to detect U2R or R2L attacks is higher than that for probing or DoS attacks because of the more serious implications of the former attack types.

Table 2. Scoring scheme for KDD Cup 1999

Truth	Prediction				
	Normal	Probing	DoS	U2R	R2L
Normal	0	1	2	2	2
Probing	1	0	2	2	2
DoS	2	1	0	2	2
U2R	3	2	2	0	2
R2L	4	2	2	2	0

As discussed above, our main interest is in performing normal/intrusion discrimination rather than multi-class classification. The standard cost matrix above has been modified accordingly. Using this modified scoring scheme, the average cost of our model trained with 30000 normal connections (as in Table 1) is equal to 0.2024. The corresponding average cost of the KDD Cup winner is 0.2263.

The main reason why our method performs better based on the scoring scheme is that our method can give significantly higher TDR values for U2R and R2L attacks, which are the attack types with higher penalties. Although our method is not always better because its TAR is lower, it is fair to say that our method can achieve performance comparable to the best methods, with the favorable characteristics that it requires no intrusion data at all and significantly less normal data for model training.

4.2. Sensitivity Analysis

To see how some model parameters can affect the performance of our model, we have performed some additional experiments for sensitivity analysis. An advantage of our model is that there are only very few parameters that need to be tuned. The major ones are the variance parameter of the Gaussian kernels and the sample size. Due to page limit, details of these experiments are not presented here. As a summary, we can conclude that the model is stable over relatively wide ranges of these two parameters.

5. Concluding Remarks

The major limitation of our Parzen-window method is its relatively high computational demand during testing, although it requires essentially no training time at all. Fortunately, since the Parzen-window method has characteristics similar to k-nearest-neighbor (k-NN) classifier, and many speedup schemes have been proposed for k-NN, our method can also take advantage of these previously proposed schemes.

An advantage of our nonparametric approach is that the models can easily adapt to data changes. Unlike many other models, our nonparametric models can simply integrate new training examples into the models without re-training the models from scratch. This makes our nonparametric approach particularly suitable for intrusion detection applications in continuously changing network environments.

Acknowledgments

This research has been supported by the Hong Kong University Grants Committee (UGC) under research grant AoE/E-01/99.

References

[1] C. Bishop. Novelty detection and neural network validation. *IEE Proceedings: Vision, Image and Signal Processing*, 141(4):217–222, 1994.

[2] J. Bonifácio Jr., A. Cansian, A. de Carvalho, and E. Moreira. Neural networks applied in intrusion detection systems. In *Proceedings of the International Joint Conference on Neural Networks*, volume 1, pages 205–210, Anchorage, AK, USA, 4–9 May 1998.

[3] J. Cannady. Applying CMAC-based on-line learning to intrusion detection. In *Proceedings of the International Joint Conference on Neural Networks*, volume 5, pages 405–410, Como, Italy, 24–27 July 2000.

[4] W. Daunicht. Autoassociation and novelty detection by neuromechanics. *Science*, 253(5025):1289–1291, 1991.

[5] D. Denning. An intrusion-detection model. *IEEE Transactions on Software Engineering*, 13(2):222–232, 1987.

[6] A. Ghosh and A. Schwartzbard. A study in using neural networks for anomaly and misuse detection. In *Proceedings of the Eighth USENIX Security Symposium*, pages 141–151, Washington, DC, USA, 23–26 August 1999.

[7] N. Japkowicz, C. Myers, and M. Gluck. A novelty detection approach to classification. In *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence*, volume 1, pages 518–523, Montréal, Quebec, Canada, 20–25 August 1995.

[8] W. Lee, S. Stolfo, and K. Mok. Mining audit data to build intrusion detection models. In *Proceedings of the Fourth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 66–72, New York, NY, USA, 27–31 August 1998.

[9] W. Lee and D. Xiang. Information-theoretic measures for anomaly detection. In *Proceedings of the IEEE Symposium on Security and Privacy*, pages 130–143, Oakland, CA, USA, 14–16 May 2001.

[10] R. Lippmann and R. Cunningham. Improving intrusion detection performance using keyword selection and neural networks. *Computer Networks*, 34(4):597–603, 2000.

[11] R. Lippmann, J. Haines, D. Fried, J. Korba, and K. Das. The 1999 DARPA off-line intrusion detection evaluation. *Computer Networks*, 34(4):579–595, 2000.

[12] D. Martinez. Neural tree density estimation for novelty detection. *IEEE Transactions on Neural Networks*, 9(2):330–338, 1998.

[13] E. Parzen. On estimation of a probability density function and mode. *Annals of Mathematical Statistics*, 33(3):1065–1076, 1962.

[14] V. Paxson. Bro: a system for detecting network intruders in real-time. *Computer Networks*, 31(23/24):2435–2463, 1999.

[15] S. Roberts. Novelty detection using extreme value statistics. *IEE Proceedings: Vision, Image and Signal Processing*, 146(3):124–129, 1999.

[16] S. Roberts and L. Tarassenko. A probabilistic resource allocating network for novelty detection. *Neural Computation*, 6(2):270–284, 1994.

[17] C. Sinclair, L. Pierce, and S. Matzner. An application of machine learning to network intrusion detection. In *Proceedings of the Fifteenth Annual Computer Security Applications Conference*, pages 371–377, Phoenix, AZ, USA, 6–10 December 1999.

[18] K. Yamanishi, J. Takeuchi, and G. Williams. On-line unsupervised outlier detection using finite mixtures with discounting learning algorithms. In *Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 320–324, Boston, MA, USA, 20–23 August 2000.