Information **S**ciences and **C**omputing
Volume 2013, Number 1, Article ID ISC070713, 8 pages
Available online at http://www.infoscicomp.com/

Research Article

A Novel Approach for Improve the Detection Rate in Intrusion Detection System Using Multilayer Perceptron Algorihtm

Anshuman Sharma[a] and M.R. Alone[b]

[a] 167 Goutam Nagar, Bhopal, India
[b] E-35 Balwant Nagar, Bhopal, India

Corresponding author: Anshuman Sharma; E-mail: anshuman515@gmail.com

Received 02 June 2013; Accepted 08 July 2013

Abstract: IDS which are increasingly a key part of system defense are used to identify abnormal activities in a computer system. In general, the traditional intrusion detection relies on the extensive knowledge of security experts, in particular, on their familiarity with the computer system to be protected. To reduce this dependence, various data-mining and machine learning techniques have been used in the literature. Traditional instance-based learning methods can only be used to detect known intrusions, since these methods classify instances based on what they have learned. They rarely detect new intrusions since these intrusion classes has not been able to detect new intrusions as well as known intrusions. In this paper, we propose neural network based method for network intrusion detection. These technique are applied to the KDD Cup 98 data set .In addition, a comparative analysis shows the advantage of Unsupervised Learning techniques over clustering-based Methods in identifying new or unseen attack.

Keywords: Intrusion detection system; neural network; data mining; false alarm.

1. Introduction

An intrusion detection system (IDS) is a component of the information security framework. Its main goal is to differentiate between normal activities of the system and behavior that can be classified as suspicious or intrusive [1]. The goal of intrusion detection is to build a system which would automatically scan network activity and detect such intrusion attacks. Once an attack is detected, the system administrator can be informed who can take appropriate action to deal with the intrusion.

IDS can be host-based (HIDS), network based (NIDS) or a combination of both types (Hybrid Intrusion Detection System). HIDS usually observes logs or system –calls on a single host, while a NIDS typically monitors traffic flows and Network packets on a network segment, and thus observes multiple hosts simultaneously. Generally, one deal with very large volumes of network data, and thus it is difficult and tiresome to

classify them manually in order to detect a possible intrusion. One can obtain labelled data by Simulating intrusions, but this will be limited only to the set of known attacks. Therefore, new types of attacks that may occur in future cannot be handled, if those were not part of the training data.

Even with manual classification, we are still limited to identifying only the known (at classification time) types of attacks, thus restricting our detection system to identifying only those types. To solve these difficulties, we need a technique for detecting intrusions when our training data is unlabeled, as well as for detecting new and un-known types of intrusions. A method that offers promise in this task is anomaly detection. Anomaly detection detects anomalies in the data (i.e. data instances in the data that deviate from normal or regular ones). It also allows us to detect new types of intrusions, because these new types will, by assumption, be deviations from the normal network usage. It is very difficult, if not impossible, to detect malicious intent of someone who is authorized to use the network and who uses it in a seemingly legitimate way. For example, there is probably no highly reliable way to know whether someone who correctly logged into a system is the intended user of that system, or if the password was stolen.

Under these assumptions we built a system which created clusters from its input data, then automatically labelled clusters as containing either normal or anomalous data instances, and finally used these clusters to classify network data instances as either normal or anomalous. Both the training and testing was done using 10% KDDCup'99 data [2], which is a very popular and widely used intrusion attack dataset. Most clustering techniques assume a well defined distinction between the clusters so that each pattern can only belong to one cluster at a time.

This supposition can neglect the natural ability of objects existing in multiple clusters. For this reason and with the aid of fuzzy logic, fuzzy clustering can be employed to overcome the weakness. The membership of a pattern in a given cluster can vary between 0 and 1. In this model a data object belongs to the cluster where it has the highest membership value.

In this paper we aim to propose a Neural Network based algorithm which is capable finding unseen attack and identify new attack.

2. Related Work

The problem of huge network traffic data size and the invisibility of intrusive patterns which normally are hidden among the irrelevant and redundant features have posed a great challenge in the domain of intrusion detection [1]. One way to address this issue is to reduce these input features n order to disclose the hidden significant features. Thus, an accurate classification can be achieved, besides identifying significant features that can represent intrusive patterns; the choice of classifier can also influence the accuracy and classification of an attack. The literature suggests that hybrid or assembling multiple classifiers can improve the accuracy of detection [14, 12]. Classifier ensembles also known as committees are aggregations of several classifiers whose individual predictions are combined in some manner (e.g., averaging or voting) to form a final prediction [6, 18]. An important advantage for combining redundant and complementary classifiers is to increase robustness, accuracy and

better overall generalization in most applications 18]. Mukkamala et al. [13] demonstrated the use of ensemble classifiers gave the best accuracy for each category of attack patterns. Ensemble methods aim at improving the predictive performance of a given statistical learning or model fitting technique. The general principle of ensemble methods is to construct a linear combination of some model fitting method, instead of using a single fit of the method. In designing a classifier, the first step is to carefully construct different connectional models to achieve best generalization performance for classifiers. Chebrolu et al. [12] proposed CART-BN approach, where CART performed best for Normal, Probe and U2R and the ensemble approach worked best for R2L and DoS. Meanwhile, Abraham and Jain. [2] illustrated that ensemble Decision Tree was suitable for Normal, LGP for Probe, DoS and R2L and Fuzzy classifier was for R2L. In their later work, Abraham et al [3] also demonstrated the ability of their proposed ensemble structure in modeling light-weight distributed IDS. Meanwhile, Mukkamala et al. [18] proposed three variants of Neural Networks, SVM and MARS as components in their IDS. This combining approach has demonstrated better performance when compared to single classifier approach. Giorgio et al. [7] took a slightly different approach. Their anomaly IDS was based on modular multiple classifier system where each module was designed for each group of protocols and services. Each module might contain either individual or combination of different classifiers. The modular architecture would allow putting a rejection threshold of each module as to optimize the overall attack detection rate given a desired total false alarm rate for the ensemble. They reported that there was an improvement on attack detection rate and significant reduction on false alarm.

3. Fundamental Theory

3.1 Intrusion Detection System

An Intrusion Detection System (IDS) constantly monitors actions in a certain environment and decides whether they are part of a possible hostile attack or a legitimate use of the environment. The environment may be a computer, several computers connected in a network or the network itself. The IDS analyzes various kinds of information about actions emanating from the environment and evaluates the probability that they are symptoms of intrusions. Such information includes, for example, configuration information about the current state of the system, audit information describing the events that occur in the system (e.g., event log in Windows XP), or network traffic. Several measures for evaluating

The more widely used measures are the True Positive (TP) rate, that is, the percentage of intrusive actions (e.g., error related pages) detected by the system, False Positive (FP) rate which is the percentage of normal actions (e.g., pages viewed by normal users) the system incorrectly identifies as intrusive, and Accuracy which is the percentage of alarms found to represent abnormal behavior out of the total number of alarms. In the current research TP, FP and Accuracy measures were adopted to evaluate the performance of the new methodology.

3.2 Network Profiling

Since the number of attacks is always increasing, IDS should be updated with signature for new attacks. Network profiling can help IDS to define labels of new

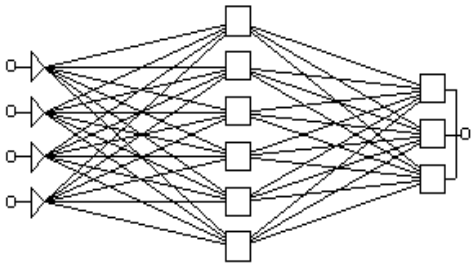
signatures. There are some problems in network profiling such as grouping the attacks that come through the network based on their types. Those problems can be solved using data mining techniques such as clustering and classification

3.3 Clustering Techniques

Cluster analysis is the process of partitioning data objects (records, documents, etc.) into meaningful groups or clusters so that objects within a cluster have similar characteristics but are dissimilar to objects in other clusters. Clustering can be viewed as unsupervised classification of unlabelled patterns (observations, data items or feature vectors), since no pre-defined category labels are associated with the objects in the training set. Clustering results in a compact representation of large data sets (e.g., collections of visited Web pages) by a small number of cluster centroids. Applications of clustering include data mining, document retrieval, image segmentation, and pattern classification (Jain *et al.* 1999). Thus, clustering of Web documents viewed by Internet users can reveal collections of documents belonging to the same topic. As shown by Sequeira and Zaki (2002), clustering can also be used for anomaly detection: normality of a new object can be evaluated by its distance from the most similar cluster under the assumption that all clusters are based on 'normal' data only. A good clustering method will produce high quality clusters in which similarity is high known as intra-classes and inter-classes where similarity is low. The quality of clustering depends upon both the similarly measure used by the method and it, s implementation and it is also measured by its ability to discover hidden patterns.
The concept of clustering algorithms is to build a finite number of clusters, each one with its own center, according to a given data set, where each cluster represents a group of similar objects. Each cluster encapsulates a set of data and here the similarities of the surrounded data are their distance to the cluster center.

Generally speaking, clustering techniques can be divided into two categories pair wise clustering and central clustering. The former also called similarity-based clustering, groups similar data instances together based on a data-pair wise proximity measure. Examples of this category include graph partitioning-type methods. The latter, also called centroid-based or model-based clustering, represents each cluster by a model, i.e., its centroid".

Central clustering algorithms[3] are often more efficient than similarity-based clustering algorithms. We choose centroid-based clustering over similarity-based clustering. We could not efficiently get a desired number of clusters, e.g., 100 as set by users. Similarity-based algorithms usually have a complexity of at least $O(N2)$ (for computing the data-pair wise proximity measures), where N is the number of data.



3.4 Classification

Classification is the task of assigning objects to one of several categories. A classification model can predict the class label of unknown object. Classification often used in biology and financial. In classification, datasets are divided into search domain and new sample. Classification technique builds a classification model from the search domain and decide the class label for each given input/object. Some classification algorithms are -Nearest Neighbor, Decision Tree, and Support Vector Machine (SVM).

4. Proposed Approach

We propose Neural Network based algorithms for network intrusion detection.

4.1 Multilayer Perceptron Algorithm

This section presents the architecture of a feed word neural network that is used to compress image in the research works. Multilayer-perceptron algorithm [7, 8] is a widely used learning algorithm in Artificial Neural Networks. The Feed-Forward Neural Network architecture is capable of approximating most problems with high accuracy and generalization ability. This algorithm is based on the error-correction learning rule. Error propagation consists of two passes through the different layers of the network, a forward pass, and a backward pass. In the forward pass the input vector is applied to the sensory nodes of the network and its effect propagates through the network layer by layer. Finally a set of outputs is produced as the actual response of the network. During the forward pass the synaptic weight of the networks are all fixed. During the back pass the synaptic weights are all adjusted in accordance with an error-correction rule. The actual response of the network is subtracted from the desired response to produce an error signal. This error signal is then propagated backward through the network against the direction of Input layer, Hidden Layer and Output Layer synaptic conditions. The synaptic weights are adjusted to make the actual response of the network move closer to the desired response.

4.2 Algorithm:

The algorithm for Perceptron Learning is based on the back-propagation rule discussed previously. This algorithm can be coded in any programming language, and in the case of this tutorial, Java for the applets. In this case we are assuming the use of the sigmoid function **f(net)** described earlier in the tutorial. This is because it has a simple derivative.

Algorithm:

1. Initialize weights and threshold: Set all weights and thresholds to small random values.

2. Present input and desired output: Present input $\boldsymbol{X_p} = \mathbf{x_0}, \mathbf{x_1}, \mathbf{x_2}, \ldots, \mathbf{x_{n-1}}$ and target output $\boldsymbol{T_p} = \mathbf{t_0}, \mathbf{t_1}, \ldots, \mathbf{t_{m-1}}$ where n is the number of input nodes and m is the number of output nodes. Set $\mathbf{w_0}$ to be $\mathbf{-ø}$, the bias, and $\mathbf{x_0}$ to be always 1. For pattern association,

$\boldsymbol{X_p}$ and $\boldsymbol{T_p}$ represent the patterns to be associated. For classification, $\boldsymbol{T_p}$ is set to zero except for one element set to 1 that corresponds to the class that $\boldsymbol{X_p}$ is in.

3. Calculate the actual output

Each layer calculates the following:

$$\mathbf{y_{pj} = f\,[w_0x_0 + w_1x_1 + ... + w_nx_n]}$$

This is then passes to the next layer as an input. The final layer outputs values $\mathbf{o_{pj}}$.

4. Adapts weights: Starting from the output we now work backwards.

$\mathbf{w_{ij}(t+1) = w_{ij}(t) + \tilde{n}\eth_{pj}o_{pj}}$, where $\tilde{\mathbf{n}}$ is a gain term and $\eth_{pj}$ is an error term for pattern **p** on node **j**.

5. For output units

$$\eth_{pj} = ko_{pj}(1 - o_{pj})(t - o_{pj})$$

For hidden units

$$\eth_{pj} = ko_{pj}(1 - o_{pj})[(\eth_{p0}w_{j0} + \eth_{p1}w_{j1} + ...+ \eth_{pk}w_{jk})]$$

where the sum(in the [brackets]) is over the **k** nodes in the layer above node **j**.

Step 5: Use simple majority of the category of nearest neighbors as the prediction value of the new sample

5. Evaluation Measures

To evaluate the system performance the following measures (based on Sequeira and Zaki 2002) were used.

True Positive Rate (TP) (also known as Detection Rate or Completeness): the percentage of terrorist pages receiving a rating above the threshold in the experiments, terrorist pages will be obtained from the users simulating terrorists.

False Positive Rate (FP): the percentage of regular Internet access pages that the system incorrectly determined as related to terrorist activities, i.e., the percentage of non-terrorist pages receiving a rating above threshold and suspected falsely as terrorists.

Accuracy: percentage of alarms related to terrorist behavior out of the total number of alarms. Since no benchmark data on content based intrusion detection is currently available, the results are compared to the best numbers achieved with ADMIT which is a command level method using the Means clustering algorithm to detect intruders.

6. Conclusion

An approach for a neural network based intrusion detection system, intended to classify the normal and attack patterns and the type of the attack, has been presented in this paper. We applied MLP method which increased the generalization capability of the neural network and at the same time decreased the training time. It should be mentioned that the long training time of the neural network was mostly due to the huge number of training vectors of computation facilities. However, when the neural network parameters were determined by training, classification of a single record was done in a negligible time. Therefore, the neural network based IDS can operate as an *online* classifier for the attack types that it has been trained for. The only factor that makes the neural network off-line is the time used for gathering information necessary to compute the features. A two layer neural network was also successfully used for the classification of connection records. Although the classification results were slightly better in the three layer network, application of a less complicated neural network was more computationally and memory wise efficient. From the practical point of view, the experimental results imply that there is more to do in the field of artificial neural network based intrusion detection systems. The implemented system solved a three class problem However; its further development to several classes is straightforward. As a possible future development to the present study, one can include more attack scenarios in the dataset. Practical IDSs should include several attack types. In order to avoid unreasonable complexity in the neural network, an initial classification of the connection records to normal and general categories of attacks can be the first step. The records in each category of intrusions can then be further classified to the attack types.

References

[1] S. Mahesh, T.R. Mahesh and M. Vinayababu, Using data mining techniques for detecting terror related activities on the web, Journal of Theoretical and Applied Information Technology, 2(3) (2010), 36-41.

[2] A. Abbasi and H. Chen, Applying authorship analysis to extremist group web forum messages, IEEE Intelligent Systems, Special Issue on Artificial Intelligence for National and Homeland Security, 20(5) (2005), 67-75.

[3] J. Baumes, M. Goldberg, M. Hayvanovych, M. Magdon-Ismail, W. Wallace and M. Zaki, Finding Hidden Group Structure in a Stream of Communications, in: S. Mehrotra, D.D. Zeng and H. Chen (Eds.), Proceedings of the IEEE Conference on Intelligence and Security Informatics, Los Alamitos, CA: IEEE, (2006), 201-212.

[4] H. Chen, Intelligence and security informatics: Information systems perspective, Decision Support Systems: Special Issue on Intelligence and Security Informatics, 41(3) (2006), 555-559.

[5] H. Chen, J. Qin, E. Reid, W. Chung, Y. Zhou, W. Xi et al., The Dark Web Portal: Collecting and Analyzing the Presence of Domestic and International Terrorist Groups on the Web, in: W.T. Scherer and B.L. Smith (Eds.), Proceedings of the 7th IEEE International Conference on Intelligent Transportation Systems, (2004), 106-111.

[6] Los Alamitos Report to Congress Regarding the Terrorism Information Awareness (TIA) Program, Submitted by the Secretary of Defense, Director of Central Intelligence and Attorney General, May (2003).

[7] J. Allen, A. Christie, W. Fithen, J. McHugh, J. Pickel and E. Stoner, State of the Practice of Intrusion Detection Technologies, CMU / SEI-99-TR-028, Carnegie Mellon Software Engg. Institute, 2000.

[8] KDDCup'1999dataset, http://kdd.ics.uci.edu/databases/kddcup'99/kddcup99.html.
[9] S. Theodoridis and K. Koutroubas, Pattern Recognition, Academic Press, 1999.
[10] Wikipedia-Cluster Analysis, http://en.wikipedia.org/wiki/cluster_analysis.
[11] J.Z. Shah and A.B. Salim, Fuzzy clustering algorithms and their application to chemical datasets, Proc. of the Post Graduate Annual Research Seminar, (2005), 36-40.
[12] Z. Chen, Data Mining and Uncertain Reasoning: An Integrated Approach, Willey, 2001.
[13] W. Chimphlee, et.al, Un-supervised clustering methods for identifying rare eventsin anomaly detection, Proc. of World Academy of Science, Engg. and Tech (PWASET), 8(Oct.) (2005), 253-258.
[14] J. Bezkek, Pattern Recognition with Fuzzy Objective Function Algorithms, Plennum Press, USA, 1981.
[15] S. Albayrak and F. Amasyali, Fuzzy C means clustering on medical diagnostic systems, International XII Turkish Symposium on Artificial Intelligence and Neural Networks, TAINN, (2003).
[16] W.O. Pedrycz, Knowledge Based Clustering, John Willey & Sons Inc., 2005.
