Pattern Based IDS Using Supervised, Semi-supervised and Unsupervised Approaches

Vinod K. Pachghare, Vaibhav K. Khatavkar, and Parag Kulkarni

Dept. of Computer Engg. & IT,
College Of Engineering, Pune, MS, India
{vkp.comp,vkk.comp}@coep.ac.in, paragakulkarni@yahoo.com
http://www.coep.org.in/

Abstract. Intrusion detection aims at distinguishing the behavior of the network. Due to rapid development of attack pattern, it is necessary to develop a system which can upgrade itself according to new attacks. Also detection rate should be high since attack rate on the network is very high. In response to this problem, Pattern Based Algorithm is proposed which has high detection rate and low false alarm rate. The work is divided into three parts: supervised approach, semi-supervised and unsupervised approach. Besides supervised learning approach, semi-supervised learning has attracted much attention in pattern recognition and machine learning for intrusion detection. Most of the semi supervised algorithms used for intrusion detection are binary classifiers, but our approach is to classify the data into multiclass. Our experimental results on KDD cup data set shows that the performance of the proposed method is more effective.

Keywords: Intrusion Detection System, Pattern Based Algorithm, Security, supervised learning, semi-supervised learning, Machine Learning, Neural Networks.

1 Introduction

There are two main approaches to design IDS: misuse based IDS and anomaly based IDS [20]. Both misuse and anomaly detection approaches are typically presented in terms of distinct training and testing phases.

Modern IDS's are extremely diverse in the techniques they employ to gather and analyze data. Rule-based analysis depends on sets of predefined rules that are provided by an administrator. This design approach usually results in an inflexible detection system that is unable to detect an attack if the sequence of events is slightly different from the predefined profile [5, 14]. The principal constituents of soft computing techniques are Fuzzy Logic (FL), Artificial Neural Networks (ANNs), Probabilistic Reasoning (PR), and Genetic Algorithms (GAs) [2].

In this paper we propose three approaches: supervised , unsupervised and semi supervised approach for intrusion detection. In the supervised approach we use the labeled data for training and unlabeled data for testing. However,

N. Meghanathan et al. (Eds.): CCSIT 2012, Part I, LNICST 84, pp. 542–551, 2012.

supervised learning approach requires labeled ground truth data. With the immense amount of network and host data available, expert labeling of the data is very expensive and time consuming. The labeled data available is often from controlled environments. This proves to be a bottleneck in applying supervised learning methods to detect novel or unknown attacks. Relying only on supervised learning methods which require a large amount of labeled data is impractical for real network environment. This motivates a need for a new and more practical learning framework.

Semi-supervised learning approach can leverage unlabeled data in addition to labeled ones. They have received significant attention, and are more suitable for real network environment because they require a small quantity of labeled data while still taking advantage of the large quantities of unlabeled data.

Several algorithms have been proposed for semi-supervised learning which is naturally inductive. Usually, they are based on an assumption, called the cluster assumption [9]. It states that the data samples with high similarity between them, must share the same label. This may be equivalently expressed as a condition that the decision boundary between the classes must pass through low density regions. This assumption allows the unlabeled data to regularize the decision boundary, which in turn influences the choice of classification models.

Many successful semi-supervised algorithms like TSVM and Semi-supervised SVM [3] follow this approach. These algorithms assume a model for the decision boundary, resulting in an inductive classifier. Manifold regularization [16] is another inductive approach, which is built on the manifold assumption. It attempts to build a maximum-margin classifier on the data, while minimizing the corresponding inconsistency with the similarity matrix. This is achieved by adding a graph-based regularization term to an SVM based objective function. A related approach called LIAM [16] regularizes the SVM decision boundary using a priori metric information encoded into the Graph Laplacian, and has a fast optimization algorithm.

The proposed semi supervised learning approach can use small amount of labeled data and large amount of unlabeled data for learning, and gives performances similar to supervised learning approach which using much larger amounts of labeled data.

The rest of the paper is organized as follows. Section 2 describes the related work about intrusion detection system. Section 3 describes our proposed approach for all the three approaches. Section 4 describes experiments and results followed by a conclusion in Section 5.

2 Related Work

2.1 Supervised Learning Based Approaches

In recent years, methods from machine learning and pattern recognition have been utilized to detect intrusions. Both supervised learning and unsupervised learning are used. There are mainly supervised neural network (NN)-based

approaches [15], [19], and support vector machine (SVM)-based approaches [12] are used in supervised learning for intrusion detection.

NN-based approaches: Many approaches have been proposed in neural network to distinguish between the behaviors of intrusions and normal. They unify the coding of categorical fields and the coding of character string fields in order to map the network data to the neural network. Some approaches propose hierarchical neural networks and evolutionary neural networks to detect intrusions.

SVM-based approaches: Mukkamala et al. [16] use SVMs to distinguish between normal and intrusions network behaviors and further identify important features for intrusion detection. The TreeSVM and ArraySVM have been proposed for solving the problem of inefficiency of the sequential minimal optimization algorithm for the large training data set in intrusion detection. Zhang and Shen [21] propose an approach for online training of SVMs for real-time intrusion detection based on an improved text categorization model. Also for intrusion detection, decision tree and discriminate analysis are applied. Comparisons between different classifiers and fusion of multiple classifiers for intrusion detection are studied in [18], [19], and [17].

2.2 Unsupervised Learning Based Approaches

Supervised learning methods for intrusion detection can only detect known intrusions. Unsupervised learning methods can detect the intrusions that have not been previously learned. K-means-based approaches and self-organizing feature map (SOM)-based approaches are the examples of unsupervised learning for intrusion detection [3].

K-means-based approaches: For intrusion detection, Guan et al. [22] propose a K-means-based clustering algorithm, which is named Y means. Xian et al. [23] combine the fuzzy K-means method and a clonal selection algorithm to detect intrusions. Jiang et al. [9] use the incremental clustering algorithm that is an extension of the K-means algorithm to detect intrusions.

SOM-based approaches: Pachghare et al. [3] gives various approaches of SOM like hierarchical SOM.

While these existing methods can obtain a high detection rate (DR), they often suffer from a relatively high false positive rate (FPR), which wastes a great deal of manpower. Meanwhile, their computational complexities are also oppressively high, which limits their applications in practice, because IDS would affect the regular tasks of the target systems if it employs too much resource. Adaboost is one of the most prevailing machine learning algorithms in recent years. Its computational complexity is generally lower than SOM, ANN and SVM in the case that the size of the data set is voluminous while the dimensionality is not too high. For this and other advantages, we employ Adaboost algorithm for our Pattern-based network security.

2.3 Semi-supervised Learning Based Approaches

Graph-based approaches represent both the labeled and the unlabeled examples by a connected graph, in which each example is represented by a vertex, and pairs of vertices are connected by an edge if the corresponding examples have large similarity. The well known approaches in this category include Harmonic Function based approach, Spectral Graph Transducer (SGT), Gaussian process based approach, Manifold Regularization and Label Propagation approach [11]. The optimal class labels for the unlabeled examples are found by minimizing their inconsistency with both the supervised class labels and the graph structure.

3 Proposed Algorithms

3.1 Supervised Algorithm

The framework of proposed algorithm is explained in our previous work [1].

Weak Classifier Design: A group of weak classifiers has to be prepared as inputs of Adaboost algorithm. They can be linear classifiers, ANNs or other common classifiers. In our algorithm, we select decision stumps as weak classifiers due to its simplicity. For every feature f, its value range could be divided into two non overlapping value subsets C_p^f and C_n^f, and the decision stump on f takes the form as follow:

$$h_f(x) = \begin{cases} +1 & x(f) \in C_p^f \\ -1 & x(f) \in C_n^f \end{cases}$$

where, $x(f)$ indicates the value of x on feature f.

Algorithm: In the AdaBoost algorithm, weak classifiers are selected iteratively from a number of candidate weak classifiers and are combined linearly to form a strong classifier for classifying the network data. In the AdaBoost algorithm,

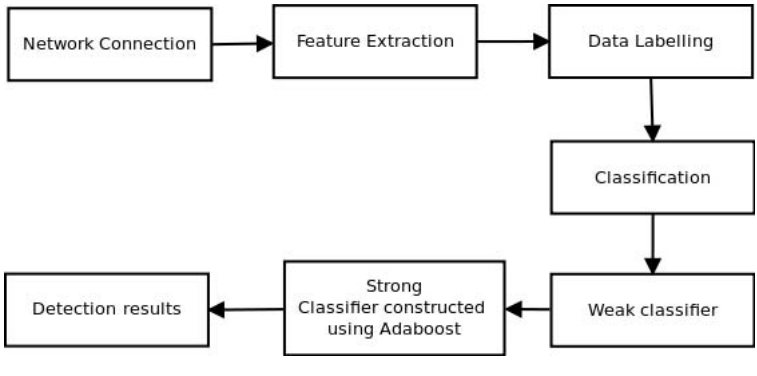

Fig. 1. Architecture for supervised IDS

weak classifiers are selected iteratively from a number of candidate weak classifiers and are combined linearly to form a strong classifier for classifying the network data.

Let $H = \left\{\tilde{h_f}\right\}$ be the set of constructed weak classifiers. Let the set of training sample data be $\{(x_1, y_1), ..., (x_i, y_i), ..., (x_n, y_n)\}$, where x_i denotes the i^{th} feature vector, $y_i \in \{+1, -1\}$ is the label of the i^{th} feature vector, denoting whether the feature vector represents a normal behavior or not; and n is the size of the data set. Let $\{w_1, ..., w_i, ..., w_n\}$ be the sample weights that reflect the importance degrees of the samples and, in statistical terms, represents an estimation of the sample distribution. The AdaBoost-based algorithm for intrusion detection is described as follows:

1. Initialize Weights as:

$$w_i(1)\,(n = 1, 2, ..., n)$$
$$\text{satisfying } \textstyle\sum_{i=1}^{n} w_i = 1$$

2. Observe the following for $(t = 1...T)$.
   (a) Let ϵ_j be the sum of the weighted classification errors for the weak classifier h_j

$$\epsilon_j = \sum_{i=1}^{n} w_i\,(t)\, I\left[y_i \neq h_j\,(x_i)\right] \tag{1}$$

   where,

$$I_{[\gamma]} = \begin{cases} 1 & \gamma = true \\ 0 & \gamma = false \end{cases} \tag{2}$$

   Choose, from constructed weak classifiers, the weak classifier $h(t)$ that minimizes the sum of the weighted classification errors

$$h(t) = arg \;\; min_{h,j \in H} \;\; \epsilon_j \tag{3}$$

   (b) Calculate the sum of the weighted classification errors $\epsilon\,(t)$ for the chosen weak classifier $h\,(t)$.
   (c) Let

$$\alpha\,(t) = 1/2 \log\left((1 - \epsilon\,(t))/\epsilon\,(t)\right) \tag{4}$$

   (d) Update the weights by

$$w_i\,(t+1) = \left(w_i\,(t) \exp\left(-\alpha\,(t)\, y_i h\,(t)\,(x_i)\right)/Z\,(t)\right) \tag{5}$$

   where,

$$Z\,(t) = \sum_{k=i}^{n} \exp\left(-\alpha\,(t)\, y_i h\,(t)\,(x_k)\right) \tag{6}$$

3. The strong classifier is defined by

$$H\,(t) = sign\left(\sum_{t=1}^{T} \alpha\,(t)\, h\,(t)\,(x)\right) \tag{7}$$

We explain two points:

- By combining the decision stumps for both categorical and continuous features into a strong classifier, the relations between categorical and continuous features are handled naturally, without any forced conversions between continuous and categorical features.
- The decision stumps minimize the sum of the false-classification rates for normal and attack samples. It is guaranteed that the misclassification rates for the selected weak classifiers are lower than 50.

3.2 Semi-supervised Algorithm

The algorithm for Semi-supervised approach is given as:

1. Train the system with supervised approach using only label data from the mixed data.
2. Give unlabelled data from mixed data for testing.
3. If the confidence of data is above the threshold value then add data with label into the training data set.
4. Train the system with this new data.

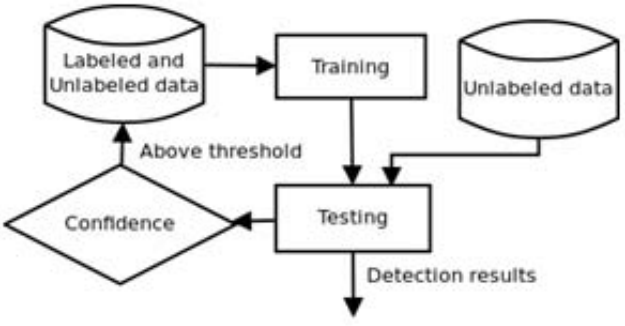

Fig. 2. Architecture for semi-supervised IDS

3.3 Unsupervised Algorithm

Heirarchical SOM have been proposed and implemented in our previous work [3].Specific attention is given to the hierarchical development of abstractions, which is sufficient to permit direct labeling of SOM nodes with connection type. Hierarchical SOM for intrusion detection use the classification capability of the SOM on selected dimensions of the data set to detect anomalies. Their results are among the best known for intrusion detection.

4 Results

We utilize the KDD CUP 1999 data set [17] for our experiments.There are four general types of attacks appeared in the data set: DOS (denial of service), U2R (user to root), R2L (remote to local) and PROBE. In each of the four, there are many low level types of attacks. Detailed descriptions about the four general types can be found in [31]. The number of samples of various types in the testing data set is listed in Table 1.

Table 1. Performance of supervised algorithm in Testing Data Set

	Normal	**DOS**	**R2L**	**U2R**	**PROBE**	**%**
Normal	**97218**	19	9	0	32	**99.93**
DOS	20	**391413**	3	4	18	**99.98**
R2L	15	0	**1102**	4	5	**98.04**
U2R	5	0	0	**45**	2	**88.46**
PROBE	40	11	9	0	**4047**	**98.53**

First, we run the classical Adaboost algorithm, whose result is shown in Table 2.

The data set for testing semi-supervised approach contains 11000 labeled data out of which 10000 are considered as unlabeled. Now, we run the semi-supervised algorithm on testing data set, whose result is shown in Table 5.

Table 6 gives the detection rate and false alarm rate for both the approaches.

Table 2. Performance of supervised algorithm in Testing Data Set

Normal	**DOS**	**R2L**	**U2R**	**PROBE**
97218	19	9	0	32
20	**391413**	3	4	18
15	0	**1102**	4	5
5	0	0	**45**	2
40	11	9	0	**4047**

Table 3. Testing Data Set for semi-supervised approach

Labeled data	Unlabeled data	Total data
1000	10000	11000

Table 4. Number of samples in testing data for semi-supervised

Normal	Attack				Total
	DOS	U2R	R2L	PROBE	
	7392	86	446	137	
1939	8061				10000

Table 5. Performance of Semi-supervised algorithm in Testing Data Set

	Normal	**DOS**	**U2R**	**R2L**	**PROBE**	**%**
Normal	**1884**	22	2	11	20	**97.16**
DOS	159	**7033**	45	106	49	**95.15**
U2R	7	24	**48**	2	5	**98.23**
R2L	28	353	23	**22**	20	**98.23**
PROBE	11	19	2	1	**104**	**97.07**

Table 6. Number of Samples in Data Set for Un-supervised approach

Normal	**Attack**				**Total**
	DOS	***U2R***	***R2L***	***PROBE***	
386	162	54	118	132	**852**
	466				

Table 7. Number of Samples in Performance of Un-supervised approach

	Normal	**DOS**	**U2R**	**R2L**	**PROBE**	**%**
Normal	380	3	1	0	2	**98.44**
DOS	1	159	0	0	1	**98.14**
U2R	3	2	0	48	1	**88.88**
R2L	2	1	114	0	0	**96.61**
PROBE	3	2	1	1	125	**94.69**

Table 8. Detection Results in Testing Data Set

Approach	**Testing Set**	
	FPR(%)	**DR(%)**
Supervised	0.06	**99.7**
Semi-supervised	0.028	**96.90**
Unsupervised	1.57	**95.35**

5 Conclusion

In the last twenty years, Intrusion Detection Systems have slowly evolved from host and operating system specific application to distributed systems that involve a wide array of operating system. The challenges that lie ahead for the next generation of Intrusion Detection Systems are many. Traditional Intrusion Systems have not adapted adequately to new networking paradigms like wireless and mobile networks. Factors like noise in the audit data, constantly changing traffic profiles and the large amount of network traffic make it difficult to build a normal traffic profile of a network for the purpose intrusion detection.

A perennial problem that prevents widespread deployment of IDS is their inability to suppress false alarms. Therefore, the primary and probably the most important challenge that needs to be met is the development of effective strategies to reduce the high rate of false alarms.

The experimental results show that the proposed algorithms have very low false alarm rate for training and testing. The semi-supervised algorithm shows better results for training and testing. The proposed algorithms have a competitive performance as compared with the published intrusion detection algorithms on the benchmark sample data.

References

1. Patole, V.A., Pachghare, V.K., Kulkarni, P.: AdaBoost Algorithm to Build Pattern Based Network Security. International Journal of Information Processing 5(1), 57–63 (2011)
2. Pachghare, V.K., Kulkarni, P.: Performance Analysis of Pattern Based Network Security. In: 2nd International Conference on Computer Technology and Development (ICCTD 2010), pp. 277–281. IEEE (2010)
3. Pachghare, V.K., Patole, V., Kulkarni, P.: Self Organizing Maps to Build Intrusion Detection System. International Journal of Computer Applications 1(8) (February 2010)
4. Song, E., Huang, D., Maa, G., Hung, C.-C.: Semi- supervised multi-class Adaboost by exploiting unlabeled data. Journal of Expert Systems with Applications (2010)
5. Pachghare, V.K., Kulkarni, P., Nikam, D.: Overview of Intrusion Detection Systems. International Journal of Computer Science and Engineering Systems 3(3), 265–268 (2009)
6. Wei, X., Huang, H., Tian, S.: Network Anomaly Detection Based on Semi-supervised Clustering. In: Proceedings of the 7th WSEAS International Conference on Simulation, Modelling and Optimization, Beijing, China, September 15-17 (2007)
7. Ermany, J., Mahantiy, A., Arlittyz, M., Cohenz, I., Williamsony, C.: Semi-Supervised Network Traffic Classification. In: SIGMETRICS 2007, San Diego, California, USA, June 12-16. ACM (2007)
8. Nigam, K., McCallum, A., Mitchell, T.: Semi-supervised Text Classification Using EM. In: In Chapelle, O., Zien, A., Cholkopf, B. (eds.) Semi-Supervised Learning. MIT Press, Boston (2006)
9. Jiang, S., Song, X., Wang, H., Han, J., Li, Q.: A clustering-based method for unsupervised intrusion detections. Pattern Recognit. Lett. 27(7), 802–810 (2006)
10. Mukkamala, S., Sung, A.H., Abraham, A.: Intrusion detection using an ensemble of intelligent paradigms. Network and Computer Applications 28(2), 167–182 (2005)
11. Zhu, X.: Semi-supervised Learning Literature Survey. Technical Report 1530, Department of Computer Sciences, University of Wisconsin, Madison (2005)
12. Hong, P., Zhang, D., Wu, T.: An intrusion detection method based on rough set and svm algorithm. In: Proceedings of International Conference on Communications, Circuits and Systems, vol. 2, pp. 1127–1130 (June 2004)
13. Rudin, C., Daubechies, I., Schapire, R.E.: The Dynamics of Adaboost: Cyclic Behavior and Convergence of Margins. Journal of Machine Learning (5), 1557–1595 (2004)

14. Mukkamala, S., Sung, A.H.: A comparative study of techniques for intrusion detection. In: Proc. Int. Conf. Tools Artif. Intell., pp. 570–577 (2003)
15. Liu, Y.H., Tian, D.X., Wang, A.M.: Annids: Intrusion Detection System Based on Artificial Neural Network. In: Proceedings of International Conference on Machine Learning and Cybernetics, vol. 3, pp. 1337–1342 (November 2003)
16. Mukkamala, S., Janoski, G., Sung, A.H.: Intrusion detection using neural networks and support vector machines. In: Proc. Int. Joint Conf. Neural Network, vol. 2, pp. 1702–1707 (2002)
17. Stolfo, S., et aI.: The third international knowledge discovery and data mining tools competition (2002), http://kdd.ics.uci.eduidatabases/kddCup99/kddCup99.html
18. Bace, R., Mell, P.: NIST Special Publication on Intrusion Detection Systems, August 16 (2001)
19. Haykin, S.: Neural Networks: A Comprehensive Foundation, 2nd edn. Prentice Hall (1999)
20. Denning, D.: An Intrusion-Detection Model. IEEE Transactions on Software Engineering SE-13(2) (February 1987)
21. Zhang, Z., Shen, H.: Application of online- training SVMs for real-time intrusion detection with different considerations. Journal Computer Communications 28(12) (July 2005)
22. Guan, Y., Ghorbani, A.A., Belacel, N.: Y-Mean: A Clustering method For Intrusion Detection. In: ICCECE 2003, pp. 1–4., www.jatit.org/volumes/researchpapers/Vol4No9/5Vol4No9.pdf
23. Guo, H.-X., Zhu, K.-J., Gao, S.-W., Liu, T.: An Improved Genetic k-means Algorithm for Optimal Clustering. In: Sixth IEEE International Conference on Data Mining Workshops, pp. 793–797 (2006)