Clustering-based Feature Selection for Internet Attack Defense

Jungtaek Seo[1], Jungtae Kim[2], Jongsub Moon[3],
Boo Jung Kang[4], and Eul Gyu Im[4]

[1] Attached Institute of ETRI
Daejeon, Korea
seojt@ensec.re.kr
[2] Secuve Inc.
Seoul, Korea
[3] Korea University, Seoul, Korea
jsmoon@korea.ac.kr
[4] College of Information and Communications
Hanyang University, Seoul, 133-791
Republic of Korea
imeg@hanyang.ac.kr

Abstract. Feature selection is as important for intrusion detection as it is for many other problems. A feature selection algorithm can help system administrators to identify and detect new network attacks efficiently since appropriately chosen features can improve accuracy of intrusion detection significantly as well as can decrease computational overheads of intrusion detection systems.

This paper describes a new proposed feature selection algorithms in detecting intrusions using network audit trails. The proposed method is based on our definition of cluster distance to select good features, and advantages of the proposed feature selection method include independence of data formats (e.g., continuous data or discrete data), suitability for binary classification, and improved intrusion detection accuracy. Experimental results using KDDCup99 datasets show that the proposed model can improve intrusion detection accuracy, compared to other algorithms.

Keywords: Feature selection, intrusion detection, network security

1 Introduction

Over the last several years, there have been many researches on the defense mechanisms against network attacks [1, 2], especially on intrusion detection. However, the most common shortcoming of the intrusion detection systems is that the mechanisms need predefined detection rules or signatures. There are many drawbacks of signature-based attack detection mechanisms: burdens of signature generation, delayed counter-measures against attacks, conflicts among signatures, and so on. To tackle these problems, there have been many proposed approaches, such as machine learning based approaches, data mining based approaches, and statistical data based approaches, to generate attack signatures automatically. However, these approaches are known to have too many false positives and cause performance overheads to generate signatures automatically. To reduce the false positive rate and to improve performance, feature selection is one of most critical processes in generating attack signatures [3–6].

The support vector machines (SVMs) are most widely used algorithm for binary feature classification and selection [7–10]. SVMs plot the training vectors in a feature space, labeling each vector by its class, and classify input data into a class by determining a set of support vectors that outline a hyperplane in the feature space. The training algorithms of SVMs try to find the optimal separating hyperplane by maximizing the margin between the hyperplane and the data and thus minimizing the upper bound of the generalization error. More detailed information about SVMs can be found in [7, 11]. SVM uses features to determine hyperplanes through support vectors. Therefore, a feature selection algorithm can affect accuracy as well as performance of SVMs [12].

Feature selection is a process that selects a subset of features from input data, such as network traffic, to reduce overheads of data processing and to improve the accuracy of attack detection. A set of features are used to compare input data with known signatures to detect attacks, and it is important to select a correct and optimal subset of features. In this paper, we propose a new feature selection method based on data clustering. The experimental results show that our proposed method can improve the accuracy of attack detection comparing with others' methods.

The advantages of the proposed feature selection method are as follows: First, it is independent of data formats (e.g., continuous data or discrete data). Second, it is suitable for binary classification that is applicable to various machine learning algorithms.

The rest of this paper is organized as follows: Section 2 illustrates related work and their pros and cons. Our proposed feature selection method is described in Section 3, and experimental results are shown in Section 4, followed by conclusions in Section 5.

2 Related Work

There have been many feature selection algorithms for machine learning applications [13, 14], and selection algorithms can be divided into two categories: clustering-based algorithms and classification-based algorithms.

Clustering-based feature selection algorithms were proposed by many people, especially using correlation criteria. There are many correlation criteria, such as the Fisher's criterion, the T-test criterion, and other similar criteria [13, 14]. A major limitation of correlation criteria is that it can only detect linear dependencies between features and targets. So, issues of clustering-based feature selection algorithms are how to reduce false positives and to increase performance of attack detection using the selected features.

In classification based algorithms, one can select features according to their individual characteristics. For example, the value itself of the feature can be used as a determining factor. A classifier is obtained by setting a threshold on the value of the feature (e.g., at the mid-point between the centers of gravity of the two classes). People need to select a subset of features that can classify input data clearly. However, in case that there is a large number of features, it may be hard to select an optimal subset of features by ranks based on classification success rate.

Several feature selection mechanisms using information theoretic criteria have also been proposed. Most of these mechanisms rely on empirical estimates of the dependency between each feature and a target data [15, 16]. The estimation obviously becomes harder with larger numbers of classes and feature values because all possible dependency among features must be considered to calculate estimates.

3 Our Proposed Model: Clustering-based Feature Selection

There are many potential benefits of feature selection for intrusion detection systems: reduction of performance and storage overheads, reduction of duration and data amounts of IDS training, and reduction of signature generation time [17]. With these reasons, the feature selection is one of most important parts in attack detection.

Among the large number of features that can be monitored for attack detection, there should be an algorithm or mechanism to identify most appropriate subset of features for a certain attack since different attacks need different subsets to have better detection accuracy. In this paper, we propose a new feature selection method based on cluster distance. In the following sections, cluster distance is addressed first, then our proposed feature selection mechanism is explained.

3.1 Our Definition of Cluster Distance

A cluster distance is used to distinguish two different clusters. If the distance between a cluster and a pre-defined attack cluster is long, it means that the distinction between two clusters is clear. On the other hand, if the distance is short, it means that the distinction between two clusters is not clear. In the support vector machines (SVMs) model, as the margin (distance between *support vectors*) becomes bigger, the classification accuracy becomes better in the trained machine. Thus, if cluster distance is bigger, the margin is bigger and the accuracy of trained machine is better.

To calculate the cluster distance, several parameters can be used. Firstly, the distance between the closest two points can be used as a parameter. Based on this parameter, the cluster distance can be defined as follows:

$$\begin{array}{l} D_{AB} = k(\boldsymbol{x}_a - \boldsymbol{x}_b) = k\sqrt{\sum_{i=1}^{n}(x_{a_i} - x_{b_i})^2} \\ x_{a_i} \in A_i, x_{b_i} \in B_i \\ m_{a_i} - 2.32\sigma_A \leq x_{a_i} \leq m_{a_i} + 2.32\sigma_A \\ m_{b_i} - 2.32\sigma_B \leq x_{b_i} \leq m_{b_i} + 2.32\sigma_B \end{array} \tag{1}$$

where D_{AB} is a cluster distance, x_{a_i} and x_{b_i} are instances that are closest to the other cluster of the i^{th} feature, A_i is an instance of an attack cluster of the i^{th} feature, B_i is an instance of a normal cluster of the i^{th} feature, m_{a_i} and m_{b_i} are means of clusters A and B, σ_A and σ_B are standard deviations of clusters A and B, k is a proportional constant, and n is the number of features.

Based on the "Cumulative Standardized Normal Distribution Function" table in [18], we found that 99% of the instances of a cluster stay within the circle with a radius of 2.32σ, where σ is the standard deviation *of the data.* Therefore, we choose the boundary of cluster as a circle with a radius of 2.32σ and select a closest instance within the boundary. The area within the circle is called the Confident Area of a cluster.

In addition, cluster distance is proportional to the distance between means of two clusters as shown in the formula (2).

$$D_{AB} = g(\boldsymbol{m}_a - \boldsymbol{m}_b) = g\sqrt{\sum_{i=1}^{n}(m_{a_i} - m_{b_i})^2} \tag{2}$$

where g is a proportional constant and m_a and m_b are means of clusters A and B.

On the other hand, a cluster distance is in inverse proportion to the standard deviations of clusters. In other words, if the distance between instances a and b that are closest instance to the other cluster, the feature that have small standard deviation would show good performance when we train the machine.

Formula (3) shows a relationship between each cluster's standard deviation and cluster distance.

$$D_{AB} = l \times \frac{1}{\sigma_A \times \sigma_B} \tag{3}$$

where l is a proportional constant.

By combining the formulas (1), (2), and (3), we defined the cluster distance as follows:

$$\begin{aligned} D_{AB} = k' \times & \sqrt{\sum_{i=1}^{n}(x_{a_i} - x_{b_i})^2} \\ & \times \sqrt{\sum_{i=1}^{n}(m_{a_i} - m_{b_i})^2} \times \frac{1}{\sigma_A \times \sigma_B} \end{aligned} \tag{4}$$

3.2 Feature Selection using Cluster Distances

In this section, we will address two strategies for the feature selection algorithms are most widely used to select an appropriate subset of features, and then our proposed selection algorithm will be introduced.

The first strategy is based on a threshold of cluster distance. The algorithm selects all features that have bigger cluster distance values than a certain threshold. One of problems with this approach is that there is no deterministic method to decide a threshold. As a threshold value increases, the number of selected features decreases, and vice versa.

The second one is based on the frequencies of cluster distance occurrences. For example, features are sorted by the number of occurrences of cluster distances, and then features with top N high frequencies are selected. A problem with this approach is that features with very small cluster distance might be selected. For example, if cluster distances of all features are very small, we have to choose features with small cluster distances.

In this paper, we employ a hybrid approach of the above two strategies. Our algorithm selects features with top N high frequencies among features that have larger cluster distance that a certain threshold. To calculate cluster distance, our algorithm uses the definition of cluster distance described in the previous section.

The procedure of our algorithm is as follows:

1. Select an input feature
2. Calculate the cluster distance of the input feature
3. Repeat steps 1 to 2 for all the input features
4. Group the input features according to their cluster distances
5. Select top N features with larger cluster distance than a certain threshold.

Using the above feature selection procedure, we select different feature sets for different attack types. From the experimental results, we have found that every attack has its own feature set that represents its attack more clearly. For example, a high SYN flag rate clearly represents the SYN Flooding attack, but it cannot clearly represent the UDP Flooding attack because UDP does not use the SYN flag. Therefore, using the different feature sets for different attacks, we can enhance the attack detection accuracy for input data of intrusion detection systems.

4 Experimental Results

In this section we show the experimental results of our proposed model applied to various DDoS attacks. To evaluate the proposed model, we applied our model to detect DDoS attacks and measured three performance indicators; a false positive rate, a false negative rate, and an accuracy. We test the proposed feature selection model with two data sets; the KDDCup99 data set and real world data set collected in our institute. We compared experimental results of the proposed model with the model proposed by S. Mukkamala et al. in New Mexico University [9, 8, 17].

We defined 41 features from the KDDCup99 data set, and calculated cluster distance of each attack (e.g. Satan, Neptune, or Smurf attacks) using our proposed method. On the other hand, in the feature selection method proposed by S. Mukkamala et al. [10], 41 features are ranked into 3 types for DDoS attacks : *important, secondary*, or *Unimportant*.

To evaluate our proposed feature selection method, we trained the machine with the proposed method and calculated the false positives and false negative rates. In the experiments, we used two kernel functions (Linear Kernel and Polynomial Kernel) of the Support Vector Machine (SVM) [9, 8]. Fig. 1 and Fig. 2 show the attack detection accuracies for each kernel. In the experiments, we used four DDoS attacks in the DARPA data sets: the Satan attack, the Smurf attack, the Neptune attack, and the *Back* attack. In Fig. 1 and Fig. 2, accuracy is calculated as a true positive rate plus a true negative rate. As you can see, the features selected with our model show higher accuracy than the model proposed by S. Mukkamala et al. In the experiment with polynomial kernel, we used various training data, but it shows approximately same attack detection accuracies. Even though the size of the training data is small, the proposed model shows approximately same detection rate with a trained machine with large training data. On the other hand, the model proposed by S. Mukkamala et al. shows lower attack detection accuracy when it uses small-sized training data.

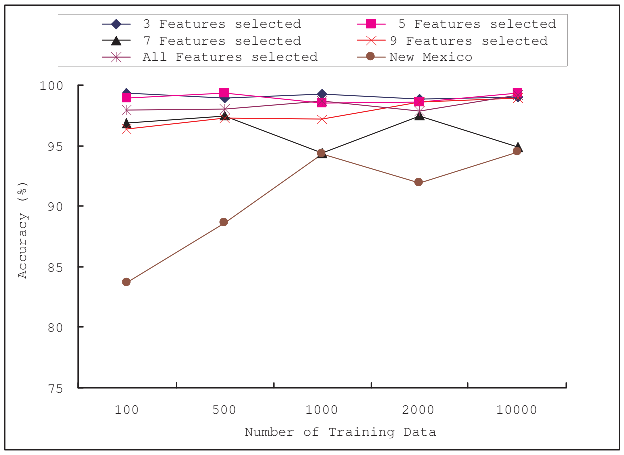

Fig. 1. Accuracy and Error Rate of Linear Kernel

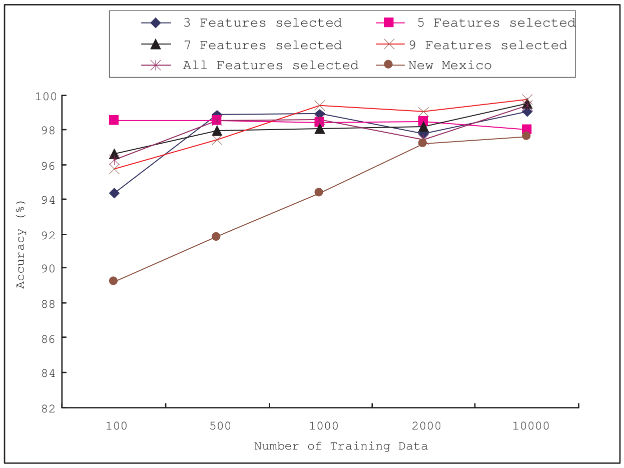

Fig. 2. Accuracy and Error Rate of Polynomial Kernel

5 Conclusions

There have been many researches on defense mechanisms against network attacks such as DDoS attacks. But it is almost impossible to analyze, create, and test an enormous number of signatures for new attacks in timely manners. There have been research efforts, such as machine learning, data mining, and statistical algorithms, to generate attack signatures automatically. However, these approaches generally have a high false positive rate when generating automatic signatures.

In this paper, we proposed a new feature selection method based on clustering. Through the experiments, we show that the attack detection rate of the machine learned by the small feature set selected by the proposed model is approximately same with that of the machine learned with all features. Our proposed feature selection method can be used to improve performance and accuracy of intrusion detection systems.

References

1. Gil, T., Poletto, M.: MULTOPS: a data-structure for bandwidth attack detection. In: Proceedings of the 10th USENIX Security Symposium. (2001) 23–38
2. Wang, H., Zhang, D., Shin, K.G.: Detecting SYN flooding attacks. In: Proceedings of the IEEE Infocom 2002, New York City, NY (2002)
3. Liu, H., Yu, L.: Toward integrating feature selection algorithms for classification and clustering. IEEE Transactions on Knowledge and Data Engineering **17**(4) (2005) 491–502

4. Heller, K., Svore, K., Keromytis, A., Stolfo, S.: One class support vector machines for detecting anomalous windows registry accesses. In: Proceedings of the workshop on Data Mining for Computer Security. (2003)
5. D. Anderson, e.a.: Detecting unusual program behavior using the statistical component of the next-generation intrusion detection. Technical Report SRI-CSL-95-06, Computer Science Laboratory, SRI International, Menlo Park, CA. USA (1995)
6. Jobo B. D. Cabrera, e.a.: Statistical traffic modeling for net work intrusion detection. In: Proceedings of the 8th International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems, San Francisco, CA. USA (2000)
7. Vapnik, V.: The Nature of Statistical Learning Theory. Springer-Verlag, NewYork (1995)
8. Sung, A.H.: Identifying important features for intrusion detection using support vector machines and neural networks. In: Proceedings of the SAINT 2003. (2003) 209–217
9. Mukkamala, S., Sung, A.H.: Identifying key features for intrusion detection using neural networks. In: Proceedings of the International Conference on Computer Communications 2002. (2002)
10. Mukkamala, S.: Intrusion detection: Support vector machines and neural networks. In: Proceedings of the IEEE international joint conference on Neural Networks. (2002)
11. Wu, K.P., Wang, S.D.: Choosing the kernel parameters of support vector machines according to the inter-cluster distance. In: Proceedings of the 2006 International Joint Conference on Neural Networks. (2006)
12. Joachims, T.: Estimating the generalization performance of an svm efficiently. In: Proceedings of the Seventeenth International Conference on Machine Learning. (2000) 431–438
13. Tusher, V.G., Tibshirani, R., Chu, G.: Significance analysis of microarrays applied to the ionizing radiation response. In: Proceedings of the PNAS. (2001) 116–121
14. Hastie, T., Tibshirani, R., Friedman, J.: The Elements of Statistical Learning. Springer, New York (2001)
15. Dhillon, I., Mallela, S., Kumar, R.: A divisive information-theoretic feature clustering algorithm for text classification. Journal of Machine Learning Research **3** (2003) 1265–1287
16. Torkkola, K.: Feature extraction by non-parametric mutual information maximization. Journal of Machine Learning Research **3** (2003) 1415–1438
17. Tamilarasan, A., Mukkamala, S., Sung, A., Yendrapalli, K.: Feature ranking and selection for intrusion detection using artificial neural networks and statistical methods. In: Proceedings of the International Joint Conference on Neural Networks, 2006. IJCNN '06. (2006) 4754– 4761
18. Brownlee, K.: Statistical Theory and Methodology. John Wiley and Sons, Inc. (1967)
19. Weston, J., Elisseeff, A., Scholkopf, B.: Use of the zero norm with linear models and kernel methods. Journal of Machine Learning Research **3**(0) (2003) 1439–1461
20. Forman, G.: An extensive empirical study of feature selection metrics for text classification. Journal of Machine Learning Research **3**(0) (2003) 1289–1306