Intrusion detection algorithm based on clustering: Kernel-ART

Hansung Lee*, Younghee Im, Jooyoung Park***, and Daihee Park***

*Computer Science, Korea University

**Computer and Communications Engineering, Daejeon University

***Control and Instrumentation Engineering, Korea University

E-mail : mohan@korea.ac.kr

ABSTRACT

In this paper, we propose a new intrusion detection algorithm based on clustering: Kernel-ART, which is composed of the on-line clustering algorithm, ART (adaptive resonance theory), combining with mercer-kernel and concept vector. Kernel-ART is not only satisfying all desirable characteristics in the context of clustering-based IDS but also alleviating drawbacks associated with the supervised learning IDS. It is able to detect various types of intrusions in real-time by means of generating clusters incrementally.

Keyword : intrusion detection, ART, mercer kernel, concept vector.

1. INTRODUCTION

In the traditional signature-based intrusion detection system (IDS), the rule base, determined by the human expert according to his or her experience, plays an important role, but may be difficult to extract optimally from the expert, particularly as the system increases in complexity. Moreover, the rule-base has to be manually revised whenever each new type of attack is discovered [1][4].

Recently, some of machine learning algorithms commenced applying to the intrusion detection system which circumvents some drawbacks of the signature-based system [1][2][3][4]. However, as far as the machine learning algorithms are concerned, most of researches are focused on a supervised learning in the context of IDS. Consequently, certain challenging problems still remain open, including: 1) expensive cost of training; 2) dependency of the data quality; 3) scalability and incrementality; 4) difficulty of detecting new intrusions which are not trained. Accordingly, it is exploited by the clustering algorithm (i.e., unsupervised learning) leading to an even better performance of IDS [1][3][4].

In general, the desirable characteristics emphasized in the performance evaluation of IDS concerning the clustering algorithm include the following:

1) The method should start to process each event data as soon as it is received and generate clusters adaptively without fixing the number of clusters.
2) The method ought to be able to cluster huge volume of event data in few seconds.
3) The result of clustering is insensitive to the order of input data, since the sequence of event data is arbitrary in general.

In this paper, our primary goal is to design a clustering-based intrusion detection algorithm which is not only satisfying all desirable characteristics in the context of clustering-based IDS but also alleviating drawbacks associated with the supervised learning IDS. Our algorithm, the so-called Kernel-ART, is composed of the on-line clustering algorithm, ART (adaptive resonance theory), combining with mercer-kernel [7] and concept vector [10]. It is shown in this paper that Kernel-ART outperforms both in classifying high dimensional sparse patterns and in separating clusters, in particular.

2. Data representation and Similarity measure

For a given set of n input patterns $X = \{\underline{x}_i\}_{i=1}^{n}$, we assume that the input pattern $\underline{x}_i$ consists of k numeric attributes and m symbolic attributes.

$$\underline{x}_i = \underline{x}_i^R + \underline{x}_i^S \ ; \ \underline{x}_i^R \in R^k, \underline{x}_i^S \in S^m \qquad (1)$$

where R^k is k-dimensional numeric space; S^m is m-dimensional symbolic space.

To avoid bias toward some features over other features, we perform L2 normalization on numeric attributes to have unit Euclidean norm.

$$\underline{x}_i^R = \frac{\underline{x}_i^R}{\|\underline{x}_i^R\|} \ ; \ \|\underline{x}_i\| = \sqrt{\sum_{j=1}^{k} x_{ij}^2} \qquad (2)$$

We present similarity measure which computes the similarity between objects of mixed variable type.

$$S(\underline{x}_i, \underline{x}_j) = \lambda \cdot <\underline{x}_i^R, \underline{x}_j^R> + (1-\lambda) \cdot \frac{\sum_{l=1}^{m} \delta(x_{il}^S, x_{jl}^S)}{m} \qquad (3)$$

where m is the dimension of symbolic space, and adjustable parameter $\lambda \in [0,1]$ is to weight the attribute types. The delta function $\delta(\cdot)$ is defined as follows:

$$\delta(x_{il}^S, x_{jl}^S) = \begin{cases} 1, & \text{if } x_{il}^S = x_{jl}^S \\ 0, & \textit{otherwise} \end{cases} \qquad (4)$$

Since numeric attributes are normalized to be unit vector, the cosine measure is obtained by inner product of two vectors.

$$\begin{aligned} <\underline{x}_i^R, \underline{x}_j^R> &= \|\underline{x}_i^R\| \cdot \|\underline{x}_j^R\| \cdot COS(\theta(\underline{x}_i^R, \underline{x}_j^R)) \\ &= COS(\theta(\underline{x}_i^R, \underline{x}_j^R)) \end{aligned} \qquad (5)$$

3. Kernel-ART

In this section, first we describe some rudimentary aspects of concept vector and mercer-kernel, then introduce the so-called Kernel-ART which is composed of the on-line clustering algorithm, ART (adaptive resonance theory), combining with concept vector and mercer-kernel.

3.1 Concept-Vector

The concept vector is the mean vector of the cluster normalized to have unit Euclidean norm. Since the concept vectors (i.e., clusters) are localized in the high dimensional sparse space, the clusters represent the class structure of dataset [10] (e.g., the clusters represent the each types of attacks respectively).

For the c disjoint clusters, the *mean vector* $\underline{m}_j$ of the cluster π_j is defined as

$$\underline{m}_j = \underline{m}_j^R + \underline{m}_j^S \ ; \quad \underline{m}_j^R \in R^k, \underline{m}_j^S \in S^m \qquad (6)$$

where $\underline{m}_j^R = \frac{1}{n_j} \sum_{x^R \in \pi_j} x^R$ and $\underline{m}_j^S$ is defined as the most frequent symbol in cluster π_j.

The *normalized mean vector* $\underline{c}_j$ of the cluster π_j is defined as

$$\begin{aligned} \underline{c}_j &= \frac{\underline{m}_j^R}{\|\underline{m}_j^R\|} + \underline{m}_j^S = \underline{c}_j^R + \underline{m}_j^S \ ; \\ & \underline{m}_j^R \in R^k, \ \underline{m}_j^S \in S^m \end{aligned} \qquad (7)$$

where $\underline{c}_j^R$ is concept vector.

The concept vector $\underline{c}_j^R$ has the following important property. For any unit vector $\underline{z} \in R^d \geq 0$, we have the Cauchy-Schwarz inequality that $\sum_{\underline{x} \subset \pi_j} \underline{x}^T \underline{z} \leq \sum_{\underline{x} \subset \pi_j} \underline{x}^T \underline{c}_j$. Thus, the normalized mean vector may be thought of as the vector that is the closest in cosine similarity to all the input vectors in the cluster π_j [10].

3.2 Mercer-Kernel

The basic idea of mercer-kernel is to map the data into some other dot product space, called feature space, F via a nonlinear map and perform the linear algorithm in F [6][7]. By replacing the inner product in similarity measure of equation (3) with kernel function $K(\cdot)$, we obtain similarity measure function in feature space.

$$S(\underline{x}_i, \underline{x}_j) = \lambda \cdot K(\underline{x}_i^R, \underline{x}_j^R) + (1-\lambda) \cdot \frac{\sum_{l=1}^{m} \delta(x_{il}^S, x_{jl}^S)}{m} \tag{8}$$

With RBF (radial basis function) kernel, $K(\underline{x}_i, \underline{x}_j) = \exp\left\{-\frac{1}{c} \|\underline{x}_i - \underline{x}_j\|^2\right\}$, we obtain similarity measure function as follows

$$S(\underline{x}_i, \underline{x}_j) = \lambda \cdot \exp\left\{-\frac{1}{c} \|\underline{x}_i^R - \underline{x}_j^R\|^2\right\} + (1-\lambda) \cdot \frac{\sum_{l=1}^{m} \delta(x_{il}^S, x_{jl}^S)}{m} \tag{9}$$

3.3 Kernel-ART

The proposed Kernel ART is described as follows.

initialization: The number of clusters is initialized to be one and perform L2 normalization on numeric attributes. Also the first input pattern is assigned to initial weight vector.

$$\underline{w}_1 = \underline{x}_1 = \underline{w}_1^R + \underline{w}_1^S = \underline{x}_1^R + \underline{x}_1^S \tag{10}$$

The matching value, computed by activation function between initial weight vector and first input pattern, is set to be one. This ensures that the first input pattern is assigned to first cluster for any vigilance parameter $\rho \in [0,1]$.

activation function: The activation function is defined by similarity measure in the feature space.

$$AF(\underline{x}_i, \hat{\underline{w}}_j) = \lambda \cdot \exp\left\{-\frac{1}{c} \|\underline{x}_i^R - \hat{\underline{w}}_j^R\|^2\right\} + (1-\lambda) \cdot \frac{\sum_{l=1}^{m} \delta(x_{il}^S, w_{jl}^S)}{m} \tag{11}$$

where $\hat{\underline{w}}_j$ is the normalized mean vector of cluster π_j ; $\hat{\underline{w}}_j^R = \frac{\underline{w}_j^R}{\|\underline{w}_j^R\|}$ is the concept vector of cluster π_j .

matching function: If activation function and matching function are chosen such that

$$MF(\underline{w}_1, \underline{x}_i) > MF(\underline{w}_2, \underline{x}_i) \Leftrightarrow AF(\underline{w}_1, \underline{x}_i) > AF(\underline{w}_2, \underline{x}_i) \tag{12}$$

then no mismatch reset condition and search process are required to resonance domain. The most simple choice of activation function and matching function under the condition of equation (12) is to choose activation function as matching function[9].

$$AF(\underline{w}_j, \underline{x}_i) \equiv MF(\underline{w}_j, \underline{x}_i) \tag{13}$$

resonance condition: Since $AF(\underline{w}_j, \underline{x}_i) \equiv MF(\underline{w}_j, \underline{x}_i)$, the resonance unit is selected as follows

$$AF(\underline{w}_{j*}, \underline{x}_i) \geq \rho \tag{14}$$

where $j* = \arg \max_{j=1,\cdots,c} \{AF(W_j, X_i)\}$.

When best-matching template does not satisfy the vigilance criterion, a new cluster unit can be created and assign the input pattern. This condition may improve the speed of algorithm.

update weight vector: When cluster $j*$ is selected by equation (14), the input pattern is assigned to cluster $j*$ and the weight vector is updated as follows

$$\underline{w}_{j*}^{R(i)} = \underline{w}_{j*}^{R(i\ 1)} + \underline{x}_i^R$$
$$\underline{w}_{j*}^{S(i)} = Most\ frequent\ symbol \qquad (15)$$

Since the weight vector of cluster $j*$ is defined by sum of input patterns that is assigned to cluster $j*$, we need not consider learning rate parameter in update of weight vectors. The result of Kernel-ART is less sensitive than that of Fuzzy ART with respect to the order of input patterns, because weight vectors memorize the normalized mean vector of input patterns that are assigned to each clusters in Kernel-ART.

The foregoing descriptions are summarized in algorithm 1.

```
Step0. Normalize input pattern with L2 norm.
       Initialize Weights:
       $\underline{w}_1 = \underline{x}_1 = \underline{w}_1^R + \underline{w}_1^S = \underline{x}_1^R + \underline{x}_1^S$
Step1. While Stopping Condition is false, do Step 2-7
  Step2. For each training input, do Step 3-6
    Step3. Set activation of all F2 to zero
    Step4. Compute Activation Function:
           $AF(\underline{x}_i, \underline{w}_j)$
           $= \lambda \cdot \exp\left\{-\frac{1}{c}\ ||\underline{x}_i^R - \hat{\underline{w}}_j^R||^2\right\} + (1-\lambda) \cdot \frac{\sum_{l=1}^{m} \delta(x_{il}^S, w_{jl}^S)}{m}$
    Step5. Find $j*$ with max activation
    Step6. Test for reset:
           If $AF(W_{j*}, X_i) \geq \rho$ then
              $\underline{w}_{j*}^{R(i)} = \underline{w}_{j*}^{R(i\ 1)} + \underline{x}_i^R$
              $\underline{w}_{j*}^{S(i)} = Most\ frequent\ symbol$
           else new processing element
              allocation: c = c + 1
              $\underline{w}_{j*}^{R(i)} = \underline{w}_{j*}^{R(i\ 1)} + \underline{x}_i^R$
              $\underline{w}_{j*}^{S(i)} = Most\ frequent\ symbol$
  Step7. Test for stopping condition
```

Algorithm 1: Kernel-ART Algorithm

4. Experimental Results

To validate the performance of Kernel-ART, we adopt KDD CUP 1999 data [5] which consists of correct labeled 311,029 data instances. Among them, we sampled 880 data instances such as 176 normal instances, 176 DOS attacks, 176 R2L attacks, 176 U2R attacks, and 176 probing instances.

4.1 Comparisons with Other Clustering Methods

To evaluate clustering performance of Kernel-ART, we compared our method with K-means algorithm and Fuzzy ART. The condition of experiments are summarized in Table 1 and the results of experiments is shown in Table 2.

Table 1 : The condition of experiments. *c parrameter* in Kernel-ART is the parameter of RBF kernel.

K-means	# of cluster = 39, repeat 30 experiments, using min-max normalization
Fuzzy ART	$\alpha = 0.00001$, $\beta = 1.0$, varying ρ from 0.35 to 0.95
Kernel-ART	$\lambda = 0.5$, c from 0.01 to 0.1, varying ρ from 0.35 to 0.95

Table 2 : The experiments of K-means, Fuzzy- ART and Kernel-ART. Where DR denotes Detection Rate (%), FP denotes False Positive Rate(%) and FN denotes False Negative Rate(%). In the best case of Fuzzy ART : $\rho = 0.9$. In the best case of Kernel-ART, < > : $\rho = 0.9$, $c = 0.1$ and ( ) : $\rho = 0.6$, $c = 0.01$

Item / Method	Average			Best		
	DR	FP	FN	DR	FP	PN
K-means	90.62	20.45	9.37	93.89	24.43	6.10
Fuzzy ART	93.96	38.73	6.03	96.73	17.61	3.26
Kernel-ART	97.74	12.68	5.25	<93.03> (96.87)	<3.40> (19.88)	<6.98> (3.12)

These experiments show that the performance of Kernel-ART is better than other methods. Also, we can derive various results of clustering by means of adjusting parameter c in RBF function.

4.2 Comparisons with Other Researches

Wenke Lee *et al.* [4] performed their classifier on DARPA dataset which is very similar to our test dataset. On the other hand, Dr. Bernhard [5] classified the correct labeled dataset which is same as our test dataset using C5 algorithm. The comparisons are given in Table 3. Since R2L and U2R are host-based attacks, these are very similar to normal data in case of KDD CUP 99 data which are collected from network packets. The comparisons indicates that our method is not only comparable to their results in general but also outperformed in separating similar patterns, in particular.

Table 3 : Comparison of three experimental results : Wenke Lee's, Dr. Bernhard's and Our method

DR(%) Class	Wenke Lee	Dr. Bernhard	Kernel-ART
Normal	-	99.5	96.59
Dos	79.9	97.1	93.18
R2L	60.0	8.4	73.86
U2R	75.0	13.2	87.50
Probing	97.0	83.3	100

5. CONCLUSION

The contribution we made in this paper was a design of more robust and efficient intrusion detection method which is able to detect various types of intrusions in real-time by means of generating clusters incrementally.

Our algorithm, Kernel-ART, satisfying all of the desired properties: incrementality and scalability; fast speed of algorithm; insensitivity of input sequence, which are mentioned in section 1. Thus, Kernel-ART overcome most of drawbacks associated with supervised learning IDS. We empirically show that Kernel-ART outperforms in separating similar patterns.

In conclusion, Kernel-ART has several advantages from a intrusion detection perspective: computational efficiency; providing information of each intrusion types; ability in detecting new intrusions which are not trained.

REFERENCES

[1] Leonid Portnoy, “Intrusion detection with unlabeled data using clustering", *Undergraduate Thesis*, Columbia University, 2000.

[2] Jack Marin, Daniel Ragsdale, and John Shurdu, "A Hybrid Approach to the Profile Creation and Intrusion Detection", Proceedings of DARPA Information Survivability Conference and Exposition, IEEE, 2001.

[3] Nong Ye, and Xiangyang Li, "A Scalable Clustering Technique for Intrusion Signature Recognition", Proceedings of 2001 IEEE Workshop on Information Assurance and Security, 2001.

[4] Wenke Lee, Salvatore J. Stolfo, Kui W. Mok, "A Data Mining Framework for Building Intrusion Detection Models", IEEE Symposium on Security and Privacy, 1999.

[5] KDD CUP 1999 DATA, Available in http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html and http://www-cse.ucsd.edu/users/elkan/kdresults.html

[6] Nello Cristianini, John Shawe-Taylor, "An Introduction to Support Vector Machines and other kernel-based learning methods", Cambridge University PRESS, pp. 26-50, 2000.

[7] Mark Girolami, "Mercer Kernel Based Clustering in Feature Space", IEEE Transactions on Neural Networks, 2001.

[8] Jiawei Han, Micheline Kamber, "Data Mining : Concepts and Techniques", Morgan Kaufmann Publishers, pp. 345-346, 2001.

[9] A. Baraldi and E. Chang, "Simplified ART : A New Class of ART Algorithms", International Computer Science Institute, TR 98-004, 1998.

[10] I. S. Dhillon and D. S. Modha, "Concept Decomposition for Large Sparse Text Data using Clustering", Technical Report RJ 10147(95022), IBM Almaden Research Center, 1999.