# Improved Clustering for Intrusion Detection by Principal Component Analysis with Effective Noise Reduction

Lu Zhao, Ho-Seok Kang, and Sung-Ryul Kim

Division of Internet and Multimedia Engineering, Konkuk University, Seoul, Korea
{ais.zhaolu,hsriver}@gmail.com, kimsr@konkuk.ac.kr

**Abstract.** PCA (Principal Component Analysis) is one of the most wildly used dimension reduction technique, which is often applied to identify patterns in complex data of high dimension [1]. In GA-KM [2], we have proposed GA-KM algorithm and have experimented using KDD-99 data set. The result showed GA-KM is efficient for intrusion detection. However, due to the hugeness of the data set, the experiment needs to take a long time to finish. To solve this deficiency, we combine PCA and GA-KM in this paper. The goal of PCA is to remove unimportant information like the noise in data sets which have high dimension, and retain the variation present in the original dataset as much as possible. The experimental results show that, compared to GA-KM [2], the proposed method is better in computational expense and time (through dimension reduction) and is also better in intrusion detection ratios (through noise reduction).

**Keywords:** Intrusion detection, Principle Component Analysis (PCA), effective noise reduction, GA-KM.

## 1 Introduction

With rapid growth of network-based services, network security is becoming more and more important than ever before. Therefore, intrusion detection system (IDS) [3] plays a vital role in network security. There are two main categories of intrusion detection techniques: signature-based detection and anomaly-based detection. Signature-based detection is also called misuse detection which is based on signatures for known attacks. Anomaly-based detection is different from signature-based detection, which is able to detect unknown attacks by learning the behavior of normal activity. In the training phase of this approach, IDS builds a profile which represents normal behavior. In the detection phase, the similarity of a new behavior with the profile is analyzed by IDS. If the new behavior is far from normal behavior of the profile, then this behavior will be labeled as an attack. We have proposed GA-KM algorithm [2], and have experimented with this algorithm using KDD--99 data set, the results show that it is efficient for anomaly-based intrusion detection [4]. However, when we experiment with this algorithm, it takes a long time to finish.

In this paper, to solve this deficiency, Principal Component Analysis is combined with GA-KM algorithm and is experimented on KDD-99 data set. PCA (Principal

K. Mustofa et al. (Eds.): ICT-EurAsia 2013, LNCS 7804, pp. 490–495, 2013.

Component Analysis) can be used to reduce the dimension of high dimensional data and remove noise from it effectively. So we can use PCA algorithm to do some noise reduction on KDD-99 data. The experimental results show that the propose method reduces the training time and testing time greatly, and also is efficient for intrusion detection.

The rest of this paper is organized as follow: in section 2, we will introduce previous works. Section 3 describes proposed method. In section 4, we report our experimental results and evaluations. Finally, in section 5, we conclude this paper.

## 2 Related Work

### 2.1 GA-KM Algorithm

GA-KM algorithm [2] combines genetic algorithm into the traditional K-means clustering algorithm [5]. In GA-KM algorithm, each individual consists of a certain number of cluster centers. The value of each cluster center is called gene of individuals. And a fitness function is defined as follows:

$$f(x) = 1/(1 + J_c) \tag{1}$$

where $J_c$ is the object function for K-means algorithm. We use this fitness function to evaluate each individual in a population. It means that if $J_c$ is the small, the fitness value is the greater, and the clustering result is better.

In GA-KM, selection, crossover and mutation operators are a little different from original GA.

- Selection

We combine elitism selection and fitness proportionate selection [6] to increase the performance of GA-KM. Elitism selection first retains the best individual which has the best fitness value in current population, and then replaces the worst individual in new population with the best individual which is retained in current population.

- Crossover

To obtain better individuals and increase convergence speed of GA-KM, good individuals which have big fitness values cross over their genes among themselves and bad individuals which have small fitness values cross over their genes in arithmetic crossover method among themselves.

The selections of genes depend on two cluster centers' distance for two individuals, one cluster center for one individual. We first calculate the distance of all pairs of cluster centers each from two individuals, and then match up cluster centers based on their minimum distance and cross over the values of these two cluster centers.

In the formula (2), $A'_1, B'_1$ are genes of two good individuals that their cluster centers' distance is the minimum distance, $A'$, $B'$ are new genes of two individuals after crossover, and $r$ is a random variable in the range [0,1].

$$A' = r * A'_1 + (1 - r) * B'_1$$
$$B' = (1 - r) * A'_1 + r * B'_1 \quad (2)$$

- Mutation

In this stage, the genes of bad individuals could have bigger chance of mutation than the genes of good individuals. So, in order to prevent early mature convergence and generate newer mutation of individuals, the genes of good individuals are modified based on original genes with a small value $Min$ , and the genes of bad individuals are modified on original genes with a big value $Max$ .

### 2.2 Principal Component Analysis

PCA (Principal Component Analysis Algorithm) known as dimension reduction technique is a useful method for identifying patterns in complex data of high dimensions, and expressing the data in such a way as to highlight their similarities and differences [7]. And this method can effectively identify data "main" elements and structure, remove noise and redundancy, and reveal the simple structure hidden in complex data.

## 3 Our Proposed Method

### 3.1 Data Description

For our experiments we use KDD-Cup 1999 dataset [8]. KDD-Cup 1999 data set contains a wide variety of intrusions simulated in a military network environment which is used for the Third International Knowledge Discovery and Data Mining Tools Competition. Each example in the data is a record of extracted features from a network connection gathered during the simulated intrusions. A connection is a sequence of TCP packets which data flows to and from a source IP address to a target IP address under some well-defined protocol. Each connection record consists of 41 fields which contain basic features about TCP connection as duration, protocol type, number of bytes transferred, domain specific features as number of file creation, number of failed login attempts, and whether root shell was obtained. Each connection is labeled as either normal, or an attack, with exactly one specific attack type.

### 3.2 Using PCA Algorithm

In this paper, the goal of PCA algorithm is to reduce unimportant information of data set which have high dimension and retain the variation present in the original dataset as much as possible. Using PCA algorithm to reduce the dimensionality of complex data can decrease computing time when we use huge data.

We use PCA algorithm to do effective noise reduction for data set $S$ which are preprocessed by mapped. Here, data set $S$ are represented as a $D \times N$ matrix, in the matrix, each column represents one data example of dataset; N is the number of data examples, D is the number of dimensionality for each connection record.

Firstly, calculate the mean value of all data examples:

$$y_{if} = \frac{1}{N} \sum_{i=1}^{N} s_{if} \tag{3}$$

Secondly, subtract mean vector for each data example:

$$y'_{if} = s_{if} - y_{if} \tag{4}$$

Thirdly, find covariance matrix $\sum$ of data set $S$, and then calculate all eigenvectors and eigenvalues of $\sum$;

Next, select number $d(d \le D)$ biggest eigenvectors based on eigenvalues to get robust representation of data, eigenvectors are represented as $u_1, u_2, \ldots, u_d$, and then find its matrix transpose $U$, represented as follows: $(d \times D \text{ matrix})$;

$$U = \begin{pmatrix} u_1^T \\ u_2^T \\ \vdots \\ u_d^T \end{pmatrix} = \begin{pmatrix} u_{11} & u_{12} & \cdots & u_{1D} \\ u_{21} & u_{22} & \cdots & u_{2D} \\ \vdots & \vdots & \vdots & \vdots \\ u_{d1} & u_{d2} & \cdots & u_{dD} \end{pmatrix} \qquad y' = \begin{pmatrix} y'_{11} & y'_{21} & \cdots & y'_{N1} \\ y'_{12} & y'_{22} & \cdots & y'_{N2} \\ \vdots & \vdots & \vdots & \vdots \\ y'_{1D} & y'_{2D} & \cdots & y'_{ND} \end{pmatrix}$$

And then we can get the new data set $x = (Uy')^T$ preprocessed by mapping and PCA. In the matrix $x$, each row represents a data example, and the number of column is the same with the number of data examples.

## 4 Experiment Results and Evaluations

In this section, we detail the experiment results of the proposed algorithm and evaluate the performance of proposed method. We experimented with this proposed method using KDD Cup 99 dataset in our paper. For training and developing normal clusters, data file "kddcup.data_10_percent" which is downloaded from KDD Cup 1999 site is used for our experiments. The training data set consists of normal data selected from the file "kddcup.data_10_percent". And the testing data set consists of normal data and attack data which are selected from the file "corrected". The main goal of our experiment is to study how the performance of proposed method.

In our experiments, we use two kinds of data sets which are original KDD CUP 99 dataset and KDD CUP 99 dataset preprocessed by PCA algorithm with effective noise reduction. In [2], we have proved that GA-KM algorithm is efficient for anomaly-based intrusion detection. Our experimental results showed that the proposed method gives better representation of data set after removing noise data, and approximately 42.5% ~ 80.7% reduction in training time. Meanwhile, we compared the performance of improved clustering method with original GA-KM.

**Table 1.** Comparison with average distance

| | PCA with GA-KM | GA-KM with scaling | GA-KM without scaling |
|---|---|---|---|
| Normal | 1145.82495 | 3.143096903 | 1219.20607 |
| Attack | 1900.862749 | 3.796500426 | 1445.27273 |
| Ratio1 | 1.66 | 1.21 | 1.19 |
| Ratio2 | | 1.37 | |

**Table 2.** Comparison with average distance with extreme cases removed

| | PCA with GA-KM | GA-KM with scaling | GA-KM without scaling |
|---|---|---|---|
| Normal | 1145.82495 | 3.143096903 | 1219.20607 |
| Attack | 1535.489161 | 3.225661856 | 1210.45372 |
| Ratio1 | 1.34 | 1.02 | 0.99 |
| Ratio2 | | 1.12 | |

**Table 3.** Comparison with median distance

| | PCA with GA-KM | GA-KM with scaling | GA-KM without scaling |
|---|---|---|---|
| Normal | 1108.653748 | 2.764393351 | 1109.26483 |
| Attack | 1359.351694 | 3.197027967 | 1213.48725 |
| Ratio1 | 1.23 | 1.15 | 1.09 |
| Ratio2 | | 1.07 | |

In the above three tables, scaling means zero-mean normalization method which is used to preprocess data set. And all distances means the minimum distance between cluster center and data. Normal represents the minimum distance between cluster center and normal data from testing data; Attack represents the minimum distance between cluster center and attack data; Ratio1 is the value which is equal to Attack divided by Normal; Ratio2 is the value which is equal to Ratio1 for PCA with GA-KM divided by Ratio1 for GA-KM with scaling. If the value Ratio1 is bigger, the difference between normal data and attack data is more evident, and detecting attack is easier. In other words, if the value Ratio2 is bigger, the performance of PCA with GA-KM is better than GA-KM with scaling.

In the experiments, we use three kinds of distances to compare the performance of proposed method and GA-KM. As shown in Table 1, we use average distance to compare PCA with GA-KM, GA-KM with scaling, GA-KM without scaling; the results show that the performance of PCA with GA-KM is 1.37 times better than GA-KM with scaling. As shown in Table 2, we remove the extreme distance and then calculate the average distance to compare these two methods, the value Ratio2 is 1.12, the result shows that the performance of proposed method is 1.12 times better than GA-KM with scaling. In Table 3, we use median distance; the result also shows that the performance of PCA with GA-KM is better than GA-KM with scaling. Also from the tables, we also can know that the performance of PCA with GA-KM, GA-KM with scaling are better than GA-KM without scaling.

From these results, we can know that performance of proposed method outperforms GA-KM. So we can know that after using PCA, the proposed method can reduce noise data effectively and retain the variation present in the original dataset as much as possible. Meanwhile, it also can retain a good performance after effective noise reduction. As a result the proposed method is also acceptable for intrusion detection.

## 5 Conclusion

Our research work is based on intrusion detection. We use KDD Cup 99 dataset to experiment and find some redundant and irrelevant data like noise data in KDD Cup 99. So to remove unimportant data effectively, we combine dimension reduction technique which is PCA algorithm with clustering method GA-KM. The goal of PCA is to reduce noise data effectively to get smaller dimensionality dataset from KDD Cup 99 dataset. Our research is to observe how Principal Component Analysis and GA-KM algorithm are used for intrusion detection. And when after the noise information of original dataset are reduced, how the performance is. The experimental results showed that the proposed method can give better and robust representation of data after effective noise reduction and have 42.5% ~ 80.7% time reduction in training, and also can retain a good performance. So, our proposed method is efficient and reliable for intrusion detection.

## References

1. Smith, L.I.: A tutorial on Principal Components Analysis. New York (2002)
2. Zhao, L., Kang, H.-S., Kim, S.-R.: K-means Clustering by Genetic Algorithm for Anomaly-based Intrusion Detection. In: International Conference on Smart Media and Applications, p. 37. Korean Institute of Smart Media (2012)
3. Intrusion detection system, http://www.sans.org/reading_room/whitepapers/detection/understanding-intrusion detection-system_337
4. Denning, D.: An Intrusion Detection Model. In: Proceedings of the Seventh IEEE Symposium on Security and Privacy, vol. SE-13, pp. 222–232 (1987)
5. Kaufan, L., Rousseeuw, P.J.: Finding groups in data: an introduction to cluster analysis. John Wiley & Sons, New York (1990)
6. Selection for genetic algorithm, http://en.wikipedia.org/wiki/Selection_(genetic_algorithm)
7. Lawrence, F.L., Sharma, S.K., Sisodia, M.S.: Network Intrusion detection by using Feature Reduction Technique. International Journal of Advanced Research in Computer Science and Electronics Engineering 1 (2012)
8. The third international knowledge discovery and data mining tools competition dataset\KDD99-Cup (1999), http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html