A clustering algorithm use SOM and K-Means in Intrusion Detection

WANG Huai-bin[1,2]
1. Tianjin Key Lab of Intelligent Computing & Novel software Technology Tianjin University of Technology, 300191, Tianjin, China
2. Key Laboratory of Computer Vision and System, Ministry of Education, 300191, Tianjin, China
E-mail: hbwang@tjut.edu.cn

*YANG Hong-liang[1,2] XU Zhi-jian[1,2] YUAN Zheng[1,2]
1. Tianjin Key Lab of Intelligent Computing & Novel software Technology Tianjin University of Technology, 300191, Tianjin, China
2. Key Laboratory of Computer Vision and System, Ministry of Education,
E-mail: 654539@163.com，xuzhijian0508@163.com
*Corresponding author

***Abstract*—Improving detection definition is a pivotal problem for intrusion detection. Many intelligent algorithms were used to improve the detection rate and reduce the false rate. Traditional SOM cannot provide the precise clustering results to us, while traditional K-Means depends on the initial value serious and it is difficult to find the center of cluster easily. Therefore, in this paper we introduce a new algorithm, first, we use SOM gained roughly clusters and center of clusters, then, using K-Means refine the clustering in the SOM stage. At last of this paper we take KDD CUP-99 dataset to test the performance of the new algorithm. The new algorithm overcomes the defects of traditional algorithms effectively. Experimental results show that the new algorithm has a good stability of efficiency and clustering accuracy**.

Keywords-IDS(Intrusion Detection System); SOM(Self-Organizing Map); K-Means

I. INTRODUCTION

With the development of internet, internet security has been one of the most important problems in the world. Network intrusion detection is the process of monitoring the events occurring in a computer system or network and analyzing them for signs of intrusion. In data mining and intelligent computing, both K-means and Self-Organizing Map (SOM) are two of the most important unsupervised learning processes used to find the patterns in collection of unlabeled data. As a powerful tool for data visualization and mining, the SOM has been applied to a variety of areas [1, 2]; such as pattern recognition, imaging analysis, industry process monitoring, fault detection, intrusion detection and so on. When it comes to Intrusion Detection System (IDS), SOM neural network has become one of the most powerful tools to help IDS distinguish the abnormal data from the raw data captured from the host or network. Nevertheless it is not satisfactory in many places. For instance, it cannot provide with us the precise clustering results; its speed of convergence is slow and so on [3]. As a simple and fast clustering algorithm, K-Means clustering algorithm can process large database effectively, however its weak point is also conspicuous: it depends largely on a selection of the initial synaptic weights and the input patterns; otherwise it fails to converge or converge to a local optimum [4].

This paper put forward a new algorithm called S-K, where the combination of SOM neural network and K-means algorithm is running to detect the abnormity of the nodes in the wireless sensor network, which will make the system more flexible, precise and easier to implement.

II. S-K ALGORITHM

In this algorithm we combine SOM neural network and K-means algorithm to cluster the data obtained from network. When we use the S-K algorithm, there are two methods to implement the S-K algorithm. They can both improve the efficiency of the algorithm to some extent.

A. SOM Algorithm

First, SOM is a neural network algorithm, the weight of its input nodes and output neurons connect with each other. Competing is ongoing among the input neurons to choose. Among the output neurons there suppression, functionally, it can be a single neuron changes in the rules and a group of neurons linked to changes in the rules. Therefore, the entire neural network is the function of self-organizing through the use of a large number of training sample data to adjust the weights. So the network output reflects the distribution of data [5].SOM learning process, including the process of competition, the process of cooperation and renewal process.

B. K-Means Algorithm

The K-means algorithm [6] is one of the most popular methods for clustering multivariate quantitative data. This algorithm is non-parametric in nature as it does not assume any probability model for the data. Given a fixed number of clusters, it determines an assignment of the data vectors (observations) to the clusters so as to minimize the total of the squared distances between the observations assigned to the

This paper provided by "863" project plan of China (No. 2007AA01Z450).

978-0-7695-3997-3/10 $26.00 © 2010 IEEE
DOI 10.1109/ICEE.2010.327

same cluster and summed over all clusters. This algorithm uses the Euclidean squared distance measure.

C. Using K-means after SOM

In this method, the sample will be clustered by SOM neural network first, and after the number of clusters and the center of each cluster are obtained by SOM, we use K-Means to refine the clusters. The detailed description of algorithm is as follows:

The first stage (SOM stage): The number of cluster and the cluster centers will be obtained by the SOM clustering algorithm in this stage.

1) Initialization of weights: To W_j (j = 1, 2, ···, p) , the weights vector that joints input node and the j-th output node, is given by random number. And the loop number t should be initialized, set t=1.

2) Weight adjustment: For each input pattern X_k (k = 1, 2, ···, m):

a) The weight vector of the smallest distance between X_i and W_j can be obtained by the following formula:

$$[X_k - W_g] = \sum_{j=1}^{p} \| X_k - W_j \| \tag{1}$$

b) Node g is defined for winner node, and Ng(t) is defined for the neighborhood of the winner. The weights in the neighborhood should be adjusted by the following formula:

$$\Delta w_{ij} = \eta(t) | x_{ik} - w_{ij} |, w_{ij} = w_{ij} + \Delta w i_j$$

In the formula, $\eta(t)$ is the learning rate, which decreases with the increment of the number of training times; x_{ik} is the input of the i-th node of the k-th data sample and j∈ Ng(t).

c) Repeat the step 2 till the network weight is steady.

d) After the convergence of the network, according to the response of output node, the sample clustering can be completed.

3) Determine the cluster centers and the number of cluster centers: The number of the clusters C and the cluster centers $Z = \{Z_1, Z_2, ..., Z_c\}$ can be obtained.

The second stage (K-means): The output in the first stage can be used as the initial input of the K-means algorithm.

4) Threshold setting: According to the clustering results obtained from the first stage, $Z = \{Z1(1), Z2(1), ..., Zc(1)\}$ is the center of each cluster. $Y = \{y1, y2, ...yn\}$ is the output after clustering by SOM in the first stage. $Sj = \{y | y \in Sj\}$ is the sample collection whose elements all falls within a cluster whose center is Zj. And the threshold of the iterative loop for stop is ε .

5) Sample division: Through this step, each sample vector should be divided into one of clusters. The condition of division is:

$$Y_p - Z_j(l) \| < \| Y_p - Z_i(l) \| \ i = 1, 2, ...C; i \neq j \tag{2}$$

If Yp meets the inequality, then Yp ∈ Sj , Sj is representative of cluster j.

6) Recalculate the centers of new cluster: With the new clusters we got from the step 5, the centers of each new cluster should be recalculated, so that the sum of distance Jj between each vector in the cluster and the new cluster center can be smallest.

$$Jj = \sum_{Yp \in Sj} \| Yp - Zj(l+1) \|^2, j = 1, 2, ...C \tag{3}$$

$Z_j(l+1)$ is the new cluster center of the cluster j, which can be calculated by the following formula:

$$Z_j(l+1) = \frac{1}{N_j} \sum_{Yp \in Sj} Y_p \tag{4}$$

In the formula above, Nj is the number of the sample vectors in the cluster j.

Convergence checking. If $\| Zj(l+1) - Zj(l) \| < \varepsilon (j = 1, 2, \cdots n)$ then the algorithm can be stopped, or it will turn to the step 2 to continue iterating.

The flow chart of algorithm can be described as follows:

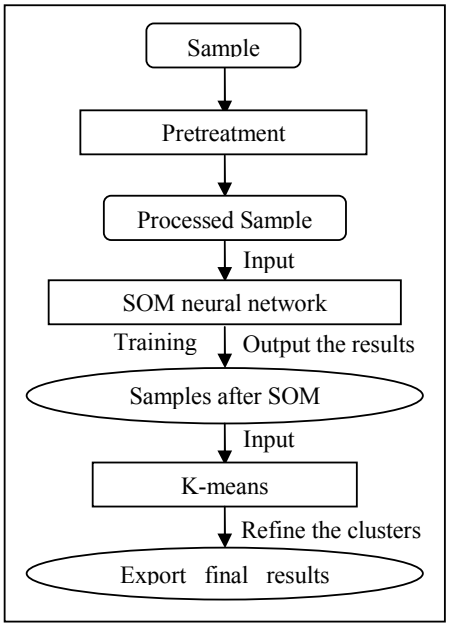

Figure 1. Flow chart of S-K 1

D. Using K-means during the SOM

In the second method, we use K-means algorithm after the training phase of SOM to refine the weights obtained from SOM training, which can eliminate the gray nodes of the weights, so it can help SOM recognize more exactly.

The differences between the two methods are the occasion and the object of the application of K-means algorithm. The former is using K-means to refine the clusters after the SOM, while the latter is using K-means to refine the weights of SOM neural network after SOM has finished training. The second method can be described as follows:

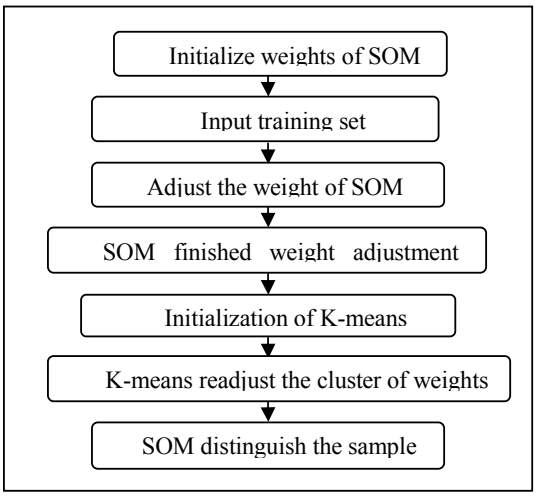

Figure 2. Flow chart of S-K 2

III. SIMULATION ANALYSIS

A. Process and result of simulation

In this experiment, we use windows XP system, and Eclipse for develop platform, the data of simulation testing comes from KDD CUP 99 [7] data source, and in the training step we use 1000 pieces of normal network records from the file called kddcup.data_10_percent.

In this experiment we combine the two methods together, that is, when the SOM finishes training, we adopt the K-means to refine the weights obtained by training, and when SOM finishes clustering, K-Means is also applied to refine the final result of clustering.

We use 20*20 for weight scale of competition layer. After training we can get the figure of weight U-Matrix [8] as follows:

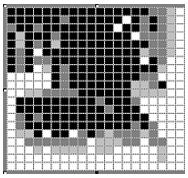

Figure 3. U-Matrix of weight by SOM

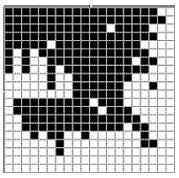

Figure 4. U-Matrix of weight after Refining by K-Means

We use the weight of SOM as input of K-Means, which can eliminate the gray nodes in the weights, and can help weight of SOM provide more precious cluster results.

After training, 100 clusters which represent the normal patterns can be obtained, for the requirement of the IDS. These 100 clusters will be all denoted by deep color, while bright color represents abnormal network data, that is, intrusion. As the figure 3 shows, the deep of color represents the distance between this node and normal pattern, deeper color means nearer the normal pattern, on the opposite, brighter color means nearer the abnormal pattern, that is, the intrusion.

We can see that there are lots of gray nodes in the figure 3, which will bring more troubles for SOM to recognize the intrusion. So we use K-means to refine the clusters of weight. After refining, as the figure 4 shows, we can see that there are no gray nodes in the U-matrix. In figure 4, there are 100 clusters which can represent the normal pattern (all denoted by black color) and only one cluster which can represent intrusion (denoted by white color).

After training, we can enter the recognizing step. In this experiment, we will use three group data, and only one type of intrusion will be introduced in each group. These testing data will be recognized by SOM, and then will be also refined by K-means. The table below will compare the S-K algorithm with the traditional SOM algorithm.

TABLE I. THE COMPARISON BETWEEN S-K AND SOM

Input type (normal record + intrusion)	S-K		SOM	
	D rate %	*F rate %*	*D rate %*	*F rate %*
Normal + buffer overflow	93	18	90	56
Normal + IP sweep	97	22	77	59
Normal + Smurf	100	9	89	32

In the table above, D rate means detection rate, while F rate means false positive rate. We will easily find that it has higher detection rate in table 1. And then we will put this testing input together, this time there is only one group of data in the simulation testing, which is made up of 3500 pieces of records, and includes normal network data and 3 types of intrusion, that is, buffer overflow, IP sweep and Smurf. The testing result is as follows:

TABLE II. THE COMPARISON BETWEEN S-K AND SOM

Input type (normal record + intrusion)	S-K		SOM	
	D rate %	*F rate %*	*D rate %*	*F rate %*
Normal + buffer overflow+ Smurf + IP sweep	61	52	79	33

B. Analysis and improve of simulation result

The first simulation shows that the detection algorithm combining K-means and SOM can increase the detection rate and reduce the false positive rate obviously. However, in the second simulation testing, when more than one type of intrusion is introduced, S-K algorithm will not work as well as in theory. That because the application of K-Means will eliminate the fuzzy nodes in the former, but in the latter K-means algorithm became chaos when clustering, so it will reduce the detection rate of SOM.

After SOM finishing training with normal inputs, the stimulated nodes can represent n kinds of normal network patterns (n clusters), while the nodes which are not stimulated, that is the (n+1) cluster can represent the intrusion pattern. In the first simulation, one type of intrusion was introduced in each group, the same kind of intrusions are similar with each other. So through the recognize by SOM, the last cluster only includes one type of intrusion, which will make this cluster which represents intrusion more compact, when we use K-means to refine the result, in the last cluster, the distance between each vector and the center of this cluster is much smaller than the distance between each vector and the centers of clusters which represent normal pattern, so we can get correct result.

However, in the second simulation testing, we put the three kinds of intrusion in the same cluster, so the distance between each element of this cluster and the center of this cluster may be larger than the distance between each element of this cluster and the centers of clusters which represent normal pattern, so when we use K-Means to refine the cluster, it will put the intrusion sample into the normal clusters incorrectly.

For this situation, an effective way to solve that is to set a threshold ε. Suppose that the distance between each sample and normal clusters (the former 100 clusters) is not larger than ε. If it is so, the K-Means algorithm will continue, on the opposite, the sample will be divided into the 101st cluster, that is, the cluster representing intrusion. We give the value of 2.0-3.0 to the threshold, which will guarantee the IDS keep a higher detection rate and lower false positive rate. The following table shows the result after the threshold is introduced to the algorithm.

TABLE III. THE COMPARISON BETWEEN S-K AND SOM

Input type (normal record + intrusion)	S-K		SOM	
	D rate %	*F rate %*	*D rate %*	*F rate %*
Normal + buffer overflow + Smurf + IP sweep	92	35	79	33

The table 3 shows that the improved S-K algorithm has a big progress on the detection rate. However, there is still a little higher false positive rate, this is because the introduction of the threshold. A larger threshold may cause a lower detection rate, while a smaller threshold may result in a higher false positive rate. So deciding a reasonable threshold is of great importance.

The relationship between detection rate and false positive rate is an eternal contradiction. After a number of experiments we can conclude that giving the value between 2.0 and 3.0 to the threshold can keep a high detection rate and control the false positive rate in a low range.

CONCLUSIONS

In this paper, we use the algorithm which combines SOM neural network and K-means algorithm to cluster the data. In this algorithm, we introduce two methods to combine the SOM and K-Means, and we use KDD CUP 99 as data source, through plenty of simulation testing, though the detection rate improved obviously, some defects are also found that the method can only detect single type of intrusion. To this limitation, a threshold mechanism is introduced to solve it. Using the improved S-K algorithm, the system can keep a high detection rate and control the false positive rate in a low range. At last, we also do the research on the relationship between the threshold and detection rate, false positive rate. In the future, we will continue doing the research of how to keep the high detection rate and reduce the false positive rate effectively.

ACKNOWLEDGEMENT

We are grateful for the computing resources provided by "863" project plan of China (No. 2007AA01Z450 and No. 2007AA01Z188), National Natural Science Foundation of China (No.60773073 & No.60604010), Key project of Ministry of Education of China (No.208010), Education Science and Technology Foundation of Tianjin (2006BA19).

REFERENCES

[1] Widrow B, Rumelhart D E, and Lehr M A, "Neural Network: Application in Industry, Business and Science," Communication of the ACM., vol.37, no.3, pp.93-105, 2006.

[2] C.-H. Chang, P. Xu, R. Xiao, and T. Srikanthan, "New adaptive color quantization method based on self-organizing maps," IEEE Trans. Neural Netw., vol. 16, no. 1, pp. 237–249, Jan 2005.

[3] Usama Fayyad, Cory Reina, P. S. Bradley, "Initialization of Iterative Refinement Clustering Algorithms," Microsoft Research Technical Report MSR-TR-98-38, June 2006.

[4] Dan pelleg, Andrew moore, "Accelerating exact k-means algorithms with geometric reasoning," Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining., pp. 277-281, 1999.

[5] S Lee, RG Lathrop, " Subpixel analysis of Landsat ETM + using self-organizing map (SOM) neural networks for urban land cover characterization," IEEE Transactions on Geoscience and Remote Sensing., vol. 44, pp. 1642-1654, 2006.

[6] Dingxi Qiu, Ajit C.Tamhane, "A comparative study of the K-means algorithm and the normal mixture model for clustering: Univariate case," Journal of Statistical Planning and Inference 137, pp. 3722 – 3740, 2007.

[7] Ding Li, Ni Gui-qiang, Pan Zhi-Song, Hu Gu-Yu, "DDoS intrusion detection using generalized grey self-organizing maps," 2007 IEEE International Conference on Grey Systems and Intelligent Services, pp. 1548-1551, 2007.

[8] N. Mitton and E. Fleur, "Distributed Node Location in clustered multihop wireless networks", RRN 5723, INRIA, 2006.