

Application of Self-Organizing Feature Map Neural Network Based on *K*-means Clustering in Network Intrusion Detection

Ling Tan^{1,*}, Chong Li², Jingming Xia² and Jun Cao³

Abstract: Due to the widespread use of the Internet, customer information is vulnerable to computer systems attack, which brings urgent need for the intrusion detection technology. Recently, network intrusion detection has been one of the most important technologies in network security detection. The accuracy of network intrusion detection has reached higher accuracy so far. However, these methods have very low efficiency in network intrusion detection, even the most popular SOM neural network method. In this paper, an efficient and fast network intrusion detection method was proposed. Firstly, the fundamental of the two different methods are introduced respectively. Then, the self-organizing feature map neural network based on *K*-means clustering (KSOM) algorithms was presented to improve the efficiency of network intrusion detection. Finally, the NSL-KDD is used as network intrusion data set to demonstrate that the KSOM method can significantly reduce the number of clustering iteration than SOM method without substantially affecting the clustering results and the accuracy is much higher than *K*-means method. The Experimental results show that our method can relatively improve the accuracy of network intrusion and significantly reduce the number of clustering iteration.

Keywords: *K*-means clustering, self-organizing feature map neural network, network security, intrusion detection, NSL-KDD data set.

1 Introduction

With the status improvement of Internet technologies in modern society, network security now has become a widespread concern. In addition, in pace with the constant upgrades of related industries brought by new technologies such as cloud computing, networking, big data and block-chain, a huge number of distributed users and ubiquitous access to the Internet make the enterprise data boundary no longer exist. Meanwhile, the amount of data is increasing, which leads to a higher value and more security challenges. Moreover, considering the openness of the Internet platform with diverse audiences and connection types, the Internet is vulnerable to external attacks and its security is bound to be greatly

¹ College of Computer and Software, Nanjing University of Information Science & Technology, Nanjing, 210044, China.

² College of Electronic and Information, Nanjing University of Information Science & Technology, Nanjing, 210044, China.

³ School of Geography, Geology and the Environment, Keele University, Keele, UK.

* Corresponding Author: Tan Ling. Email: cillatan0@nuist.edu.cn.

threatened [Lin (2015)]. In view of these severe issues, intrusion detection technology is particularly important. Network intrusion detection is a research hot spot but also a difficulty in the field of network security and its essence is a clustering problem [Wang, Peng and Luo (2012)].

As we can see, regardless of labeled supervised learning or unlabeled unsupervised learning, there are different network models and algorithms for data classification problems with the constant development of machine learning. However, due to the high dimension of intrusion feature data of these classification problems in the field of network security as well as small difference between different intrusion categories, many intrusion patterns cannot be accurately clustered. Li et al. [Li, Guan, Zan et al. (2003)] Proposed a network intrusion research based on fuzzy clustering generalized regression neural network (GRNN), in which the method was focus on specified types of intrusion action pattern with the combination of fuzzy clustering and GRNN analysis. Despite of its high classification accuracy and reliability, the method cannot reach the requirements in terms of the diversity of network intrusion. Yi et al. [Yi, Li and Wang (2009)] Proposed a network intrusion detection method based on support vector machine, in which the method still has some problems such as high false alarm, excessive demands data for building up detection models, and low detection performance in small sample conditions without enough training data. By comparing network intrusion detection methods with different network models and algorithms, intrusion detection method based on self-organizing feature map (SOM) neural network is better, which has a shorter training time and a higher detection efficiency [Wang (2007); Wu and Yan (1999); Wan, Wang, Feng et al. (2012)], and it is able to make up deficiencies of the above network models and algorithms, however, the accuracy of network intrusion detection still has room for further improvement, and there are also shortcomings in the application of unsupervised learning. Therefore, according to the advantages of K -means clustering algorithm to local classification [Kohonen (1995); Jari and Kohonen (1996)] based on the above analysis, an intrusion detection algorithm based on K -means clustering self-organizing feature map neural network (KSOM) network is proposed in this paper. This method even has higher classification accuracy for network intrusion detection and supports unsupervised network anomaly detection.

This paper is organized in four sections. In the second section, an improved network intrusion detection method based on K -means clustering self-organizing feature map neural network (KSON) is comprehensively introduced, as well as introduced respectively theoretical background of the related SOM network model and K -Means clustering algorithm. The third section gives the experimental process and result analysis. In this section, we introduce a new network intrusion detection database. In addition, we propose a new feature extraction method. The conclusion of this paper is drawn in the fourth section.

2 KSOM theoretical backgrounds

Due to the high dimensions of the network intrusion feature data, data differences among different intrusion classes are relatively small. Local optimal clustering result cannot be achieved if SOM neural network is simply adopted. And the corresponding learning rate

will be low, which makes many intrusion modes unable to be accurately clustered. Note that K -means clustering algorithm has high clustering performance in dealing with big data sets. And its time complexity is almost linear, whose clustering iteration time is significantly reduced with the advantage of local optimal clustering result. According to the above analysis, the feature of local optimum based on K -means clustering is adopted to improve the SOM neural network to cluster the network intrusion data in this paper.

2.1 SOM neural network

SOM neural network was first proposed by the Finnish mathematician Teuvo Kohonen in 1982 [Teuvo (1982)]. It is an unsupervised competitive learning network that simulates the working characteristics of human brain. By automatically searching the intrinsic and essential attributes of the sample, the parameters and structure of the network are changed with self-organization and self-adaption. It mainly implements the function of mapping similar sample points in the high dimensional space to the adjacent neurons in the network output. For a certain problem, a certain number of neurons are set, and each neuron corresponds to a weight vector. The different classes of data are clustered into different neurons for classification purpose through training [Bukka, Sivarama and Raju (2018)] as shown in Fig. 1.

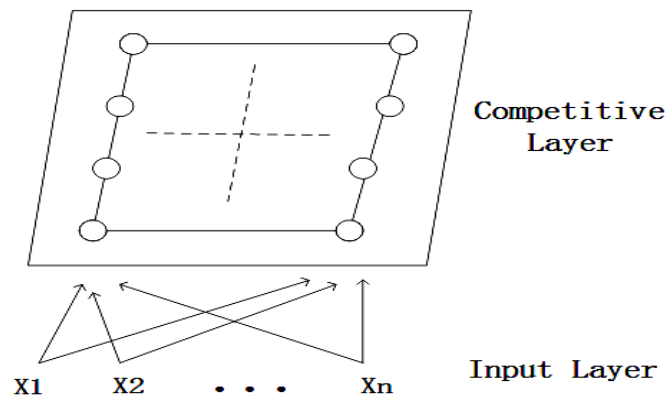


Figure 1: The network topology of SOM neural network

It is a feed forward neural network consisting of two layers of neurons, and Fig. 1 is divided into the input layer and the competition layer, i.e., the output layer. The explanation between the two layers is as follows:

Input layer: The outside information is collected to each neuron of the output layer through the weight vector.

Output layer: The winning neuron and the neighborhood are obtained through weight competition using competition algorithm.

The neurons between the two layers are connected by bidirectional connections, and there is no hidden layer in this network. The basic thought of this network model is the corresponding competition opportunities of the competitive layer and the neuron competition to the input mode, and there exists a neuron that eventually becomes the

winner of the competition, this winning neuron represents the classification of the input pattern, thus achieving the result of the clustering. The keywords of the SOM mainly include winning neuron, neighborhood and weight adjustment function.

(1) Winning Neurons (WN)

For input vector $X = (x_1, x_2, \dots, x_n)$ and competitive layer (output layer) neuron j , the network weight is $W_{ij} (i = 1, 2, \dots, n; j = 1, 2, \dots, m)$. The distance between the input vector and the weight vector is calculated by Euclidean distance formula Eq. (1), and the minimum value of the distance corresponding to the neuron in the SOM is then obtained.

$$d_j = \sqrt{\sum_{i=1}^m (x_i - w_{ij})^2} \quad (1)$$

The main approach is to promote competition among the output neurons of the network in an attempt to be activated. As a result, there is only one output neuron being activated at each moment, while the statuses of the other neurons are inhibited. This activated neuron is called winning neuron (WN).

(2) Neighborhood

For each WN, the input vector X represented by its nearest neuron generally refers to the neuron near WN in SOM, i.e., the neuron neighborhood (N).

(3) Weight Adjustment Function

This function describes the adjustment of WN and N that are changing with the number of iterations t relative to input vector X , which can be represented by (2),

$$W_{ij}(t+1) = W_{ij}(t) + \eta(X_i - W_{ij}) \quad (2)$$

, where t is the number of iterations, and η is the learning rate.

The training process of SOM neural network can be divided into four steps:

Step 1: Initialize network weight W_{ij} , and set the maximum number of iterations T ;

Step 2: Normalize each input vector X in NSL-KDD network intrusion detection database, and calculate WN and N;

Step 3: Update WN, N and W_{ij} according to the weight adjustment function;

Step 4: Execute the above steps repeatedly until the number of iterations reaches the given maximum number of iterations.

2.2 K-means clustering

The K -means clustering was proposed Johnson and Wichern in 2002, and then it is probably the most popular methods in the world [Capó, Pérez, Jose et al. (2017)]. K -means clustering algorithm is also a most commonly used indirect clustering algorithm based on sample similarity. It belongs to an unsupervised learning method, and the cluster is divided by minimizing the clustering error [Yu, Chu, Wang et al. (2017)]. In addition, it is a classic dynamic clustering algorithm on the basis of point wise modification iteration, the key point of which adopt the sum of square errors as criterion function. Its basic thought is to randomly select data objects as the initial center from the

data set containing data objects first, then calculate the distance between each data object and each center. According to the nearest neighbor rule, all data objects will be partitioned into the clusters represented by the nearest center. Then the mean of the data objects in each cluster is calculated as the center of each cluster. By comparing the new center and the last center, it can be asserted that if the new center does not change, then the algorithm converges successfully and will output the result, otherwise all the data objects will be re-divided according to the new center until the convergence conditions of the algorithm are satisfied. However, the accuracy of the K -means procedure is very dependent upon the choice of the initial seeds. Thus, this paper presents the efficient strategy to improve the K -means performance which hybrid SOM neural network, the next section will explain in detail carefully.

The concrete implementation steps are as follows: Given data set $X = \{x_1, x_2, \dots, x_N\}, x_n \in R^d, (n = 1, 2, \dots, N)$, K -means clustering algorithm will divide the data set into M disjoint clusters according to the optimal clustering criteria. The clustering algorithm needs to assign again cluster number k and initial cluster center m_i , and calculate the distance between the input vector of each point to the cluster center through different equations, then a new cluster center m_i^* can be obtained, later the training process will repeat until the change of the cluster center is smaller than a given value ε , which is the ending-criterion of the algorithm.

2.3 KSOM algorithm

KSOM algorithm proposed in this paper uses K -means clustering algorithm to cluster local data classified by SOM, so that the accuracy of the classification of the whole network intrusion data will reach its best level. The core idea of the KSOM network intrusion detection method proposed in this paper is that the winning neural has reached by SOM to merge by K -means clustering algorithm. Thus, the weight vector of SOM neural network is adopted as the input of K -means clustering algorithm. Then the input data set is normalized to facilitate model training, and finally sent to the network for training and learning. Besides, in the local clustering process, the network weight of the SOM will be constantly adjusted along with the change of the iterations, and will gradually approximate to the input data. Therefore, we will not solve the SOM cluster center in KSOM algorithm, but only calculate its network weight vector. Then the network weight vector is adopted as the input of the K -means clustering algorithm for local clustering, and finally the result of clustering is achieved.

The training data set logic chart as shown in Fig. 2:

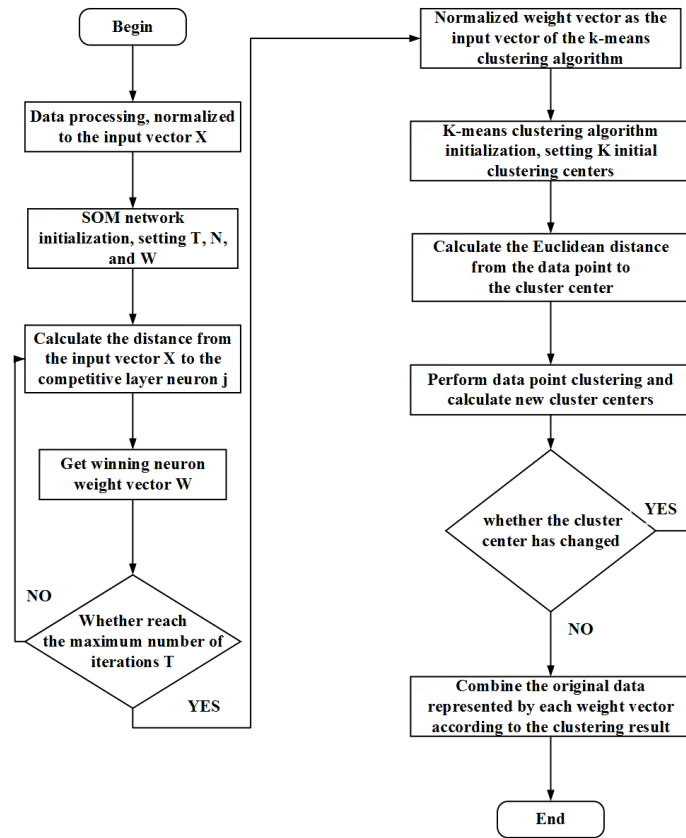


Figure 2: Entire clustering process using the KSOM method

The training process of KSOM algorithm can be divided into the following seven steps:

Step 1: Use the network weights extracted from the training results of the SOM neural network as the input of the K -means clustering, and the input is processed with normalization;

Step 2: Select k data elements m_1, m_2, \dots, m_k randomly as initial cluster center;

Step 3: Calculate the Euclidean distance between each data point and the cluster center;

Step 4: Cluster the data in accordance with the criterion that minimum clustering can be obtained when it reaches cluster center;

Step 5: Adjust the cluster center to regulate Eq. (3), where N_j represents the number of data contained in class j , and C_j represents the data sets of class j ;

$$m_j^* = \frac{1}{N_j} \sum_{x_i \in C_j} I(x_i), (j = 1, 2, \dots, k) \quad (3)$$

Step 6: Judge whether $\|m_i - m_i^*\| < \varepsilon$ is satisfied or not (where ε is a given value, and its value should be as small as possible), if so, go to Step 7, otherwise go to Step 3;

Step 7: According to the clustering results of Step 6, cluster locally the original data represented by each network weight value and the algorithm ends.

Through the above steps, the modification of training algorithm of SOM in this research affects neurons which are not close to the winner neuron on feature space, and the winner neuron pulls more neurons into the same cluster. Further, WN in competitive layer is clustered again by the improved KSOM based on K -means algorithm to ensure that the neurons are not being suppressed in the winning neighborhood (N) of the competitive layer of SOM during the competition. For each iteration, the local WN will lead the adjustment degree of the weight vector change greatly, which makes weight vector become smaller and smaller, and the number of iterations decreases. The final learning rate is significantly improved as well as the clustering accuracy.

After training, the weight vectors of each neuron in the SOM neural network will separately cluster the input vectors to certain neurons, thus the classification result of different data is obtained. As mentioned above, network intrusion detection is essentially a clustering problem. Therefore, SOM neural network has a certain network intrusion detection effect. Although the intrusion detection method based on SOM neural network has the features of high detection rate, short training time and high versatility, there is still further room for improvement of its detection accuracy, and there are also shortcomings in unsupervised learning applications. Especially the choice of the learning rate makes it have to compromise between the learning rate and stability of the final weight vector, and sometimes the local neurons cannot converge precisely, which makes the initial weight vector of the neuron more distant from the input vector, so that it is unable to win in the competition. Therefore, according to the advantages of K -means clustering algorithm in local clustering implementation as well as its significant learning rate performance, a KSOM intrusion detection algorithm based on SOM is proposed in this paper.

3 Experiments procedure and result analysis

3.1 Data description

In this paper, NSL-KDD network intrusion detection database is adopted as the training and testing data set [Revathi and Malathi (2013)]. Compared with KDD-CUP99 network intrusion detection data set, this data set is provided in CSV format, which has eliminated redundant data in KDD-CUP99 data set, and has overcome the shortcoming that the classifier is biased towards repeated records, and the performance of learning methods is vulnerable to be affected. Moreover, NSL-KDD data set has properly selected the proportion of normal and abnormal data and the volume of test set and training set data is more reasonable, therefore, it is very suitable for model algorithm validation and result evaluation analysis. NSL-KDD network intrusion detection database not only can reflect the structure of network traffic and intrusion conditions, but also can be modified, extended and duplicated. The record of NSL-KDD network intrusion detection database is as shown in Tab. 1.

Table 1: NSL-KDD network intrusion detection database record truncation table

0	tcp	ftp_data	SF	491	0	...
0	udp	other	SF	146	0	...
0	icmp	Ecr_i	SF	1480	0	...
0	tcp	http	SF	232	8153	...
0	tcp	http	SF	199	420	...
15159	tcp	ftp	SF	350	1185	...
0	tcp	private	S0	0	0	...
315	udp	other	SF	146	105	...
240	tcp	http	SF	328	275	...
0	tcp	private	S0	0	0	...
0	tcp	private	REJ	0	0	...
5607	udp	other	SF	147	105	...

As shown in Tab. 1, NSL-KDD network intrusion detection data set contains five types of data, that is, Normal, Probe, R2L, U2R and Dos, and consists of 125973 training data and 22544 test data sets, each record contains 41 features, such as duration, protocol, service, symbol, source byte and target byte. “attack” vs. “normal” labels as shown in Figs. 3, and 4 different types of attacks as shown in Fig. 4.

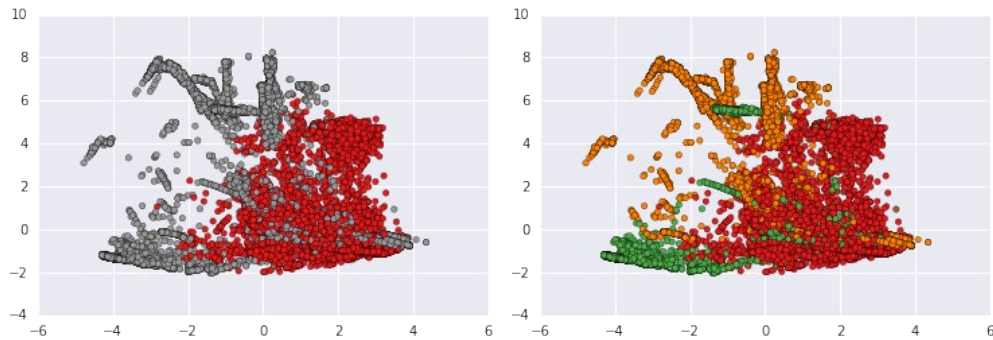


Figure 3: The distribution of attack and normal **Figure 4:** The distribution of 4 different types of attack

3.2 Data process

In this paper, the NSL-KDD data sets are used as training sets and testing sets of our paper, the KSOM utilizes numerical values for the process of data training and testing. Further, the data preprocessing approaches are divided into feature extraction and feature transformation models based on working methodology over the big data processing [Kamei and Kawakami (2018)]. Thus, in order to facilitate the next model training, first step in experiments is that turn the character values into numerical values and use the attribute ratio data feature extraction method which ascending feature according to its repeated time

for data processing. For example, TCP protocol type have a greater number of repeated time, so we convert TCP protocol type into number 1, etc. as shown in Tab. 2.

Table 2: Numeric values of NSL-KDD features

Protocol Type	Feature Value	Service	Feture Value	Flag	Feature Value
TCP	1	HTTP	1	SF	1
UDP	2	Private	2	S0	2
ICMP	3
	
	
		HTTP_2748	70	OTH	11

Meanwhile, use the forward error correction (FEC) coding technology to encode the classification features into discrete features for the symbolic features of the 41 features, so that the one-dimensional vectors can be transformed into 3 dimensional vectors, such as TCP, UDP and ICMP. Similarly, several other features are mapped into different dimensions of vectors, as shown in Tab. 3.

Table 3: Different symbol feature vector dimensions before and after processing

Sign features	Pre-processing dimension	Pre-processed dimension
Protocol	1	3
Service	1	70
Flog	1	11

Then execute the normalization process of the numerical characteristics, the normalization Eq. (4) is given as follows:

$$val_{new} = \frac{val - V_{min}}{V_{max} - V_{min}}, \tag{4}$$

where *val* is the original value, *val_{new}* is the normalized data value, *V_{min}* and *V_{max}* are the maximum value and the minimum value of this specific data, respectively. When normalization process is done, each feature will be restricted to an interval of [0,1]. After the data set is preprocessed and normalized, network training will start.

3.3 Experimental process

In this paper, Python 3.6 environment is adopted to training and testing the NSL-KDD data sets. The implementation of the corresponding pseudo-code is shown in Fig. 5.

```

Begin
Import NSL-KDD data sets
Initialized t, T, k, m(k)
If  $\|m(k)-m(k)^*\| < \varepsilon$  :
  If t = T:
    For data in NSL-KDD:
      For weight in range(Len(data)):
        Normalized data ,weight
        find max dot(data, weight)
        Get WN and N
        Readjust weight
      receive weight
    Normalized weight
    Calculate the ED from weight
     $m(k)^* = m(k)$ 
  receive new weight

```

Figure 5: KSOM model algorithm pseudo code

Firstly, data import and initialization processes of the NSL-KDD network intrusion detection data set are implemented as well as normalization. Secondly, the clustering algorithm is executed including the judgment of the end condition. Then the output is transferred to the next layer of K -means clustering model. Similarly, the data normalization is first implemented, then clustering process starts, and finally the judgment of the end condition is implemented. When both of the two end conditions are satisfied, the clustering result is obtained.

In this paper, we tested respectively SOM neural network, K -means and KSOM method with NSL-KDD set data. First approach is SOM, the learning rate η was set to 0.01. Second approach is K -means. For over fitting not happen we set the kind numbers k equal 8 and the initial point is chosen as far as possible. The proposed KSOM method the learning rate η was also set to 0.01. But there is a difference that the initial point was chosen as the weight vectors of each neuron in the SOM neural network.

3.4 Result analysis

After repeated training and we have tested three different methods, as shown in Fig. 6, the average error of SOM neural network clustering along with the increase of iterations. It can be seen from Fig. 6, with the increase of iterations, the average error is gradually decreased, yet the update of the weight vector of SOM neural network in the process of training is not global optimal, thus the sum of the average error tends to be increased slightly, followed by the adjustment it begins to decline, hence it will show a saw tooth downward decline. And although the sum of the average error is small, the number of iterations takes 1600 times before it starts to stabilize. Compare the Fig. 7 the average error of K -means clustering along with the increase of iterations, the number of iterations have a substantial reduction which reached 10 times. Unfortunately this method is at the expensive of increased average error. And the proposed of KSOM method as shown in Fig. 8, It can be seen that clustering ends after 80 iterations, the error of which stopped at a certain value, and there is also no saw tooth downward attenuation found. It means that the KSOM method is able to solve the problem that SOM neural network cannot achieve a local optimum, and the number of iterations is reduced to a large extent,

which improves the clustering efficiency. Meanwhile, the KSOM method also deals with the problem of lower clustering result which happened in the *K*-Means method.

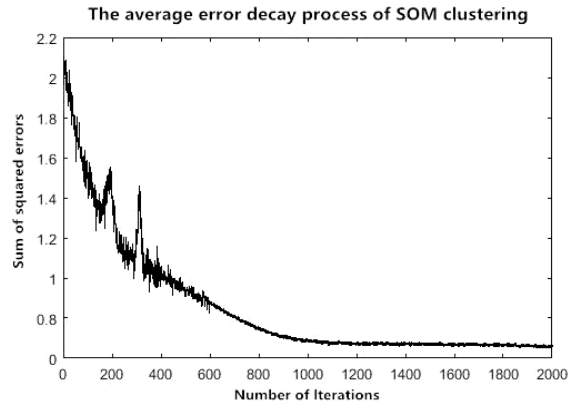


Figure 6: The average error decay process of SOM clustering

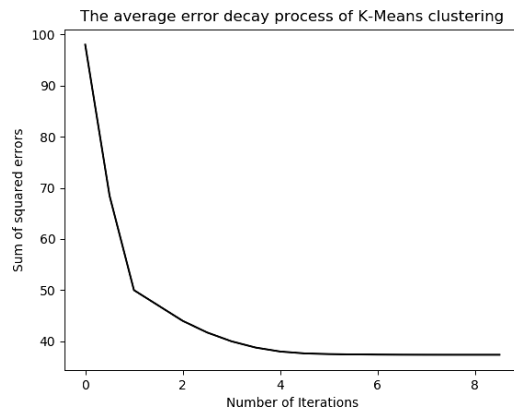


Figure 7: The average error decay process of *K*-Means clustering

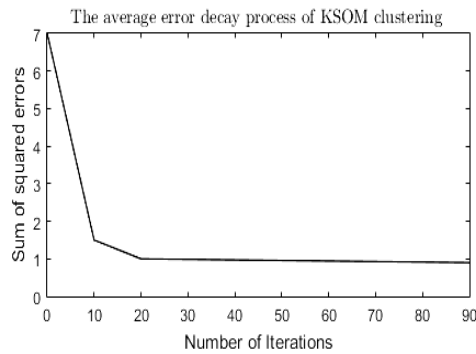


Figure 8: The average error decay process of KSOM clustering

The next in this paper, accuracy(AC) is adopted as the criterion to evaluate the network intrusion accuracy. The calculation Eq. (5) is given as below:

$$AC = \frac{TP + TN}{TP + TN + FP + FN}, \quad (5)$$

where TP is the number of correct classified attack behavior samples, TN is the number of correct classified normal behavior samples, FP is the number of wrongly classified normal behavior samples, and FN is the number of wrongly classified attack behavior samples.

In order to test the effect of the proposed KSOM network intrusion detection method based on K -means, NSL-KDD network intrusion detection data set was also trained and tested on other methods, and a comparison was made on the performances before and after the improvement of SOM neural network clustering method, as shown in Tab. 3.

Table 3: Comparison of clustering methods

Model Method	Number of iterations	Accuracy
K -means	8	83%
SOM	1600	96%
KSOM	80	94%

From Tab. 3 we can see that the accuracy of the improved KSOM based on K -means algorithm is higher than that of the original counterpart under the same data set. In addition, the KSOM model proposed in this paper not only is the number of iterations better than other algorithms, it also has good results in accuracy. Thus, this paper also compares other existing network intrusion detection algorithms on the accuracy of intrusion detection under the same data set. The comparison results are shown in Tab. 4. The result shows that the proposed KSOM based on K -means algorithm has a better performance in network intrusion detection.

Table 4: Comparison of accuracy of intrusion detection in different method

Model Method	Accuracy rate	Misuse rate
GA	85%	7.2%
BP	89%	4.8%
SVM	92%	0.92%
KSOM	94%	0.55%

4 Conclusion

To deal with the drawback of self-organizing neural network that it has a large number of iterations in network intrusion detection, a KSOM model was proposed in this paper by utilizing the characteristic that K -means clustering algorithm has faster clustering speed. The experimental results showed that the iterations of the KSOM method was significantly reduced in comparing with self-organizing neural network model and the accuracy was significantly improved with respect to other methods of network intrusion.

Meanwhile, our method achieved a higher efficiency relatively when implementing network intrusion detection. Thus, the KSOM method has high performance to provide more reliable data and support for the establishment of network security mechanism.

References

- Brugger, D.; Bogdan, M.; Rosenstiel, W.** (2008): Automatic cluster detection in Kohonen's SOM. *IEEE Transactions on Neural Networks*, vol.19, no. 3, pp. 442-459.
- Capó, M.; Pérez, A.; Lozano, J. A.** (2017): An efficient approximation to the *K*-means clustering for massive data. *Knowledge-Based Systems*, vol. 117, no. 1, pp. 56-69.
- Chen, Y. H.; Qin, B.; Liu, T.; Liu, Y. C.; Li, S.** (2010): The comparison of SOM and *K*-means for text clustering. *Computer and Information Science*, vol. 3, no. 2, pp. 63-69.
- Cuomo, S.; De, Angelis, V.; Farina, G.; Marcellino, L.; Toraldo, G.** (2017): A GPU-accelerated parallel *K*-means algorithm. *Computers & Electrical Engineering*.
- De, C. L. A.; Duarte, D. N. A.; De, S. M. B. C.** (2009): New developmental stages for common marmosets (*Callithrix jacchus*) using mass and age variables obtained by *K*-means algorithm and self-organizing maps (SOM). *Computers in Biology and Medicine*, vol. 39, no. 10, pp. 853-859.
- Furukawa, T.** (2009): SOM of SOMs. *Neural Networks*, vol. 22, no. 4, pp. 463-478.
- Huang, X. C.; Su, W.** (2014): An improved *K*-means clustering algorithm. *Journal of Networks*, vol. 9, no. 1, pp. 161-167.
- Jari, K.; Kohonen, T.** (1996): Developments and applications of the self-organizing map and related algorithms. *Mathematics and Computers in Simulation*, vol. 41, no. 1, pp. 116-122.
- Jia, W. T.; Zhang, F. L.; Tong, B.; Wan, M. C.** (2009): Application of self-organization map neural network in intrusion detection. *Computer Engineering and Applications*, vol. 45, no. 23, pp. 115-117.
- Kamei, K.; Kawakami, S.** (2018): Suppression of isolated clusters occurrence in self-organizing map considering the distance on data space. *ICIC Express Letters*, vol. 12, no. 1, pp. 449-456.
- Kohonen, T.** (1982): Analysis of a simple self-organizing process. *Biological Cybernetics*, vol. 44, no. 2, pp. 135-140.
- Li, H.; Guan, X. H.; Zan, X.; Han, C. Z.** (2003): Network intrusion detection based on support vector machine. *Journal of Computer research and Development*, vol. 40, no. 6, pp. 800-807.
- Lin, C. H.** (2015): Research and analysis on network security detection system implementation based on B/S architecture. *Beijing University of Posts and Telecommunications*.
- Maldonado, M. E.; Brady, P. G.; Mamel, J.; Robinson, B.** (1999): Incidence of pancreatitis in patients undergoing Sphincter of Oddi Manometry (SOM). *American Journal of Gastroenterology*, vol. 94, no. 2, pp. 387-390.
- Revathi, S.; Malathi, D. A.** (2013): A detailed analysis on NSL-KDD dataset using various machine learning techniques for intrusion detection. *International Journal of Engineering Research & Technology*, vol. 2, no. 12, pp. 1848-1853.

- Sakthi, M.; Selvadoss, T. A.** (2011): An efficient constrained *K*-means clustering using self-organizing map. *International Journal of Computer Science and Information Security*, vol. 9, no. 4, pp. 94.
- Kohonen, T.** (1982): Self-organized formation of topologically correct feature maps. *Biological Cybernetics*, vol. 43, no. 1, pp. 59-69.
- Wan, Q.; Wang, C.; Feng, Z. Y.; Ye, J. F.** (2012): Review of *K*-means clustering algorithm. *Electronic Design Engineering*, vol. 20, no. 7, pp. 21-24.
- Wang, B.; Peng, Y. T.; Luo, C.** (2012): Research on network intrusion based on fuzzy clustering generalized recurrent neural network. *Journal of Jiangxi Normal University*, vol. 36, no. 3, pp. 288-291.
- Wang, L. J.** (2007): An improved structure-adapting self-organizing neural network algorithm. *Microelectronics and Computer*, vol. 24, no. 1, pp. 106-109.
- Wu, Y.; Yan, P. F.** (1999): A study on structural adapting self-organizing neural network. *Acta Electronica Sinica*, vol. 27, no. 27, pp. 56-59.
- Yi, R. Q.; Li, W. H.; Wang, D.** (2009): Feature recognition based on self-organized neural network. *Journal of Jilin University (Engineering and Technology Edition)*, vol. 39, no. 1, pp. 148-153.