2017 10th International Symposium on Computational Intelligence and Design

Research on Intrusion Detection Based on Feature Extraction of Autoencoder and the Improved K-means Algorithm

Xingang Wang
School of Information
Qilu University of Technology
Jinan, China
wxg@qlu.edu.cn

Linlin Wang
School of Information
Qilu University of Technology
Jinan, China
2250924094@qq.com

***Abstract*—Nowadays, intrusion detection is a technology to effectively avoid a number of risks of network intrusion. The K-means algorithm is widely used in intrusion detection. But, the algorithm has some shortcomings, such as random selection of k value, sensitive selection of initial cluster centers, and low accuracy in clustering high-dimensional data. In order to make up for these shortcomings of the K-means algorithm, this paper proposes an AE-Kmeans architecture which combines an autoencoder with the improved K-means algorithm. The AE-Kmeans architecture realizes the dimension reduction and feature extraction of these original data by introducing an autoencoder, and uses the improved K-means algorithm to cluster these processed data. These improvements of the improved K-means algorithm mainly include two aspects: the first aspect, a new method is introduced to select initial cluster centers; the second aspect, the algorithm calculates the weight of each attribute with the coefficient of variation, and then uses these weights in Euclidean distance formula, resulting in a weighted Euclidean distance formula. Finally, the AE-Kmeans architecture uses KDD CUP99 data set for intrusion detection simulation experiments. The Experimental results show that the AE-Kmeans architecture not only enhances the ability to deal with high-dimensional data, but also improves the detection rate and reduces the error-detection rate compared with K-means algorithm.**

Keywords-intrusion detection; K-means clustering algorithm; autoencoder; AE-Kmeans architecture; feature extraction; weighted Euclidean distance formula

I. INTRODUCTION

With the rapid development of the information society, the Internet has quietly permeated all aspects of our daily life. Network has brought us a great deal of convenience, but also has brought many security risks. For computer security and network security, network intrusion has undoubtedly become the biggest threat. As one of the important security monitoring technologies, the intrusion detection technology can detect attacks before network attacks cause extensive damage and provide an important basis for the prevention strategy [1].

When dealing with network data, intrusion detection technology faces some problems, including large amount of data and high-dimensional data. If we select these high-dimensional data directly for the experiment. The results lead to lower detection efficiency and cumbersome calculation steps. A lot of researchers have put forward many methods that transform these high-dimensional data objects into lower dimensional samples to reduce the burden of intrusion detection. In view of data dimensionality reduction and feature extraction, principal component analysis (PCA) is widely used. These methods of linear discriminant analysis, local linear embedding, locality preserving projection and so on are also used to reduce dimension [2].

The K-means clustering algorithm is one of the most common algorithms in intrusion detection system. However, the algorithm is sensitive to select initial cluster centers. In view of the deficiency, the literature [3] has proposed an MDKM algorithm which uses the dynamic iterative calculation process to obtain those initial cluster center points. The document [4] has introduced a new algorithm for selecting initial cluster centers based on the distribution of data samples. And literature [5] has improved K-means algorithm according to the density of these data samples.

This paper introduces an AE-Kmeans architecture which is composed of an autoencoder model and an improved K-means algorithm. The research work of the AE-Kmeans architecture mainly includes a few aspects as follows: the first aspect, this paper introduces the autoencoder model to achieve dimensionality reduction and feature extraction of these original data; the second aspect, in order to avoid the blindness of randomly selecting the initial cluster centers, this paper proposes a new method to select initial cluster centers; the third aspect, taking into account the difference in importance of different attributes, this paper introduces the coefficient of variation to calculate the weight of each attribute, and then assigns these weights to Euclidean distance formula, resulting in the weighted Euclidean distance formula. These experiments indicate that the AE-Kmeans architecture effectively compensates for these shortcomings of the K-means algorithm.

II. RELATED CONCEPTS AND KNOWLEDGE

A. Autoencoder

The autoencoder put forward by Yoshua, Bengio and so on is an effective method for deep learning, and it can be regarded as a neural network model that can reproduce these important features of input samples as much as possible [6].

2473-3547/17 $31.00 © 2017 IEEE
DOI 10.1109/ISCID.2017.170

The autoencoder is mainly composed of two parts: encoder network and decoder network. The encoder compresses these raw data to obtain lower-dimensional data that can represent these characteristics of high-dimensional original data to the greatest extent. The decoder can decode these resulting data and restore them to these original data to a maximum extent through the decoding process. Their work process mainly includes three steps. In the first step, the encoder encodes the unlabeled data samples and obtains code encoding. In the second step, the decoder decodes the code encoding to obtain new data. In the third step, we calculate the information error between these new data and those original data, and then adjust these weight parameters of the encoder and decoder according to the error in order to reduce the reconstruction error to the minimum. Code encoding obtained is the characteristic representation of these original data samples [7]. Figure 1 is the working process of an autoencoder.

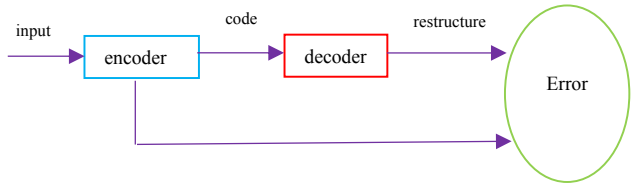

Figure 1. The working process of an autoencoder

B. K-means Clustering Algorithm

K-means algorithm proposed by MacQueen [8] is a classical clustering analysis algorithm. Compared with other clustering algorithms, the K-Means algorithm is relatively simple. However, the algorithm also has some disadvantages: the algorithm is sensitive to the selection of these initial cluster centers, and the k value needs to be given in advance. In addition, the algorithm is difficult to detect non-spherical clusters.

The K-means clustering algorithm is described as follows:

Input: a data set, k value
Output: clustering result
Step1: Select k points randomly from the data set as initial cluster centers.
Step2: Calculate the distance of each data object to each cluster center by using Euclidean distance formula. Then each data object is clustered in the cluster that's cluster center is closest to the data object.
Step3: Calculate the average value of each cluster, and use the mean as new center of the cluster.
Step4: Repeat Step2 and Step3. The clustering process is finished until each cluster does not change or the objective function converges.

C. The Improved K-means Algorithm

There are two improvements in the improved K-means algorithm: the first aspect, the algorithm proposes a weighted Euclidean distance formula; the second aspect, the algorithm uses a new method to select initial cluster centers.

1) The weighed Euclidean distance formula

All attributes of these data samples are treated equally in the K-means algorithm. But, the important attribute with great discrete degree plays more important role in clustering. Therefore, this paper introduces the coefficient of variation to obtain the weight of each attribute, and then uses these weights in Euclidean distance formula, resulting in a weighted Euclidean distance formula. The new formula not only reflects the importance of each attribute, but also achieves better clustering results.

The dataset X consisted of n data objects can be represented as $X = \{x_1, x_2, x_{3,} \cdots x_i, \cdots x_j, \cdots x_n\}$. Each data object made up of m-dimension property can be expressed $x_i = (x_{i1}, x_{i2}, x_{i3,} \cdots x_{ij,} \cdots x_{im})$. "$x_i$" means the i-th data object; "x_{ij}"represents the value of j-dimensional property of the i-th data object. The attribute value matrix of the dataset X can be expressed as follows:

$$X = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1m} \\ x_{21} & x_{22} & \cdots & x_{2m} \\ \vdots & \vdots & \cdots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{nm} \end{bmatrix}$$

The coefficient of variation is the ratio of the standard deviation to the mean. We use these formulas (1)(2)(3) to calculate the weight of each attribute, and use the weighted Euclidean distance formula which is formula (4) to calculate the distance between data points. These mathematical formulas are as follows.

$$\overline{x_j} = \frac{1}{n}\sum_{i=1}^{n} x_{ij}, \quad (1)$$

$$V_j = \frac{1}{\overline{x_j}}\sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_{ij} - \overline{x_j})^2} \quad (2)$$

$$w_j = \frac{v_j}{\sum_{j=1}^{m} v_j} \quad (3)$$

$$\mathrm{dis}(x_a, x_b) = \sqrt{\sum_{j=1}^{m} w_j (x_{aj} - x_{bj})^2} \quad (4)$$

"$\overline{x_j}$" means the average value of the j-dimensional attribute. "v_j" represents the coefficient of variation of the j-dimensional attribute. "w_j" represents the weight of the j-dimensional attribute.

2) The new method for selecting initial cluster centers

In K-means algorithm, the weakness which is random selection for initial cluster centers not only leads to poor clustering results and fall into local optimum, but also increases times of iterations and reduce the convergence speed of the algorithm. Based on the shortcoming of the K-means algorithm, this paper proposes a new method to select these initial clustering centers. The process of this method is described as follows:

a) Calculate the distance between each two data samples using the weighted Euclidean distance formula (4).
b) Calculate the average distance between all data samples according to the following formula (5).

$$Dis_{(Average)} = \frac{1}{A_n^2}\sum dis(x_a, x_b)\,(1 \le a \le n, 1 \le b \le n) \quad (5)$$

"n" represents the number of data samples, "A_n^2" represents the number of permutations that are made up of random two sample points from the data set.

c) Select a data point x_i $(1 \le i \le n)$ and find another data point $x_k, (1 \le k \le n \ and \ k \ne i)$ that's distance from x_i is less than $Dis_{(Average)}$. Then we check all the data points in this way, and count the number of such data that like x_k. We call these data points as nearest neighbors. Similarly, we count the number of nearest neighbors of each data point and sort all data points in descending according to their number of nearest neighbors.

d) Select the data point with the largest number of nearest neighbors as the first initial cluster center, and select the data point sorted to the second as another initial cluster center. We seek remaining cluster center points in this way. If a data point x_j $(1 \le j \le n)$ sorted to P is one of nearest neighbors of these selected cluster centers, we can ignore x_j, and check x_z $(1 \le z \le n)$ sorted to P+1. If x_z is not one of nearest neighbors of these existing cluster centers, we are able to use x_z as an initial cluster center and continue to look for these remaining initial cluster centers.

3) The description of the improved K-means algorithm

The improved K-means algorithm is described as follows:

Input: a data set;

Output: clustering result

Step1: Calculate the weight of each attribute for the data set according to these formulas (1)(2)(3);

Step2: Calculate the distance between every two data points by using the formula (4);

Step3: Calculates the average distance $Dis_{(Average)}$ of all data samples using the formula (5);

Step4: Select a data point x_i $(1 \le i \le n)$ and find another data $x_k, (1 \le k \le n \ and \ k \ne i)$ that's distance from x_i is less than $Dis_{(Average)}$. Then we check all the data points in this way, and count the number of such data that like x_k. We call these data as nearest neighbors of the data x_i.

Step5: For the data set, we look for these nearest neighbors of each data point and count the number of these nearest neighbors of each data point. Finally, we sort all data points in descending according to their number of nearest neighbors.

Step6: Select the data point with the largest number of nearest neighbors as the first initial clustering center, and select the data point sorted to the second as another initial cluster center. We seek remaining cluster center points in this way. If a data point x_j $(1 \le j \le n)$ sorted to P is one of nearest neighbors of these selected cluster centers, we can ignore x_j, and check x_z $(1 \le z \le n)$ sorted to P+1. If x_z is not one of nearest neighbors of these existing cluster centers, we are able to use x_z as an initial cluster center and continue to look for the remaining initial cluster centers, until we find all the initial cluster centers.

Step7: Calculate the distance of each data object to each cluster center by using formula (4). Then each data object is clustered in cluster that's cluster center is closest to the data object.

Step8: Calculate the average value of each cluster, and use the mean as new center of the cluster.

Step9: Repeat Step7 and Step8. The clustering process is finished until each cluster does not change or the objective function converges.

D. AE-Kmeans Architecture

The AE-Kmeans architecture combines an autoencoder and the improved K-means algorithm. Firstly, the AE-Kmeans architecture preprocess these original data samples, and then the introduced autoencoder extract the characteristics of these pretreated data by adjusting the weights through the optimization function and these connecting neurons [8]. Finally, we can use the improved K-means algorithm to cluster these new data samples. The process of the AE-Kmeans architecture is shown in Figure 2.

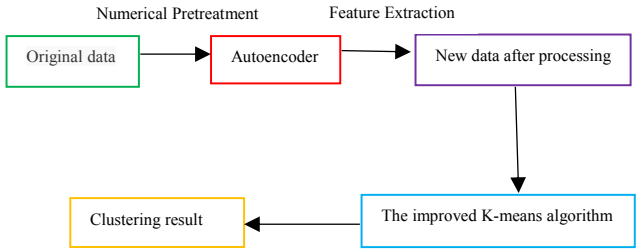

Figure 2. The process of the AE-Kmeans architecture

III. EXPERIMENT AND RESULT ANALYSIS

A. Experimental Data

This experiment selects the KDD CUP99 dataset. The dataset is a network traffic test data set established by Lincoln Laboratory of Massachusetts Institute of Technology, which is a relatively authoritative test data set [9]. KDD CUP99 data set mainly consists of 4898431 data records, of which there are 972781 normal records and 3925650 abnormal records. There are twenty-two kinds of network attacks types in the data set, which can be summarized into four categories: DOS, R2L, U2R and Probe [10].

B. Data Pretreatment

1) Numerical Pretreatment

There are two data types in the KDD CUP99 data set: one type is numeric and the other type is character. Because intrusion detection can only deal with numerical data, we need to numerically preprocess the nine kinds of attributes that are character type. For example, we numerically deal with protocol_type: tcp=1, udp=2, icmp=3. These remaining eight kinds of attributes are also done similar numerical preconditioning.

2) Reduce dimension and extract features of these data by using the autoencoder

KDD CUP99 data set contains forty-one kinds of fixed attributes and one class identifier, but there are many properties that possess little practical significance. Therefore, it is necessary to extract these important attributes of the KDD CUP99 dataset. Through the use of single-layer autoencoder, we have obtained these 18-dimensional data samples that are reduced dimension and extracted feature.

3) Standardization and normalization of data

After the numerical preprocessing and feature extraction, it is necessary to standardize and normalize these data in order to avoid the difference in the range of numerical fluctuation of different attributes. We use these formulas (6)(7)(8) to standardize these data and use formula (9) to normalize these data.

$$\bar{x_j} = \frac{1}{n}\sum_{i=1}^{n} x_{ij}, \tag{6}$$

$$\theta_j = \frac{1}{n}\sum_{i=1}^{n}\left(x_{ij} - \bar{x_j}\right) \tag{7}$$

$$x'_{ij} = \frac{x_{ij} - \bar{x_j}}{\theta_j} \tag{8}$$

$$x''_{ij} = \frac{x'_{ij} - min\{x'_{ij}\}}{max\{x'_{ij}\} - min\{x'_{ij}\}} \tag{9}$$

C. Experiment process and result analysis

1) Experiment one

DOS attack is not only the most attack, but also the most common attack. In experiments, we select these data records of DOS attack, and verify the accuracy of AE-Kmeans algorithm under the condition of selecting different clustering centers k. We introduce two evaluation criteria: detection rate and error-detection rate.

$$\text{detection rate} = \frac{The\ number\ of\ abnormal\ records\ detected}{The\ total\ number\ of\ abnormal\ records} \times 100\%$$

$$\text{error} - \text{detection rate} = \frac{The\ number\ of\ normal\ records\ that\ are\ detected\ as\ abnormal\ reconds}{The\ total\ number\ of\ normal\ records} \times 100\%$$

Through experiments, we obtained the comparison of detection rate between the AE-Kmeans algorithm and the traditional K-means algorithm，as shown in Figure 3. The contrast of error-detection rate is shown in Figure 4.

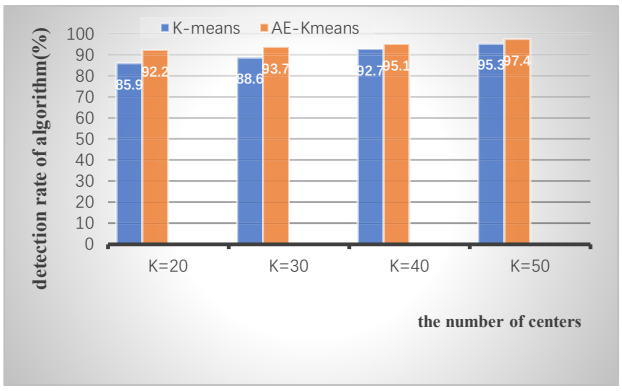

Figure 3. The detection rate of AE-Kmeans algorithm and K-means algorithm

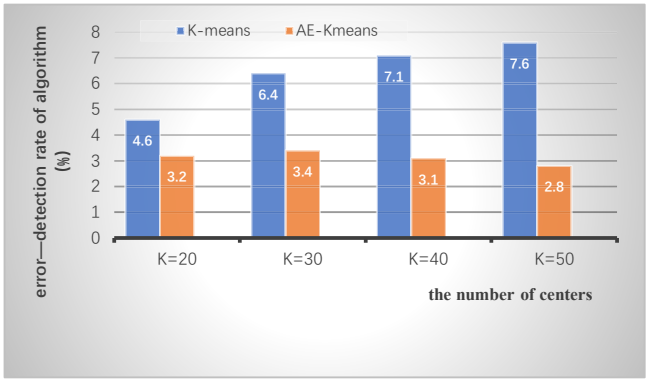

Figure 4. The error-detection rate of AE-Kmeans algorithm and K-means algorithm

It can be seen from the experiments that the AE-Kmeans has higher detection rate and lower error-detection rate than the traditional K-means algorithm in the case of selecting different cluster center k. At the same time, when the k value is 50, the detection rate and error-detection rate of AE-Kmeans algorithm are obviously better.

2) Experiment two

In the experiment, we use these 18-dimensional data samples obtained by the autoencoder and five types of attacks. We mainly select 3000 data samples, of which 2500 were used to train and generate the detection model, 500 as testing samples to verify the feasibility of the generated detection model. These types of attacks selected and the number of samples are shown in TABLE I.

TABLE I. THE DISTRIBUTE OF KDD CUP99 DATA SAMPLES

Attack Type	Total sample number	Number of training samples	Number of testing samples
neptune	1250	1000	250
smurf	350	300	50
satan	700	600	100
portsweep	450	400	50
warezclient	250	200	50

Through experiments, we obtain the comparison diagram of detection accuracy about the AE-Kmeans algorithm, K-means algorithm and the algorithm mentioned in literature [11], as shown in Figure 5. And we summarize the average detection accuracy and running time of these three algorithms as shows in TABLE II.

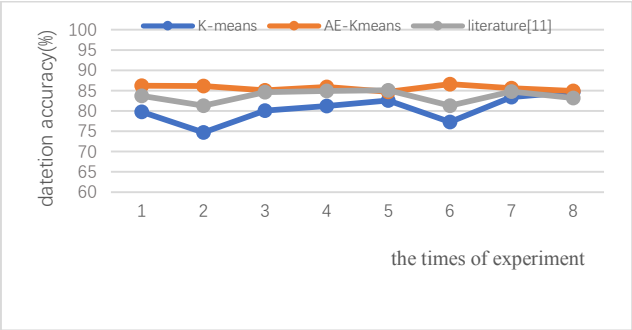

Figure 5. The detection accuracy of AE-Kmeans algorithm, K-means algorithm, the algorithm mentioned in literature [11]

TABLE II. THE RESULTS OF THREE DIFFERENT CLUSTERING ALGORITHMS

Algorithm	average detection accuracy (%)	running time(s)
K-Means	79.8	10.8
AE-Kmeans	87.2	7.2
Literature [11]	83.3	7.4

As can be seen from Figure 5, compared to the other two algorithms, the detection accuracy of AE-Kmeans algorithm are relatively higher, and it is relatively steady with the increase in the number of experiment. TABLE II shows that the average detection accuracy of the AE-Kmeans algorithm has reached 87.2%, meanwhile, it is 7.4% higher than that of the K-means algorithm, and it is also 3.9% higher than that of the algorithm mentioned in literature [11]. In addition, the average running time of the AE-Kmeans algorithm is 3.6s higher than the K-means algorithm, and it is 0.2s higher than the algorithm mentioned in the literature [11].

IV. CONCLUSION

This paper presents an AE-Kmeans architecture which combines the autoencoder with the improved K-means algorithm. The AE-Kmeans architecture realizes the dimension reduction and feature extraction of these original data by introducing autoencoder, and then uses the improved K-means algorithm to cluster these processed data. There are improvements on K-means algorithm mainly includes two aspects: the first aspect, the improved K-means algorithm uses the coefficient of variation to obtain the weight of each attribute, and then these weights are used in the Euclidean distance formula, resulting in the weighted Euclidean distance formula; the second aspect, in order to avoid the blindness of randomly selecting initial cluster centers, this paper introduces a new method that is adopted to select all initial cluster centers. The AE-Kmeans architecture compensates for some shortcomings of the K-means algorithm, and enhances the ability to deal with high-dimensional data, meanwhile, it increases the computational speed. When applied in network intrusion detection, the AE-Kmeans architecture improves the detection rate and reduces the error-detection rate.

ACKNOWLEDGMENT

This work was supported by Shandong Provincial Natural Science Foundation, China (No. ZR2014FQ021), and Key Research and Development Program of Shandong Province (No.2016GGX101039). The authors also sincerely wish to thank the instructors for several helpful suggestions.

REFERENCES

[1] Guo Chun. Research on Key Technologies of network intrusion detection based on Data Mining[D]. Beijing: Beijing University of Posts and Telecommunications, 2015.

[2] http://blog.csdn.net/xiaowei_cqu/article/details/7522368/.

[3] Li Han. Using a Dynamic K-means Algorithm to detect Anomaly Activities[C]//2011 Intelligence and Security. Sanya China: [s.n.],2011:1049-1052.

[4] Tong Xueqiao, Meng Fanrong, Wang Zhixiao. Optimization to k-means initial cluster centers[J]. Computer Engineering and Design, 2011,32(8):2718-2724.

[5] Xu Z. Methods for aggregating interval-valued intuitionistic fuzzy information and their application to decision making[J]. Control and Decision,2007,22(2):14-16.

[6] Li Mantian, You Jiali. K-mean clustering based user online behavior analysis in P2P networks[J]. Microcomputer Applications, 2009,30(11):24-27.

[7] Wu Haiyan. Research on semi supervised representation learning and classification learning based on autoencoder [D]. Chongqing: Chongqing University,2015.

[8] Li Senlin, Peng Xiaoyu, Huang Longhua. A clustering algorithm for single layer autoencoder [J]. Journal of Huaihua University, 2015, 34(11):40-41.

[9] Gao Ni, Gao Ling, He Yiyue, Wang Hai. A lightweight intrusion detection model based on Autoencoder network with feature reduction[J]. Acta Electronica Sinica.2017, 45(3):734-735.

[10] Tong Hongyan. Clustering analysis of network intrusion data [D]. Shenzhen: Shenzhen University, 2015.

[11] Li Yinhuan, Zhang Jian. Application of improved K-means algorithm in Intrusion Detection[J]. Computer technology and development, 2013,23(1):165-128.