# Comparison of Internet Traffic Identification on Machine Learning Methods

Lingjing Kong, Guowei Huang*, Keke Wu, Qi Tang, Suying Ye
Computer College
Shenzhen Institute of Information Technology
China
e-mail:lingjk11@gmail.com, gwhuang_sziit@163.com

***Abstract*—Traffic classification is the essential part in computer network. It can identify the traffic application so as to better manage the network, filter the insecure network flows and provide better network services. However, traditional traffic identification methods cannot work well when encounter opaque packets or more complex flows. Machine learning methods become the most efficient way to solve the problems existed in traditional ways, mainly including supervised learning and unsupervised learning. In this paper, two classic methods in supervised and unsupervised learning ways are applied to achieve the identification of abnormal traffic based on flow-level features. Besides, the comparison of training time, prediction time and the accuracy are given, which helps deep understand machine methods for traffic identification and design more efficient traffic identification solutions.**

***Keywords*-Traffic identification; unsupervised learning; supervised learning**

## I. INTRODUCTION

Computer network has a great influence on every aspect of modern life including communication, emails, news reading, commerce, education and many other fields hence network security presents a big challenge to current Internet. How to prevent computer network? In fact, traffic classification is the key component in abnormal detection, intrusion detection and firewalls[1][2][3], which helps identify the applications of Internet traffic and differentiate bad traffic from benign traffic. Besides, traffic classification can assist network administrator to better manage network and control the quality of services. However, conventional traffic classification methods including port-based method and DPI (Deep Packet Inspection) method are not able to work well because of the change of network environment. Machine learning (ML) methods have been utilized in traffic classification to solve new challenges and achieve good results, which mainly aims at training the classification model based on Internet traffic features.

There are two main machine learning methods: unsupervised machine learning and supervised machine learning. Unsupervised learning method is to find homogeneous groups with respect to their familiarities. Supervised learning methods utilize labeled dataset to learn a classifying model which is capable of indicating the classes of coming traffic and identify the categories. They can all be used to classify and identify Internet traffic, and have their own advantages and disadvantages. In this paper, two classic algorithms in unsupervised learning and supervised learning methods are introduced and adopted to establish the model for abnormal traffic identification using KDD'99 dataset. Finally the comparison and analysis will be shown in detail.

## II. RELATED WORK

The key idea of traffic application identification is to inspect the features of internet traffic and identify the application categories of network flows. Traditional methods commonly rely on the port numbers of TCP or UDP packets (Traffic Identification based Ports), or inspect the payload content to find out the signatures (Deep Packet Inspect, DPI). However, these methods have some defects. Ports can recognize the application running on two endpoints, so port numbers can be utilized to identify the application categories. Usually, one kind of the application is associated with a well-known port number registered in the Internet Assigned Numbers Authority(IANA)[4], e.g., SSL traffic is associated with port 443 [4]. The classifier compares the port numbers with the IANA list to identify protocols and flow application types. Though this method is fast and easily to realize, it has some defects. Many applications use dynamic ports to transfer data to the other end such as video stream traffic. It is also common that some applications may not use well-known port numbers. Furthermore, some applications hide their own well-known port numbers to avoid the block of the firewalls. All these above result in the failures of the method based on ports.

DPI is a method widely used in many companies which searches the specific bytes in the packet payload (usually called signatures) to identify flow applications. Compared with port-based method, DPI can much improve the accuracy of identification[5][6][7][8]. However, this method need keep pace with the change of protocols, and put a lot of effort into analyzing the structure of network packets. It is difficult to identify encrypted traffic when using this method. Furthermore, inspecting the content of packets may violate the privacy laws.

Nowadays, the statistical characteristics of network flow are used to identify the traffic application such as flow durations, packets size, arrival time *et, al.* This method can overcome the faults of the methods based ports and payload, so it becomes the important direction in this research area [9][10][11][12]. To better cope with large amounts of internet traffic and their behavior patterns, machine learning(ML) techniques are involved. ML techniques are

978-1-5386-6136-9/18/$31.00 ©2018 IEEE

firstly introduced in traffic identification in intrusion detection system in 1994[13], which mainly include supervised learning and unsupervised learning algorithms. Supervised learning is to establish the model with the guidance of the labeled dataset[14], and unsupervised learning is to look for similar groups without the guidance[15]. In this paper, we will use classic algorithms in unsupervised learning and supervised learning methods to establish the model for abnormal traffic identification, and discuss the differences between them.

## III. DATA PREROCESSING

### A. Feature Extraction

Features are extracted from network flows(also called sessions or connections) consisted of multiple packets with the same source IP, destination IP, source port, destination port and protocol. Features reflects the pattern of flow behaviors such as the duration, source bytes, destination bytes, the number of root accesses.

In KDD'99 dataset, 42 features are determined to distinguish bad traffic from normal traffic.

### B. Data Format Transformation

All extracted features should be firstly represented as numeric digits and then transformed to different formats for different classification algorithm. For K-means, the data format is transformed as CVS file format, while for SVM algorithm, in the training data, the first column is the label and then followed by "index: feature". In the test data, every column is represented as "index: feature" without labels.

### C. Data Normalization

Normalization is mapping the features to a new specific space through rescaling so that the mean equals 0 and the standard deviation from the mean equals 1.

Here, the approach we used in normalization is Min-Max scaling and the data will be usually scaled to [0,1] range. The equation of Min-Max scaling can be represented as:

$$X_{norm} = \frac{X - X_{\min}}{X_{\max} - X_{\min}}$$

With data normalization, the convergence rate and the accuracy of the model can be much improved.

## IV. K-MEANS AND SUPPORT VECTOR MACHINE(SVM) ALGORITHMS

### A. K-means Algorithm

K-means is a classic unsupervised algorithm to cluster the data with similar properties into the same groups, and separate data with dissimilar properties through distance measuring method such as Euclidean space. The core idea of K-means to find the centroids of the clusters. The centers of cluster has a great influence on clustering results, so how to find the centers is the key point in this algorithm. Certainly, the best centroids should be the ones that are the least close to others in clusters. Firstly, randomly choose some centroids, and assign the data to these centers. Then determine the new centroids in these clusters and assign the other data again. Through the repeated iteration, the best centroids will be found and the cluster with the most familiar objects will be formed. It can be well used to cluster the Internet traffic into disjoint groups based on similar flow features and identify the specific application of clusters.

Given $\{x^{(1)}, x^{(2)}, ..., x^{(m)}\}$, $k$ is the number of clusters, $c^{(i)}\left(i = 1, 2, ..., m\right)$ is the index of the cluster(from 1 to $k$ ) to which example $x^{(i)}$ is assigned and $\mu_k$ is the cluster centroid $k\left(\mu \in R^n\right)$ . K-means algorithm can be seen as the following:

Input:

$k, \{x^{(1)}, x^{(2)}, ..., x^{(m)}\}, x^{(i)} \in R$(drop 1 to $K$)

Begain:

Randomly initialize $K$ cluster centroids $\mu_1, \mu_2, ..., \mu_K \in R$

Repeat{

for $i = 1$ to $m$

$c^{(i)} :=$ *index*(from 1 to $K$) of cluster centroid closet to $x^{(i)}$

for $k = 1$ to $K$

$\mu_K :=$ mean of points assigned to cluster $K$

finish

K-means algorithm can classify the data to different disjoint classes, and then we can identify the application of classes based on the majority label of each cluster.

### B. SVM Algorithm

SVM is a supervised learning algorithm including training and testing phases. For training, each example contains a set of flow features and an application label. The target of SVM is to establish a model that is capable of predicting the test data to a application class. In fact, training model is to learn a mapping relationship between features $x_i\left(x \in X\right)$ and the specific application(label $y_i$, $y \in Y$ ).

The mapping relationship is infact a function $F = f\left(x_i\right) = h_\theta\left(x_i\right)$ , where $\theta$ is the weight matrix of the mapping function. To determine $F$ , we must obtain the $\theta$ matrix. The cost function $J$ is the deviation of hypothesis $h_\theta\left(x\right)$ and the real value $y$ .Through minimizing the cost function $J$ , we can get $\theta$ matrix and determine $F$ as shown in the following:

$$\min_\theta \sum_{i=1}^{m} \left[ y^{(i)} cost_1(\theta^T x^{(i)} + \left(1 - y^{(i)}\right) cost_0\left(\theta^T x^{(i)}\right) \right]$$

In the above formulation, $C$ is the penalty parameter. When $\theta^T x^{(i)} \geq 1$ , $y^{(i)} = 1$ ; When $\theta^T x^{(i)} \leq -1$ , $y^{(i)} = 0$ .

If the data can be linear classified, SVM is to look for a linear hyperplane to partition the data into binary classes with a maximum margin seen in Figure 1:

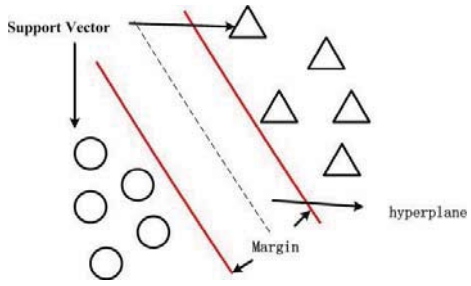


Figure 1. Linear classification

But in most cases, dataset is not linear but nonlinear seen in Figure 2.

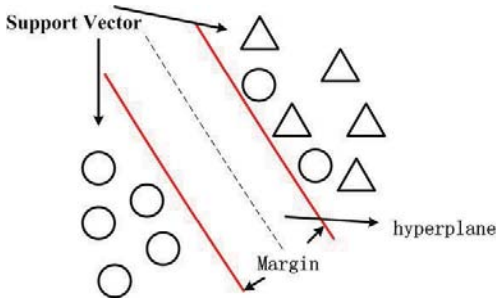


Figure 2. nonlinear classification

Kernels can be used to transform the initial features to a higher dimensional space so as to solve the nonlinear classifying problems. In SVM, four kernels are defined:

*1) Linear kernel: It is the case without kernel which is defined as* $K\left(x_i, x_j\right) = x_i^T x_j$.

*2) Polynomial kernel: It is defined as* $K\left(x_i, x_j\right) = (\gamma x_i^T x_j + r)^d$ *, where* $\gamma > 0$ .

*3) RBF(Radial basis function) kernel: It is defined as* $K(x\,i\,,x\,j\,) = \exp(-\gamma \,||\, x\,i - x\,j\,|\,|\,2\,)$ *, where* $\gamma > 0$ .

*4) Sigmoid kernel: It is defined as* $K\left(x_i, x_j\right) = tanh\left(\gamma x_i^T x_j + r\right)$.

In view of the experiments in [16], Linear Kernel and RBF kernel are ones with higher accuracy. So in this paper, we will give a comparison between K-means and SVM with linear and RBF kernels.

## V. MODEL TRAINING AND ANALYSIS

### A. Dataset

In this paper, the data from KDD Cup 1999[17] will be used to carry on the experiments. Each TCP connection is processed as connection record, and is labeled as normal or attack type.

In this dataset, there are four main types of attacks[17]:

*1) DOS: Denial of service.*

*2) R2L: Remote-to-Local.*

*3) U2R: unauthorized access to local superuser.*

*4) Probing: surveillance or others.*

We will use the reduced training dataset, that is the 10 percent of training dataset(494021). 24 specific attacks are concluded in the dataset, and 311029 data is used to be test.

### B. Implementations

The following experiments will be conducted on a server with 3.6GHZ CPU(8 cores) and 16GB RAM by K-means and SVM methods.

*1) Implementation on K-means*

First, we will train the model by k-means method by using Python[19] based on the algorithm introduced in Section 3.1. Through training phase, 39 cluster centers will be get to distinguish the labels of instances from test dataset. The training time, and predict time are also defined in the code, which will compare with the results by SVM method in the next part. Accuracy can directly judge a classifier and is another critical comparison aspect. It can be calculated as:

$$Accuracy = \frac{\sum_{i=1}^{m} TP_i}{\sum_{i=1}^{m} \left(TP_i + FN_i\right)}$$

In the formulation , *TP* represents the true positive and *FN* represents the false positive.

The steps of the experiment based on K-means algorithm mainly includes the following steps:

*a) Training data and test data format transformation.*

*b) Feature Scaling.*

*c) Training K-means model.*

*d) Test the model.*

The result of experiment shows that the training time is 111.436 seconds, prediction time is 0.255 seconds and the accuracy is 83.12%. Next, we will use the same dataset to train and test on SVM.

*2) Implementation on SVM*

Based on the study in[16], linear kernel and RBF kernel showed better performance. So in this part, we conducted the following experiments using the two kernels:

*a) Before scaling, linear kernel and RBF kernel are chosen to train and test.*

*b) After scaling, the above kernels are used to train and test to compare with the first experiment.*

Before scaling, the accuracy is only 72.6466% for linear kernel and 89.3569% for RBF kernel. The training time and prediction time are much longer than that after scaling, which is not recommended in real environment.

After scaling, the training time is to 39.847257 seconds, the prediction time is 55.034239 seconds and the accuracy is improved to 91.5381% for linear kernel; For RBF kernel, the training time and the prediction time are 80.991311 seconds and 168.577879 seconds separately, and the accuracy is about 91.4821%.

### C. Experiment Analysis

Through the experiments, it is not difficult to find that after scaling K-means method is slower than SVM in training phase, while much faster than SVM method in testing phases. But the accuracy of SVM is higher than K-

means algorithm in linear and RBF kernel. The comparison can be clearly seen in Figure 3 and Figure 4:

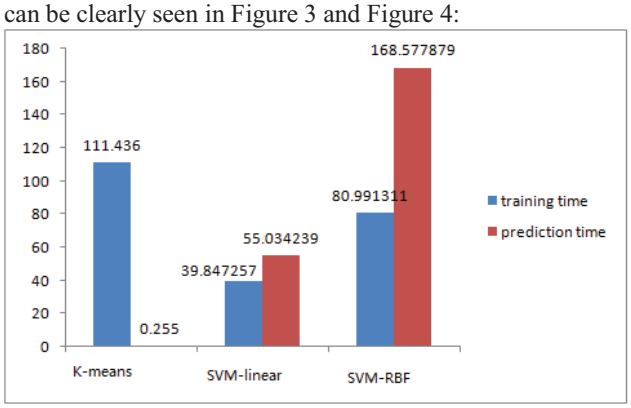


Figure 3. Comparsion of training time and prediction time

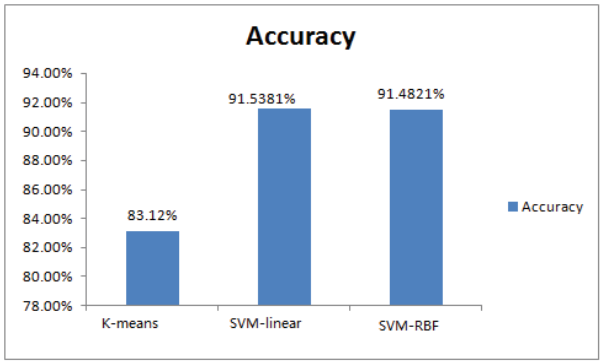


Figure 4. Comparsion of the accuracy

Though unsupervised learning method is slower in training the model, it is much fast to identify the application of the traffic. So it is more suitable for coarse-grained clustering or fast abnormal report, while supervised learning is better for conducting precise classification and identification. In some cases, the combination of these two approaches can better improve the efficiency and the accuracy of traffic identification.

## VI. Conclusion

In this paper, we introduced the important methods—unsupervised learning and supervised learning in traffic identification in which K-means and SVM are the classic algorithms. We utilized these two algorithms to train the classifier and identify the abnormal traffic based on KDD'99 dataset. Through the comparison, K-means method is slower in training the classifier and much faster in prediction, but less accurate. SVM is much slower to predict the specific attack of the test data, but the accuracy is higher. Besides, if using the default parameters *C* and -γ, linear kernel is better than RBF kernel in identification speed and the accuracy. In the future, we prefer to combine theses two methods so as to achieve the identification of Internet traffic more efficiently.

## Acknowledgment

This work was supported by Major Fundamental Research Project in the Science and Technology Plan of Shenzhen (Grant No. JCYJ20160527101106061, JCYJ20160307101532282, JCYJ20170817114239348 and JCYJ20170306095622684), Guangdong College Students' Science and Technology Innovation Cultivation Project(Grant No. pdjh2017b0724).

## References

[1] Paxson V. Bro: a system for detecting network intruders in real-time[J]. Computer networks, 1999, 31(23): 2435-2463.

[2] Bro intrusion detection system - Bro overview [OB/EL].http://bro-ids.org,2016.

[3] Roesch M. Snort: Lightweight Intrusion Detection for Networks[C]//LISA, 1999, 99(1): 229-238.

[4] Internet Assigned Numbers Authority (IANA). http://www.iana.org/assignments/port-numbers.

[5] 1T. Choi, C. Kim, S. Yoon, J. Park, B. Lee, H. Kim, and H. Chung. Content-aware internet application traffic measurement and analysis. In IEEE/IFIP NOMS, April 2004.

[6] P. Haffner, S. Sen, O. Spatscheck, and D. Wang. Acas: Automataed construction of applicatin signatures. In SIGCOMM MineNet Workshop, August 2005.

[7] Moore A W, Papagiannaki K. Toward the accurate identification of network applications[M]//Passive and Active Network Measurement, 2005: 41-54.

[8] Sen S, Spatscheck O, Wang D. Accurate, scalable in-network identification of p2p traffic using application signatures[C]//Proceedings of the 13th international conference on World Wide Web, 2004: 512-521.

[9] V. Paxson, "Empirically derived analytic models of wide-area TCP connections," IEEE/ACM Trans. Networking, vol. 2, no. 4, pp. 316–336, 1994.

[10] Bernaille L, Teixeira R, Salamatian K. Early application identification[C]//Proceedings of the 2006 ACM CoNEXT conference, 2006: 6.

[11] Crotti M, Dusi M, Gringoli F, etal. Traffic classification through simple statistical fingerprinting[J]. ACM SIGCOMM Computer Communication Review, 2007, 37(1): 5-16.

[12] Dainotti A, De Donato W, Pescape A, et al. Classification of network traffic via packet-level hidden Markov models[C]//IEEE on Global Telecommunications Conference, 2008: 1-5.

[13] Frank J. Artificial intelligence and intrusion detection: Current and future directions[C]//Proceedings of the 17th national computer security conference, 1994, 10: 1-12.

[14] Y. Reich and J. S. Fenves, "The formation and use of abstract concepts in design," in Fisher, D. H. and Pazzani, M. J. (editors), Concept Formation: Knowledge and Experience in Unsupervised Learning. Morgan Kaufmann, 1991.

[15] H. D. Fisher, J. M. Pazzani, and P. Langley, Concept Formation: Knowledge and Experience in Unsupervised Learning. Morgan Kaufmann, 1991.

[16] Lingjing kong, Guowei Huang, Keke Wu. Identification of Abnormal Network Traffic Using Support Vector Machine[C]//The 18th International Conference on Parallel and Distributed Computing, Applications and Technologies, 2017,12: 288-292.

[17] KDD Cup 1999 data[OB/EL]. http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html.

[18] Scikit-learn[OB/EL]. http://scikit-learn.org/stable/.