International Journal of Security and Its Applications
Vol.9, No.7 (2015), pp.153-162
http://dx.doi.org/10.14257/ijsia.2015.9.7.14

A Density-based Binary SVM Algorithm in the Cloud Security

Mingyuan Yu, Shuhang Huang, Qing Yu, Yan Wang, and Jiaquan Gao

College of Computer Science and Technology, Zhejiang University of Technology, Hangzhou 310023, China
yu_mingyuan@163.com, Shuhang_huang@qq.com, 1842622373@qq.com, 113772360@qq.com, gaojq@zjut.edu.cn

Abstract

In recent years, cloud computing is becoming popular in the field of information, however, the development of cloud computing have to face the problem of cloud security. Intrusion Deletion System (IDS) is one of the possible solutions to the problem of cloud security, but the correct rate of general application of the IDS is not very satisfactory, for this purpose we propose a density-based binary Support Vector Machine (SVM) method (D-BSVM). Its main idea is based on the density of each class in the data set, and gets a binary sequence of training, according to this sequence obtained binary SVM training model to predict the behavior of the system. Further, the method for calculating the density is the paralleled, thereby improving efficiency of overall system. Finally, we present experimental results, and by contrast our approach can improve the accuracy and detection rate of IDS.

Keywords: *Cloud computing, intrusion detection, Hadoop, SVM*

1. Introduction

Since 2006, the Google put forward the concept of "cloud computing" for the first time in the search engine assembly. In recent years, cloud computing is becoming one of hot topics in the field of current network information technology. Cloud computing is to a large number of network resources, hardware resources and computing resources together to form a huge pool of computing resources sharing, provides the resources needed for the user. Hadoop [1] is developed by the Apache foundation to large-scale distributed cluster parallel programming calculation of open source cloud computing platform framework. The framework allows the user to do not understand the distributed low-level details of the case, make full use of cluster's ability to deal with huge amounts of data to develop a distributed application, and the architecture is suitable for the distributed storage of cheap machines and distributed management, has the characteristics of high scalability, fault tolerance. Hadoop has gradually become the most popular application of cloud computing platform. But at present, cloud computing development is faced with many problems [2], which are the most important security issues. In recent years, Google, amazon and other cloud computing initiator out all kinds of safety accidents, more intensified the concerns of the people. For example, Google Gmail emails a global fault, amazon web server outage, VMare source stolen, and so on.

Intrusion Detection System (IDS) [3] is a kind of network security technology, which can real-time monitor of network transmission, alarms when in suspicious transmission or take the initiative response measures. Intrusion detection system based on the analysis of the network packet, detect suspicious activity, such as the network access and resource request without permission, found in whether someone is trying to attack the network or host from known attack types, and successfully defensive attack. Information collected through intrusion detection system, the network administrator or safety management personnel to take effective measures to reinforce its own system, so as to avoid more loss.

ISSN: 1738-9976 IJSIA

In order to protect the cloud environment from intrusion attack, the proposed paper launches an idea of federation defense in the cloud computing. Based on this concept, IDS system is deployed in each cloud computing region. These IDSs will detect the suspicious network behavior, by gathering network packets, generate intrusion prediction model training. While, support vector machine based on binary tree structure is the key technology intrusion detection. Literature [4] designs this SVM classification and the classification order solely relying on the class super-sphere volume and considers the class having largest super-sphere volume as the first class to separate from other classes. However, this design is unreasonable in IDS, because its premise is that the data set for each category are super-sphere, while the data sets in IDS are no rules. Therefore, we must design a method, which can identify data without rules.

In this article, a new intrusion detection method of binary support vector machine (SVM) with Hadoop is put forward, whose basic idea is, according to the density of the data set to set the priority classification, let the class of the most easy to separate, and generate the training decision tree, so as to improve the accuracy of classification model.

The rest of the paper is organized as follows. Section 2 introduces the structures of existing multi-class SVM and Hadoop. Classification strategy is presented in section 3. We present and discuss the experiment results in section 4. Section 5 concludes this paper.

2. Related Works

2.1. Multi-class SVM

Support vector machine [5] is a new learning machine based on statistical learning theory, which is a small sample statistical learning algorithm based on structural risk minimization principle (SRM) and Vapnik-Chervonenkis (VC) dimension conception. The traditional SVM is used to solve the problem of two categories, when using SVM in the question of multi-class classification, needs to be multi-class problem into two types of problems. There are mainly four methods belonging to this way: one against-all [6] method, one-against-one [7] method, Binary Tree Support Vector Machines (BTSVM) [8] method and Directed Acyclic Graph Support Vector Machines (DAGSVM) [9] method.

One-against-all method to class I sample, and all except the class as negative samples, between the two types of samples training vector machine, this method a total of k classification support vector machine (SVM) is constructed. So this method has the advantage of fast training speed, but the unbalance of training sample points influences the accuracy.

One-against-one method is to build out a classification hyper plane between every two types of samples, all K class samples a total of K can be structured K*(K - 1) /2 classification hyper plane. Obviously when the value of K is much big, we need to build more classification hyper plane.

DAGSVM method is for that one-to-one SVM is inseparable, error classification, and that is a kind of classification is relatively perfect. This way also needs to construct K*(K-1)/2 sub-classifiers where K is the number of classes, and then combines these sub-classifiers into a directed acyclic graph.

Multi-class SVM based on binary tree structure first divides all classes into two sub-sets at the root node, then considering the sub-set as the root node of a new binary tree, divides the sub-sets into two new sub-sets, and so continue until each leaf only contains one class. Figure 1 shows the structure of BTSVM, in which the class number is K and the classification order is 1, 2, and 3…K.

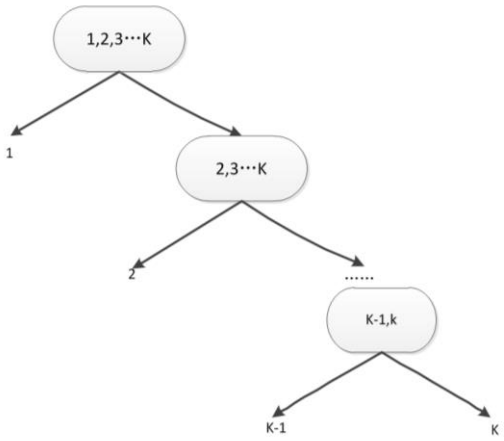


Figure 1. The Structure of Binary Tree SVM

Therefore, it can be seen that binary SVM has simple structure, and only needs n-1 sub-classifiers where n is the number of classes, which is less than one-against-one and DAGSVM method.

2.2. Hadoop and MapReduce

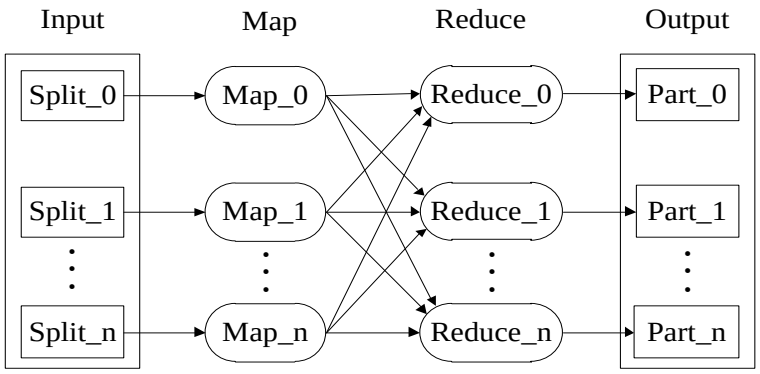


Figure 2. MapReduce Implementation Process

Hadoop is a computational framework proposed by Apache for large-scale distributed data processing, in a reliable, efficient and scalable way to process, use of Hadoop, users can not understand the underlying architecture on the basis on the development of distributed applications. Hadoop is mainly composed of two parts, MapReduce parallel computing framework and the Hadoop Distributed File System (HDFS). MapReduce is an open source implementation of Google MapReduce; a MapReduce program contains three main functions, namely Map function, Reduce functions and Main functions.

The execution of MapReduce [10] functions has shown in Figure 2. First the input data were divided into a number of blocks; in each Map function of the system execute a block data. After the Map function execution is completed, the output of the Map function Key-Value passed to the Reduce function, Reduce function appropriate merge the Key-Value on the Map function outputs, each Reduce function to get a part of the results. Finally, the output of the function Reduce summarized to obtain the final output.

3. The Binary Tree SVM Algorithm based on Density (D-BTSVM)

So can be seen from the second quarter, the closer to the root node of the SVM classification, the more the performance of the classification effect is most affected, therefore in the process of classification, the class which is the most easy to separate out should first points out, that means the split on the top of the binary tree. Different binary tree structure will have different classification model, each class segmentation regions are different, and resulting in the classification model of promotion ability will also be different.

Algorithm 1. intrusion detection algorithm process based on the density of the binary SVM classificationr

Input: train data set D contains of N classes, radius ε and Minimum threshold minP.
Output: classification order of N classes.
1: select any one of the objects Xi of a class Ni, and determines whether the point Xi is the core object
2: if Xi is the core object, find all the density-reachable points in the ε-neighborhood.
3: if Xi is not, select another point of class Ni, until it is a core object.
4: repeat 1-3, until traversal all the points of class Ni.
5: In the collection of core objects, delete duplicate points, and computed the density of class Ni.
6: select another class, repeat 1-5, until all densities of classes is computed.
7: finally, output classification order of N classes.

In order to explain the generation of binary tree based on the density [11], firstly introduced several concepts about density. Assuming the data set S = {S1, S2, S3…, Sn}, Si is one of the points.

a. ε-neighborhood: The radius ε region of given object, is called ε- neighborhood of the object.
b. Core Point: If ε- neighborhood of an object comprises at least a minimum number MinPts (Minimum threshold) objects, the object is called the core object.
c. Density-reachable: Suppose S_i, S_j are in the set S of data points, and Si is in the ε-neighborhood of Sj, Sj is a core point, and then Si is a density-reachable point of Sj.
d. Density: The density of data set S is P, M is the number of density-reachable point of data set S, and N is the number of all points of S.

P=M/N.

As Algorithm 1 shows, intrusion detection algorithm process is based on the density of the binary SVM.

From the Algorithm 1, we can see that when the sample points of each class has a large size in the train data set, the efficiency of the algorithm will become very poor, which means that the time to calculate the density become longer. Therefore, it is necessary to divided into the algorithm which should be executed in parallel, and this algorithm is modified to one based on MapReduce, which will greatly improve the efficiency of the algorithm, as Algorithm 2.

Algorithm 2. Parallelization D-BTSVM (PD-SVM)

Input: train data set D contains of N classes, radius ε and Minimum threshold minP.
Output: classification order of N classes.
Class Mapper
Method map ()


```
   for m=1,2,3…K   //K is dimension of data set
   //Point data sets obtained and temporarily stored in an array
        temp[m]=valueOf(value.toString().split(","));
   Point p = new Point(temp);  //Generate a point object
   // Different types, the different point of the object is added to the list, and
outputs it to reduce to processing
   If(temp[m]==1)
     output.write(1, p);
   If(temp[m]==1)
     output.write(2, p);
  …


Class Reducer
   method reduce ()
     resultList=new ArrayList<List<Point>>(); //List uses to store result
     List = getList(key);
     for(Point  item: List)
      if( isKeyPoint(pointsList, p, e, minp))    // Determine whether the core object
         resultList.add(tmpLst);    //If yes, add to resultList.
     mergeList(resultList.get(i), resultList.get(j));  //Merge the same point object
     P=  (double)resultList.size()/pointsList.size();        //P is density of key
corresponding class
  Return P;
```

According to the introduction to the MapReduce in 2.2, we design a MapReduce job to calculate density values. The training data set as input, then the output should be the density of each category descending sort. In the Map function, main task is that point of data set divided by class ,each line of the data set as an input, the corresponding point of the object and the class which is this object belonged to as a (Key, Value) pair. The task of reduce function is to calculation.

4. Experiments

We have used the KDD CUP 99 DATA as train data of IDS, the selection of this dataset is due to its standardization, content richness and it helps to evaluate results with existing researches in the area of intrusion detection. This data set contains two kinds of normal and abnormal behavior, and abnormal behavior also mainly includes four attacks. They are DOS, R2L, U2R and PROBE. DOS is denial-of-service, such as ping-of-death, flood, and smurf and so on. R2L is unauthorized access from a remote machine to a local machine, for example guessing password. U2R is unauthorized access to local super user privileges by a local unprivileged user, such as buffer overflow attacks. PROBE is meaning of surveillance and probing, such as port-scan, ping-sweep.

Table 1. 10%of Data Distribution

Class type	NORMAL	PROBE	DOS	U2R	R2L
Training data	97278	4107	391458	52	1126
Testing data	60593	4166	229853	228	16189

KDD99 data sets consists of a total of 5,000,000 records, it provides a subset of 10% of the training and testing subsets, sample class distribution of training data in table1. Each row records contain the 41 properties, which can be divided into four categories, they are

the basic features of TCP connections (9), content features of TCP (13), based on the statistical characteristics of network traffic time (9), host-based network traffic statistics feature (10).

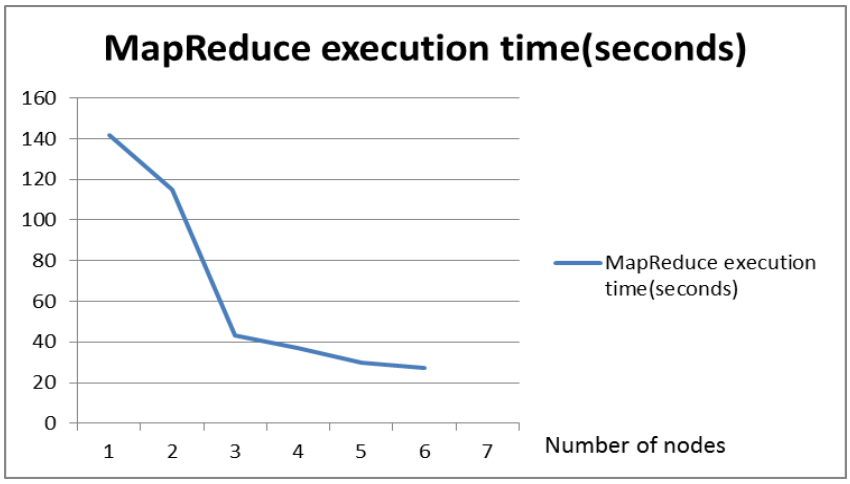


Figure 2. Results of PD-SVM

Within each virtual machine, we have the same basic configuration software environment, as follows:

- Operating System: Ubuntu 12.04.
- JDK version: jdk1.7.0_05.
- Hadoop version: 0.20.2.
- SVM tool: Weka package.

KDD99 data sets of a total of 5,000,000 records form, it provides a subset of 10% of the training, and we use the latter as this experiment the training data set. First, the density of five categories is calculated in the stand-alone environment (Windows 7 Operating System 64 bits, 4 GB RAM, Intel Core i5), while it takes 65.991seconds and the order of descending density is Normal>R2L>PROBE>U2R>DOS.

In case the result of the same experiment, we also do experiments with PD-SVM, and the experimental results are shown in figure 2. We could find that the more virtual machine nodes and less execution time of the algorithm, the reason is that the parallel processing to reduce the workload. When the larger data set, we use MapReduce can save more time to computing density, and this will give IDS provides more time to detect unknown behavior.

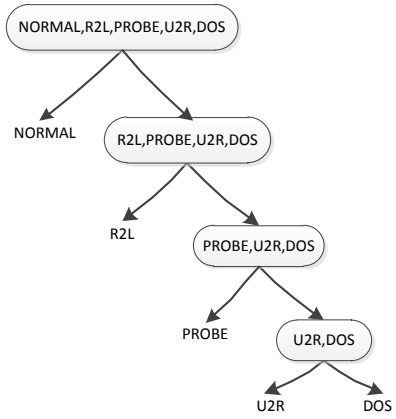


Figure 3. SVM Binary Tree Structure of the Training Data Set

Now according to the classification order which was computed before, we can get the training data set of binary SVM structure, as show in figure 3. We train the data set by Weka package, and we will firstly separate the NORMAL class from the all data set. Then, we will remove the NORMAL class, and just the remain data to separate the R2L from others. Next we remove the R2L class, and separate the PROBE from the other two classes. Finally, as long as the rest do a two classification, and we get the result as Table 2 and Table 3.

Table 2. Accuracy of Classification Results

Class type	NORMAL	R2L	PROBE	U2R	DOS
Our method	99.5%	95.5%	99.8%	94.2%	100%
One-against-all	99.3%	85.2%	91.9%	50%	99.7%
One-against-one	96.1%	92%	91.8%	81.5%	97.7%

As can be seen from Table 2 and Table 3, the proposed density-based binary tree SVM classification detection method has a better overall performance; the total accuracy of our method is higher than one-against-all and one-against-one method. We tried to generate a random sequence, generating a binary SVM training model, the correct rate of the first class is less than 50%, so we do not give results. Therefore, we can conclude that to build a right binary tree structure is necessary. From the experimental results above, we can see that our method of generating the binary tree structure can get a higher accuracy of the results.

Table 3. Detected the Number of Records and the Records of the Data Set

Class type	NORMAL	R2L	PROBE	U2R	DOS
Total	97278	1126	4107	52	391458
Our method	96634	1075	4099	49	391458
One-against-all	96597	959	3774	26	390283
One-against-one	93484	1035	3770	42	382454

Now we've got this binary SVM classification model, and then we will analyze the predictive power of the model. We will use the above mentioned 10% of test data sets do predict experimental. The results obtained are shown in Table 4. After experimental comparison, it is clear that the proposed prediction model we obtained, there is a better detection rate and accuracy.

Table 4. Test Results

Class type	Total testing samples	Detection Rate		
		Our method	One-against-all	One-against-one
NORMAL	60593	97.3%	97.4%	91.6%
R2L	16189	94.5%	82.9%	90.7%
PROBE	4166	95.6%	89%	90.1%
U2R	228	94.1%	48.3%	79.8%
DOS	229853	98.6%	97.1%	95.4%

From the above experiments can be concluded that there is our proposed MP-SVM application in IDS, and has a better detection rate, and can also predict the majority of

attacks. Therefore, we propose a method that can effectively protect the security of cloud computing

5. Conclusions and Future Work

In this article, we propose a new method for calculating binary SVM classification order, which is based on the density of data, and put in the method implemented in MapReduce, which can significantly reduce the time to calculate the density in the case of the same order as obtained. In the experiment, we can see that the method has a good accuracy. In the future, we will discuss distributed SVM for better efficiency.

Acknowledgements

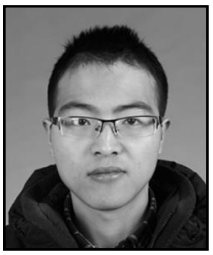
This work was partly supported by the National Natural Foundation of China (NSFC) project under grant No. 60703002, Science and Technology Plan Project of Zhejiang Province (2014C33077), the Natural Science Foundation of Zhejiang Province (No.Y1110768), and Dr start-up fund research of Zhejiang University of technology (No: 119001229), and the School Foundation of Zhejiang University of Technology(No.119002415).

References

[1] R. C. Taylor, "An overview of the Hadoop/MapReduce/HBase framework and its current applications in bioinformatics", BMC Bioinformatics, vol. 11, **(2010)**.
[2] F. B. Shaikh and S. Haider, "Security threats in cloud computing. 6th Internet Technology and Secured Transactions (ICITST)", **(2011)** Dec 11-14; Abu Dhabi, United Arab Emirates.
[3] O. Al-Jarrah and A. Arafat, "Network Intrusion Detection System using attack behavior classification", 5th International Conference on Information and Communication Systems (ICICS), **(2014)** April 1-3; Irbid, Jordan.
[4] F. Tang, Z. Wang and M. Chen, "On Multi-class Classification Methods for Support Vector Machines", Control and Decision, vol. 20, **(2005)**.
[5] Z. Qi, Y. Tian and Y. Shi, "Robust twin support vector machine for pattern classification", Pattern Recognition, vol. 46, **(2013).**
[6] K. Tatsumi and T. Tanino, "Nonlinear extension of multiobjective multiclass support vector machine based on the one-against-all method", The 2011 International Joint Conference on Neural Networks (IJCNN), **(2011)** July 31-August 5; San Jose, CA.
[7] X. Yang, Q. Yu, L. He and T. Guo, "The one-against-all partition based binary tree support vector machine algorithms for multi-class classification", Neurocomputing, vol. 113, no. 3, **(2013).**
[8] S. Gang, Z. Wang and M. Wang, "A New Multi-Classification Method Based on Binary Tree Support Vector Machine", Innovative Computing Information and Control **(2008)** June 18-20; Dalian, Liaoning.
[9] J. Martínez, C. Iglesiasb, J. M. Matíasc, J. Taboadab and M. Araú, "Solving the slate tile classification problem using a DAGSVM multiclassification algorithm based on SVM binary classifiers with a one-versus-all approach", Applied Mathematics and Computation, vol. 230, (**2014**).
[10] K. Shim, "MapReduce algorithms for big data analysis", Databases in Networked Information Systems, Lecture Notes in Computer Science, vol. 7813**, (2013)**.
[11] Q. Wu, "DSDBSCAN: A novel clustering algorithm based on double sampling for DBSCAN", Journal of Computational Information Systems, vol. 10, no. 1, **(2014)**.

Authors

Mingyuan Yu, he received his Ph.D. degree in Spatial Information Science & Technology from Huazhong University of Science and Technology, China, in 2009. He is currently an associate professor of the College of Computer Science and Technology at the Zhejiang University of Technology in Hangzhou, China. His current research interests are in the areas of trusted computing, service computing and big data analysis. He is a member of the Chinese Computer Federation and a member of the ACM.



Shuhang Huang, he received the B.S. degree in Electronic Information Science & Technology from Wenzhou University, Wenzhou, China, in 2012. Currently he is undertaking a M.E. course in Computer Technology at Zhejiang University of Technology in Hangzhou, China. His current research interests are cryptography, information security and user authentication.

Qing Yu, he received her B.E. in Software Engineering from Nanyang Institute of Technology in Henan, China, in 2013. She is currently undertaking a M.E. course in Computer Technology at Zhejiang University of Technology in Hangzhou, China. Her current research interests are big data analysis and cloud computing.

Yan Wang, she received her B.E. in Computer Science & Technology from Xi`an University, Xi'an, China, in 2014. She is currently undertaking a M.E. course in Computer Technology at Zhejiang University of Technology in Hangzhou, China. Her current research interests are big data application and cloud computing.

Jiaquan Gao, he received the PhD degree in Computer Science from the Institute of Software, Chinese Academy of Sciences in 2002. He is currently an associate professor of the College of Computer Science and Technology at the Zhejiang University of Technology in Hangzhou, China. He respectively worked as a visiting scholar at the McGill University, Canada, from September 2007 to September 2008 and the Georgia Institute of Technology, US, from December 2011 to May 2012. His current research interests include high-performance computing (HPC), parallel algorithms and computational intelligence.