Effective Framework of J48 Algorithm using Semi-Supervised Approach for Intrusion Detection

Sharmila Wagh
Professor
MES College of Engineering
Pune, India

Anagha Khati
Student
MES College of Engineering
Pune, India

Auzita Irani
Student
MES College of Engineering
Pune, India

Naba Inamdar
Student
MES College of Engineering
Pune, India

Rashmi Soni
Student
MES College of Engineering
Pune, India

ABSTRACT

Network security is a very important aspect for internet enabled systems. As the internet keeps developing the number of security attacks as well as their severity has shown a significant increase. The Intrusion Detection System (IDS) plays a very important role in discovering anomalies and attacks in the network. The aim of an intrusion detection system is to identify those entities that attempt to destabilize security controls that have been put in place. The field of machine learning is rapidly gaining more attention in the development of these intrusion detection systems. Machine learning techniques can be broadly classified into three broad categories: Supervised, Un-supervised and semi-supervised. The supervised learning method displays good classification accuracy for those attacks that are aready known to us. But this method requires a large amount of training data.The availability of labelled data is not only time consuming but also very expensive. The evolving field of semi-supervised learning offers a promising direction for supplementary research. Hence, in this paper we propose a semi-supervised approach for a pattern based IDS to improve performance and to reduce the false alarm rate. The experimentation is performed on KDD CUP99 dataset and we use the J48 Algorithm in order to implement the semi-supervised learning.

General Terms

Network Security, Pattern Based Security, J48 Algorithm, Semi-Supervised Approach

Keywords

Network security, KDD CUP99, intrusion detection, semi-supervised learning, supervised learning, J48 Algorithm.

1. INTRODUCTION

In this modern world intrusion occurs in the blink of an eye. Intruders use the modified version of command and erase their footprints in audit and log files. IDS is a device or software application that monitors network or system activities for malicious activities or policy violations and produces reports to a management station. Successful intrusion detection systems intellectually differentiate between intrusive and non-intrusive records. IDS was first introduced by James Anderson in the year 1980. Most existing systems have security breaches that make them easily vulnerable. Substantial research has been undertaken on intrusion detection technology as it is still considered immature and is not a perfect tool against intrusion. IDS draws many research interests and we have attempted to add to this large research effort. Our project is an attempt to reduce false alarm rate for Intrusion Detection System by using Machine Learning algorithm. We aim to design IDS by using machine learning which can meet the demands of Reducing False Alarm Rate with higher detection rate. In order to do this we use the decision tree classifier and J48 algorithm. Many types of IDS already exist in the world which provides assistance at different stages of project development. Even though many types of IDS exist, a problem that is commonly observed is that of high False Alarm Rate. Our software is able to assist the developers in this department greatly along with the individual stage support.

This paper is organized as follows: Section 1 provides related work based on the semi supervised learning. Section 2 insights problem definition. Section 3 gives detail study attacks. Section 4 describes J48 algorithm selection and implementation. Section 5 overviews our proposed approach for semi supervised learning method for intrusion detection followed by conclusion and discussion for future research in section 6 and section 7.

2. RELATED WORK

Semi-supervised learning methods use unlabeled data to either modify or reprioritize premises obtained by using only labeled data. Of late, learning with labeled and unlabeled data, also known as semi- supervised learning has drawn much attention. It aims to attain good classification performance with the assistance of unlabelled data in the presence of the small sample problem, and a few promising results have been reported. Therefore, instead of training the model with only labeled data, we incorporate the unlabelled data before active learning starts. G.V.Nadiammai, S.Krishnaveni, M. Hemalatha [6] have been referred to in order to understand the use different data mining techniques in order to implement an intrusion detection system. In this paper, they are provided with a summary of the current research directions in detecting such attacks using collaborative intrusion detection systems (CIDSs). Sandip Sonawane, Shailendra Pardeshi and Ganesh Prasad [4] are presented with three types of intrusion detection based on the source of detection – host based, network based and hybrid intrusion detection and also focuses on intrusion detection techniques that is, misuse detection and anomaly detection techniques, supervised and unsupervised based learning based on the various approaches.

Yuanqing Li et.al [10] proposes self training algorithm .This algorithm is used for semi supervised learning, which is an iterative algorithm.

Table 1: Feature categories in KDDCup99 dataset

Categories	Features
TCP basic features(1~9)	duration, protocol type, service, flag, src_bytes, dst_bytes, land, wrong fragment, urgent
TCP content features(10~ 22)	hot,num_failed_logins, logged_in,num_compromised, root_shell, su_attempted,num_root, num_file_creations, num_shells,num_access_files, num_outbound_cmds,is_hot_login, is_guest_login
TCP Traffic features(23~ 31)	count, srv_count, serror_rate, srv_seror_rate,rerror_rate, rv_rerror_rate, same_srv_rate,diff_srv_rate, srv_diff_host_rate
Host-Based Network Traffic(32~41)	dst_host_count, dst_host_srv_count,dst_host_same_srv_rate, dst_host_diff_srv_rate,dst_host_same_src_port_rate, dst_host_srv_diff_host_rate, dst_host_serror_rate,dst_host_srv_serror_rate , dst_host_srv_serror_rate,dst_host_rerror_rate , dst_host_srv_rerror_rate

3. PROBLEM DEFINITION

There are various types of learning methods such as supervised, unsupervised and semi-supervised learning. The supervised learning refers to training the system with labeled data and un-supervised learning refers to training the system with unlabeled data. While semi-supervised learning represents training the system with labeled as well as unlabeled data. The supervised learning method exhibits good classification accuracy for known attacks. But the main problem with it is that it requires a large amount of training data. In the real world, the availability of labeled data is time consuming and costly. An emerging field of semi- supervised learning offers a promising direction for further research. When the unlabeled data is used in combination with a small amount of labeled data it can produce substantial improvement in learning precision. Hence we are making use of semi-supervised learning. We will employ the use of the KDD CUP 1999 data set for our project. It was devised at MIT's Lincoln Lab and developed for IDS evaluations by DARPA. It represents the activities at US Air Force local area network (LAN), which have normal traffic and malicious activities that were injected in the datasets. In spite of several drawbacks, it has served as a dependable benchmark data set for many researches on network based intrusion detection algorithms.

4. TYPES OF ATTACKS

Various types of attacks are possible on a system and to construct an efficient intrusion detection system we have to take into account all of these different types of attacks. A DOS attack is a denial of service attack. A denial-of-service (DoS) or distributed denial-of-service (DDoS) attack can be described as an attempt to make a machine or network resource unavailable to its intended or authorized users. U2R attack refers to an unauthorized access to the local super user (root) privileges. The R2L, that is remote to local, attack is one in which there is unauthorized access to the local super user (root) privileges. The fourth type of attack is the probe attack.

4.1 Classification of Attacks

The KDD Cup 99 data set contains 23 different attack types. Their names are shown in Table II and its features are grouped as follows: Basic Features, Traffic Features and Content Features.

1. Basic features contain all the attributes of the TCP/IP connection and lead to delay in detection.

2. Traffic features are evaluated according to the window interval and two features as same host and same service.

(i) Same host feature: It examines the number of connections in the past 2 seconds from the same destination host. The probability of connections will be done in a specific time interval.

(ii) Same service feature: It inspects the number of connections in a particular time interval that possesses same service.

3. Content features: Dos & probe attack have frequent intrusion sequential patterns compared to R2L & U2R. These two attacks include many connections to several hosts at a particular time period whereas R2L and U2R achieve only a single connection. In order to detect these types of attacks, domain knowledge is important to access the data portion of the TCP packets.

Table II: Classification of attacks under four groups

Denial of Service	Back, land, neptune, pod, smurf, teardrop
Probes	Satan, ipsweep, nmap, port sweep
Remote to local	ftp_write, imap, guess_passwd, phf, spy, warezclient, multihop, warezmaster
User to root	Buffer_overflow, load module, Perl, root kit

4. J48 ALGORITHM SELECTION AND IMPLEMENTATION

Decision tree induction is the learning of decision trees from class labeled training tuples. A decision tree is a flow chart like tree structure where each internal node denotes a test on an attribute. Each branch represents an outcome of the test and each leaf node holds a class label. Weka is an open source data-mining tool that supports only supervised and clustering algorithm. One can integrate java code implementations of semi-supervised algorithms in Weka.

The J48 algorithm comes under the decision tree classifier. We have selected this algorithm as it has the highest accuracy. One of the main benefits of the J48 classifier is that is relatively quick to train, and should finish almost immediately on a small data set. In this algorithm classification model is formed according to the training set and later the unlabeled set can be labeled according to the model. Our implementation will be capable of various options like accepting the training and testing set. The tree can be printed in pruned as well as un-pruned form.

The supervised J48 Algorithm is as follows:

Generate_decision_tree: Generates a decision tree from the training tuples of data partition D.

Input:

1. D is a set of training tuples
2. Attribute list
3. Attribute selection method

Output: A Decision tree

Method

```
create a node N;
if tuple in D are all of the same class, C, then
return N as a leaf node labeled with the class C;
if attribute_list is empty then
return N as a leaf node labeled with majority class in D;
apply Attribute_selection_method(D,attribute_list) to find the "best" splitting_criterion;
label node N with splitting_criterion;
if splitting_attribute is discrete-valued and multiway splits allowed then
attribute_list <- attribute_list – splitting_attribute;
for each outcome j of splitting_criterion
let Dj be the set of data tuples in D satisfying outcome j;
if Dj is empty then
attach a leaf labeled with the majority class in D to node N;
return N;
```

5. ARCHITECTURE OF THE SEMI-SUPERVISED INTRUSION DETECTION SYSTEM

The architecture of our system is as follows. The first module will be a training module. We will provide labeled datasets to the system. The next module will be a testing module. In this module unlabeled data will be used in order to test the trained system. The mixed module will contain both labeled as well as unlabeled data. An algorithm will be used for classification. Entropy calculation will be used to select the most confident data and a semi-supervised module will be created.. In this way both labeled as well as unlabeled data will be used in order to create the semi-supervised learning system.

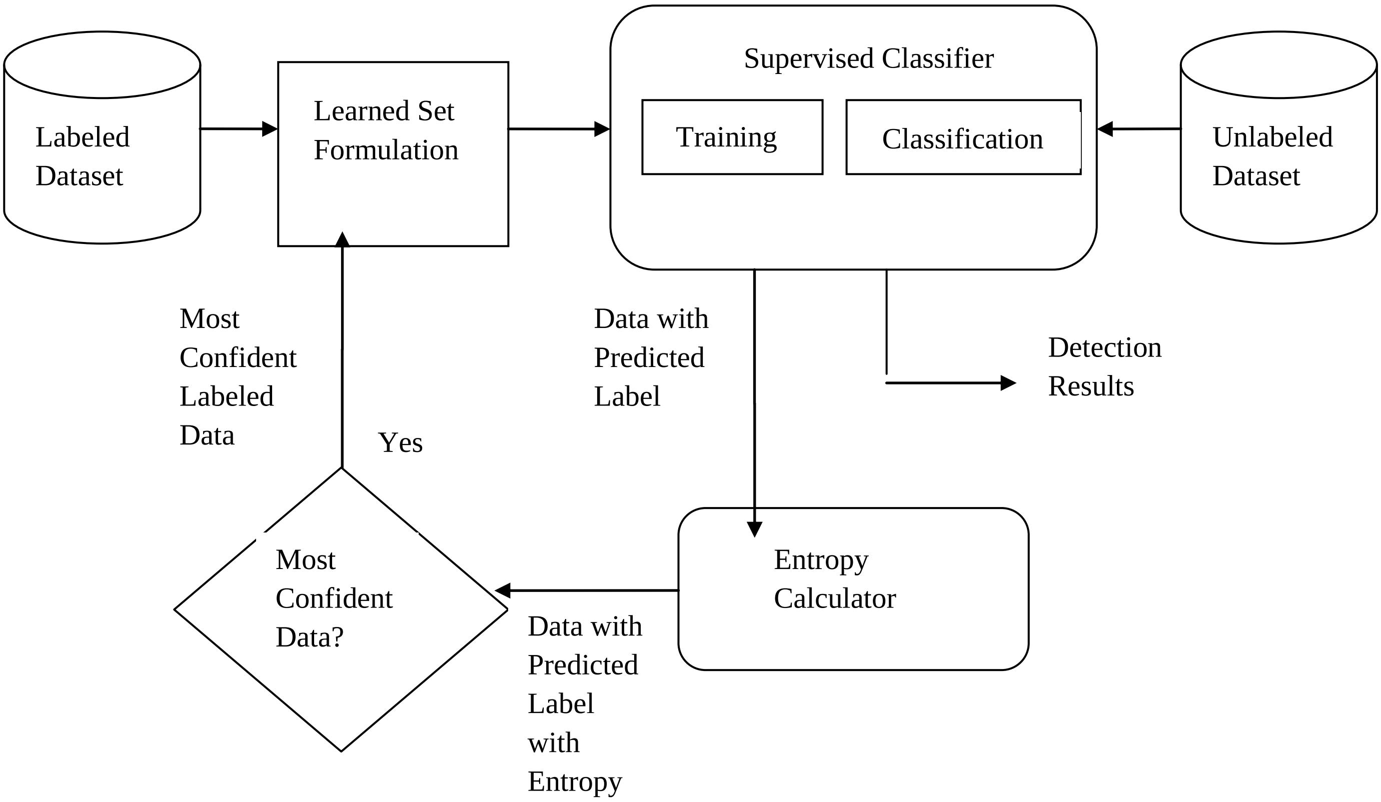

Fig 1: Architecture of proposed semi-supervised intrusion detection system

5.1 Semi-supervised J48 algorithm

The supervised algorithm is inefficient as it requires a large amount of training data hence making it very costly and causes it to become time inefficient and hence we converted it into semi-supervised algorithm using the following steps:

Step 1: Train f from labeled data

Step 2: Predict on x ϵ unlabeled data.

Step 3: Add (x; f (x)) to labeled

Step 4: Repeat

The variations in self training are:

1. Add a few most confident (x, f (x)) to labeled data

2. Add all (x, f (x)) to labeled data

3. Add all (x, f (x)) to labeled data, weight each by confidence.

5.2 Entropy Calculation

Entropy needs to be calculated for each data tuple to identify the most confident data that will be used further. The confident data that has been obtained is then combined with train set and filtering is done. Entropy of a tuple D is given by following formula.

$$E(D) = -\sum_{i=1}^{m} Pi \ \log_2(Pi) \qquad 1$$

Where d is a data packet, m is the number of attributes and P_i is the probability of the i^{th} attribute. Based on the entropies of each packet, the most confident data will be chosen which will be decided according to a threshold value. This data will then be appended to the training set and therefore enhances it.

6. CONCLUSION

We have proposed an algorithm for semi-supervised learning using decision tree classifier and the J48 Algorithm. The strength of our proposed algorithm lies in its ability to improve the performance of any given base classifier in the presence of unlabeled samples. Our algorithm will be capable of giving higher accuracy than already available algorithms and will also ensure that there will be a lower false alarm rate.

7. FUTURE USE

This system can be put to use as a real-time application in networking. We can also use the system for different data-sets format. It also finds applications in detecting new attacks. We can use more machine learning algorithms to improve accuracy. Based on this work, a recurring scan cloud can be created to quickly monitor which services and programs run on a machine allowing for an even more precise rule set. In this way, a sensor will not capture all the traffic on a network instead, the client machines will monitor their own traffic. The clients will automatically report data to centralized monitoring station.

8. REFERENCES

[1] Sharmila Kishor Wagh, Vinod K Pachghare and Satish R Kolhe. "Survey on Intrusion Detection System using Machine Learning Techniques." *International Journal of Computer Applications* 78(16): 30-37, September 2013.

Published by Foundation of Computer Science, New York, USA

[2] Sharmila Kishor Wagh, Gaurav Nilwarna, S. R. Kolhe, “A Comprehensive Analysis and Study in Intrusion Detection System Using KNN Algorithm”, the 6th multidisciplinary workshop on Artificial Intelligence 2012 (MIWAI 2012), organized at Ho Chi Minh city, Vietnam.

[3] Abd Jalil, K, , Shah Alam, Kamarudin, M.H., Masrek, M.N., “Comparison of Machine Learning algorithms performance in detecting network intrusion “,Published in: Networking and Information Technology (ICNIT), 2010 International Conference, Date of Conference: 11-12 June 2010, Print ISBN: 978-1-4244-7579-7

[4] Sandip Sonawane, Shailendra Pardeshi, Ganesh Prasad, “A survey on intrusion detection techniques*”*, World journal of science and technology, vol. 2, pp.127-133, 21st April 2012.

[5] Dorothy E. Denning, “An Intrusion-Detection Model,” IEEE transactions on software engineering, vol. SE-13, no. 2, pp.222-232, Feb 1987

[6] G. V. Nadiammai, S.Krishnaveni, M. Hemalatha, “A Comprehensive Analysis and study in Intrusion Detection System using Data Mining Techniques”, International Journal of Computer Applications, Volume 35 - Number 8, Year of Publication: 2011

[7] Phurivit Sangkatsanee, Naruemon Wattanapongsakorn, Chalermpol Charnsripinyo **, “**Practical real-time intrusion detection using machine learning approaches”, Computer Communications 01/2011; 34:2227-2235. DOI: 10.1016/j.comcom.2011.07.001

[8] Charles Elkan, “Results of the KDD’99 Classifier Learning”, SIGKDD Explorations 1(2): 63-64 (2000)

[9] S Stolfo et al, “The third international knowledge discovery and data mining tools competition” [online]. Available:http://kdd.ics.uci.eduidatabases/kddCup99/kddCup99.html, 2002 .

[10] Yuanqing Li *, Cuntai Guan, Huiqi Li, Zhengyang Chin, “A self-training semi-supervised SVM algorithm and its application in an EEG-based brain computer interface speller system”, Pattern Recognition Letters 29 (2008) 1285–1294