# 비인가된 사용자로부터의 네트워크 침입 탐지 및 분류를 위한 기계 학습 접근법

Linh Van Ma, Akm Ashiquzzaman, 김상우, 이동수, 김진술
전남대학교, 전자컴퓨터공학

# Machine Learning Approach for Detecting and Classifying Network Intrusions from Unauthorized Users

Linh Van Ma, Akm Ashiquzzaman, Sang Woo Kim, Dongsu Lee, Jinsul Kim
School of Electronics and Computer Engineering, Chonnam National University

## Abstract

In this article, we present a research on detecting and classifying network intrusions from network connection information which is gotten from users. We apply three machine learning algorithms which are multiperceptron layer, decision tree, and vector support machine. We perform the three algorithms and compare result to find the most appropriate algorithm for a network connection. As a result, we can apply the research outcome to secure a network, such as protecting information of a video streaming/conferencing session. The result can be used to protect information for communication between users and a system if they are at risk region such as disaster area where it is usually abandoned.

## 1. Introduction

An intrusion detection system (IDS) monitors the network traffic looking for suspicious activity, which could represent an attack or unauthorized access. Traditionally, networking systems were designed to detect known attacks but cannot identify unknown threats. The intrusion detector learning task is to build a predictive model (i.e. a classifier) capable of distinguishing between various kind of attacks, such as DOS (Denial-of-service attack), R2L (Remote to Local) attack, U2R (User to Root) attack, etc.

In this paper, we classify the type of a network connection into six classes and find rules to deduce whether the connection is a type of attack or not. Attacks were assigned with real values in new field called xAttack: DOS (1), U2R (2), R2L (3), Probe (4), Normal (5), Unknown (6).

## 2. Related Works

In this research, we involve three data mining techniques. First, multilayer perceptron [1] is a type of neutral artificial network. It consists of a minimum of three layers which are input, hidden and an output layer. Each neutron of a node is a neutron which applies a nonlinear activate function such as sigmoid function, max-pool function. Those functions assist the neutron network classifying objects into groups upon a conditioning dataset. In another approach of classification, decision tree [2] is a tree model which divides objects into groups based on various condition, such as probability and entropy. Lastly, support vector machine [3] is a discriminative method in classification. Its general idea is using hyper-plane in multi-dimensional space to put objects in different groups.

## 3. Dataset Description

NSLKDD-Dataset which is an original dataset with slight modification to include attack categories, e.g. DOS, U2R as having done with the original KDD99 dataset (The Third International Knowledge Discovery and Data Mining Tools Competition). NSL-KDD is a data set suggested to solve some of the inherent problems of the KDD'99 data set. Number of features are forty-one which can be divided into six parts {Basic features of individual TCP connections, Content features within a connection suggested by domain knowledge, Traffic features computed using a two-second time window, Traffic features computed in a destination host, Service feature, Flag feature}, and one more xAttack feature. Number of instances are 148,604 instances.

## 4. Experiment

We utilize Waikato Environment for Knowledge Analysis (WEKA) which is a suite of machine learning software written in Java, to solve the above-defined problem.

We first encode discrete features to numerical values, such as logged_in {0,1}, root_shell {0,1}, su_attempted {0,1}, is_hot_login {0,1}, is_guest_login {0,1}, and land {0,1}. The feature protocol_type has three protocols {'tcp','udp', 'icmp'} is encoded as {1,2,3}. The service and flag features are encoded as shown in Table 5 and Table 6 respectively.

Secondly, we run three separate classification methods (Multiperceptron layer, Decision Tree, and Support Vector Machine) and compare achieved results. We divide the dataset into the training set with 126017 instances (85%) and 22587 instances for the testing set (15%). We perform multilayer perceptron method four times with different layers and epoch. we utilize an implemented algorithm in WEKA, REPTree for Decision Tree, and Sequential Minimal Optimization (SMO) for Support Vector Machine.

Finally, we apply an association rules algorithm to determine whether a connection is DoS attack or normal. More details, we try to find rules from the dataset, which lead to decide whether a request is DoS attack or normal. We first reduce feature dimension for detecting a type of connection request (DoS or normal) by considering fewer features as shown in Table 1. We extract some features to reduce complexity when applying an association rule algorithm since it would increase computation time if a number of features is large. In addition, we consider features which assist to find a network connection which is a type of attack or not such as features shown in Table 4 (traffic features computed in destination host).

Table 1: Selected features for determining DoS and normal connection

| Connection Type | Selected Features |
|---|---|
| DoS<br>(13 features) | Protocol type, service, flag, src_bytes, dst_bytes, count, srv_count, serror_rate, srv_serror_rate, dst_host_count, dst_host_srv_count, dst_host_serror_rate, dst_host_srv_serror_rate. |
| Normal<br>(14 features) | Protocol_type, service, flag, src_bytes, dst_bytes, logged_in, count, srv_count, same_srv_rate, srv_diff_host_rate, dst_host_count, dst_host_srv_count, dst_host_same_srv_rate, dst_host_same_src_port_rate. |

We utilize an open source Frequent Patterns and Association Rule Miner (FPARM) which is published on Github [4]. It is an implementation of Apriori algorithm for frequent item set generation and association rule generation. In the experience, we set confidence equal to 0.999 for DoS, and confidence equal to 0.98 for the normal case.

With six classes of networking attacks, we obtain the comparison result of the three methods, which is shown in Table 2. The result shows that SMO has the highest percentage of correctly classified instances though it is the most time-consuming method. In oppose, MLP has the lowest percentage of correctly classified instances with four separated runs. For one hidden layer, it is even worse when we increase the number of the epoch from three to one hundred. It is the result of overtraining the dataset when the time to build the model increases from nine to 287 seconds. Therefore, the result would be much better if we appropriately increase the number of neutrons.

The REPTree seats in the middle with the result nearly approaches the highest correctly classified instances of SMO which is about 75%. Its performance is even better than SMO with 8 seconds.

Table 2: Comparison result of three different classification methods

| Classification Method | Time to build model | Correctly Classified | Note |
|---|---|---|---|
| MultilayerPerceptron | 8.96 seconds | 73.1801 % | One hidden layer (10), 3 epochs |
| MultilayerPerceptron | 286.45 seconds | 71.4235 % | One hidden layer (10), 100 epochs |
| MultilayerPerceptron | 4.89 seconds | 68.8684 % | Two hidden layers (5,3), 3 epochs |
| MultilayerPerceptron | 147.74 seconds | 71.2106 % | hidden layers (5,3), 100 epochs |
| REPTree | 7.84 seconds | 75.3582 % | Decision Tree |
| SMO | 449.46 seconds | 76.4007 % | Support Vector Machine |

Regarding association rules, the program finds 4915 rules which lead to DoS attack with accuracy more than 99.90%. We also obtain 8893 rules for normal cases with confidence higher than 0.98.

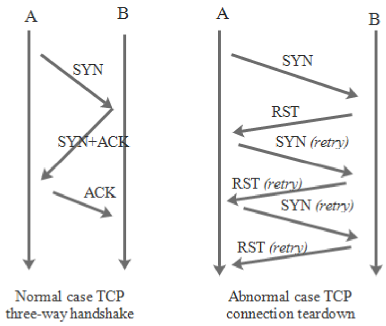
A
B
SYN
SYN+ACK
ACK
Normal case TCP three-way handshake
A
B
SYN
RST
SYN (retry)
RST (retry)
SYN (retry)
RST (retry)
Abnormal case TCP connection teardown

Figure 1: TCP handshake and TCP connection teardown

As shown in Table 3, the first rule can be explained as if service is http_8001 (using HTTP protocol via port 8001) and a number of connections to the same service as the current connection in the past two seconds for destination host, equals 255, then the connection is considered as a normal connection. The second rule is, if a user logged in successfully and a number of connections to the same service as the current connection in the past two seconds for destination host, equals 255, then it is also a normal connection.

As shown in Table 4, the first rule is that, if the sum of connections to the same destination IP address equals 255 with flag RSTOS0 along with the percentage of connections that have “SYN” errors, equals 1. It is determined as DoS attack. Recall that “RSTOS0” describes “Originator sent a SYN followed by an RST (TCP/IP reset (RST)), we never saw a SYN-ACK from the responder”. The detail of the connection is shown in the right of Figure 1 followed by the TCP handshake on the left.

Table 3: Association rules consider as normal network connection

| Rule | Attack Type & Confidence |
|---|---|
| service=25, dst_host_srv_count=255, (38924) | xAttack=5, (77053) conf(0.98080873) |
| logged_in=1, dst_host_srv_count=255, (37511) | xAttack=5, (77053) conf(0.9850977) |
| srv_diff_host_rate=0, logged_in=1, dst_host_srv_count=255, (21501) | xAttack=5, (77053) conf(0.9829775) |
| protocol_type=1, service=25, dst_host_srv_count=255, (38924) | xAttack=5, (77053) conf(0.98080873) |
| logged_in=1, dst_host_srv_count=255, dst_host_same_srv_rate=1, (37511) | xAttack=5, (77053) conf(0.9850977) |
| service=25, logged_in=1, dst_host_srv_count=255, (37507) | xAttack=5, (77053) conf(0.9852028) |

Table 4: Association rules consider as DoS attack

| Rule | Attack Type & Confidence |
|---|---|
| dst_host_count=255, flag=4, serror_rate=1, (33490) | xAttack=1, (53383) conf(0.99955213) |
| dst_host_count=255, flag=4, dst_host_serror_rate=1, (33763) | xAttack=1, (53383) conf(0.9998519) |
| protocol_type=1, dst_host_count = 255, flag = 4, dst_host_serror_rate = 1, (33763) | xAttack=1, (53383) conf(0.9998519) |
| protocol_type=1, dst_host_count=255, flag=4, serror_rate=1, (33490) | xAttack=1, (53383) conf(0.99955213) |
| src_bytes=0, dst_host_count=255, flag=4, dst_host_serror_rate=1, (33763) | xAttack=1, (53383) conf(0.9998519) |
| src_bytes=0, dst_host_count=255, flag=4, serror_rate=1, (33490) | xAttack=1, (53383) conf(0.99955213) |

## 5. Conclusion

In this article, we researched the problem of detecting network intrusion employing three data mining classification methods (multilayer perceptron, decision tree, and vector support machine). Besides, we applied the Apriori algorithm to find rules which address network attack (DoS) or normal. The result shows that the classification methods can correctly classify 75% of total instances. We can also find rules to determine whether a network connection is DoS attack or normal. Regarding the dataset, several researchers have claimed the NSL-KDD dataset is old, unrealistic and should not be used. This is facts. However, it is still a base dataset for research performance comparison.

## Acknowledgement

This research was supported by the IT R&D program of MSIT(Ministry of Science and ICT), Korea / NIPA(National IT Industry Promotion Agency) 12221-14-1001, Next Generation Network Computing Platform Testbed. Besides, this research was supported by the MSIT(Ministry of Science and ICT), Korea, under the ITRC(Information Technology Research Center) support program(IITP-2018-2016-0-00314) supervised by the IITP(Institute for Information & communications Technology Promotion). Furthermore, this research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science, and Technology (MEST)(Grant No. NRF-2017R1D1A1B03034429).

## References

[1] Gardner, Matt W., and S. R. Dorling. "Artificial neural networks (the multilayer perceptron)—a review of applications in the atmospheric sciences." Atmospheric environment, Vol. 32, No. 14-15, pp. 2627-2636, 1998.

[2] Safavian, S. Rasoul, and David Landgrebe. "A survey of decision tree classifier methodology." IEEE transactions on systems, man, and cybernetics Vol.21, No.3, pp. 660-674, 1991.

[3] Hearst, Marti A., Susan T. Dumais, Edgar Osuna, John Platt, and Bernhard Scholkopf. "Support vector machines." IEEE Intelligent Systems and their applications, Vol.13, No.4, pp. 18-28, 1998.

[4] Implementation of Apriori algorithm, "https://github.com/kevalmorabia97/FPARM-Frequent-Patterns-and-Association-Rule-Miner," Accessed: November, 10, 2018.