Intelligent Network Intrusion Detection

Yangxuan Wu

A thesis submitted for the degree of Master of Network Security and Management

Supervisor: Dr. Yongning Tang
School of Information System
Illinois State University

December 2020

Abstract：With more and more complex environment and services of modern networks，individuals and businesses face an exponential increase in the number of cyber attacks. Building secure applications, systems and networks is a major challenge we face today. To mitigate the impact of these threats, researchers have proposed many solutions for anomaly detection. Among them, detection based on network anomaly points is the main idea to find network attacks. The purpose of this project is: Try to identify network attacks by using machine learning methods; Try to study the law of the characteristic values of anomaly points and find out the typical characteristics of network attacks. Try to meature the performance of machine learning algorithm for different network attacks; Try to explain the performance difference from the principle of machine learning algorithm. In this context, it is attempted to find the network datasets containing anomaly indicators in the real network, and classify them according to different attack categories by labels. The random forest classification algorithm is used to select the characteristics of the datasets. Nine different machine learning algorithms are used in the application, and good performance is obtained. Machine learning algorithms and Perceptron accuracy were as follows :Naive Bayes 78%, QDA 31%, Random Forest 95%, ID3 95%, AdaBoost 95%, MLP 84%, K neighbor 97%, SVM 88%, Perceptron 74%.

Inderx terms: Machine Learning, Network Security, machine learning algorithm, anomaly detection, Naive Bayes, QDA, Random Forest, ID3, AdaBoost, MLP, KNN, SVM, Perceptron

Table of Contents

Ⅰ. Introduction:

Hacking incidents are increasing day by day as tech nology rolls out, a large number of hacking a distributed denial of service (DDoS) attack happened in hackers events report every year, which began in 2007. On June 17, 2008, Amazon started receiving some authenticated request from multiple users in one of its location. The requests began to increase significantly causing the servers slow down. On Jan 2013, European Network and Information Security Agency (ENISA) reported that Dropbox was attacked by DDoS and suffered a substantial loss of service for more than 15 hours affecting all users across the globe. Facebook was hit by suspected distributed denial of service attack on Sept 28,2014. Attackers are not only launching flooding and probing attacks but also spreading malware files in the form of virus, worm, spams to exploit the vulnerabilities present in existing software, causing a threat to the sensitive information of users stored on machines. Cisco Annual Security report mentioned that spam related to the Boston Marathon bombing comprised 40% of all spam messages delivered worldwide on April 17, 2013. On a recent survey done by Cisco in 2017, Trojan was classified as one of the top five malware which is used to gain initial access to the user' s computers and organizational networks. In the face of so many cyber attacks, traditional detection methods are no longer effective. For example, more than half of Internet use today is encrypted using SSL/TLS(Secure Sockets Layer/Transport Layer Security) protocols, and this rate is increasing. Signature-based methods cannot effectively handle this type of data because they cannot observe the contents of an encrypted Internet stream. Hence, security in such a complex technological environment is a big challenge and needs to be tackled intelligently.

The purpose of this study is to use the algorithm of supervised learning to detect network attacks by using the attack categories and the data sets with existing labels, and to explore the data characteristics of network attacks and the performance of the algorithm in identifying network attacks in network data.

Ⅱ.Related Works：

The application of machine learning to intrusion detection has been studied. And put forward specific contributions that make our work different from others. Agrawal and Agrawal[39] studied intrusion detection using data mining technology for anomaly detection. They classified the anomaly detection methods according to three factors: clustering based method, classification based method and mixed method K - means. The clustering based method describes the k-MEOID, EM clustering and outlier detection algorithms. Naive Bayesian algorithm, genetic algorithm, neural network, support vector machine and other classification methods have been described. Hybrid methods describe combinations of machine learning methods. They provide a brief comparison of papers using a holistic approach.

Haq et al. [40] investigated the application of machine learning technology in intrusion detection. They broadly divided these techniques into three categories: supervised learning, unsupervised learning and intensive learning. In supervised learning, the classifier is trained on the tagged datasets. Unsupervised learning is used when we do not have a tagged data set.In reinforcement learning, domain experts can tag untagged instances. They provide brief descriptions of various single classifiers and integration algorithms, and provide reference papers using machine learning intrusion detection without giving any critical analysis or observations. Ahmed et al. [22] provided a survey on network anomaly detection methods. Based on KDD 99 data set [12], attacks can be divided into four types :DoS, probe, U2R and R2L. Each category points to a specific type of exception. However, those network attacks are limited today. They provide a discussion of different types of machine learning approaches, namely category-based, clustering, statistics-based and information-theoretic approaches. The application of various machine learning methods in intrusion detection distinguishes between normal instances and abnormal instances. The problems of various network intrusion detection data sets are briefly summarized. However, their investigation lacks a detailed and in-depth description and analysis of the various machine-learning-based solutions available. Nor do the authors provide a future direction for machine learning algorithms. Buczak and Guven[41] discussed machine learning and data mining techniques for intrusion detection. Their survey describes the application of machine learning and data mining techniques to misuse and anomaly detection. They illustrate the difference between machine learning (ML) and data mining (DM), and point out that ML is an older brother of DM because they both use the same methods to classify or discover knowledge of data, so they use the term ML/DM to describe the algorithm under study.In their investigation, they describe various methods and associate them with misuse, anomaly, and hybrid detection techniques. The time complexity of the algorithm is also described.The KDD 99 and DARPA primary use data sets that they used were precursors to this paper's data set, which makes the comparison relevant to this project. However, some researchers use NetFlow and Tcpdump data sets. They respectively recommend which ML/DM method is suitable for misuse and anomaly detection.

Ⅲ. Contributions：

This article attempts to gather different methods, strategies, and steps on how and when to use ML techniques for feature extraction and attack classification. It will examine the ML solution from the monitoring phase to implementation. The primary purpose is to provide comprehensive guidance to practitioners in the field who intend to use ML technology to detect cyber attacks in network traffic. In this sense, this paper focuses on the steps of feature extraction in 14 attack data sets, and analyzes some feature values that can represent attack types, namely specific network traffic or indicators.

In short, the main contributions are summarized as follows.

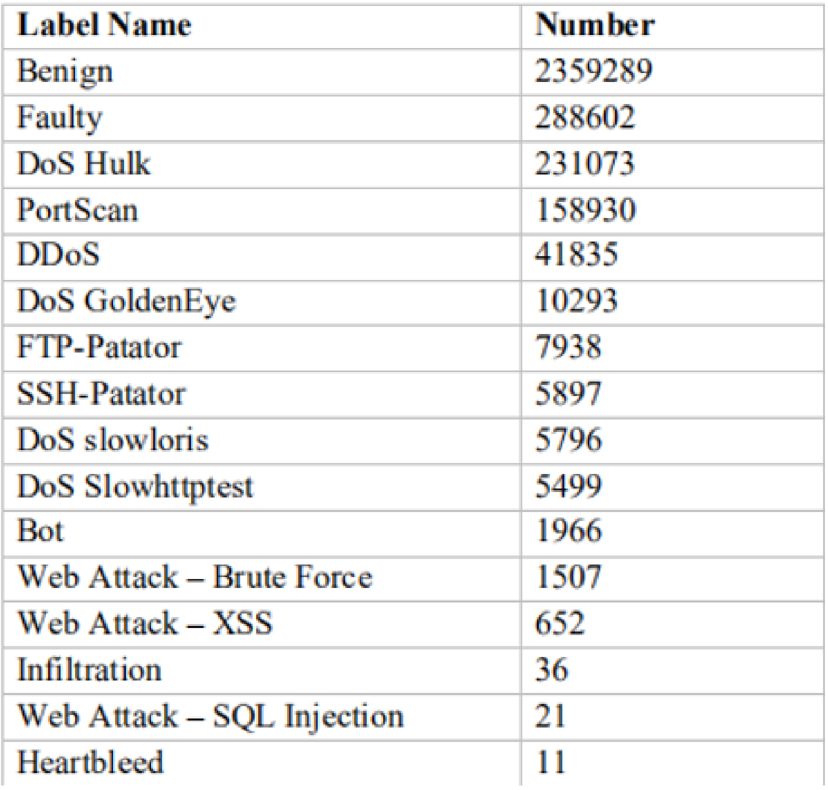
- It provides a comprehensive workflow to understand how to calculate the weighting of features in a dataset.
- It explains attack classification based on the nine machine learning algorithms used and through the ML technology based on tag values.
- Try to find the relevance from the principle of the algorithm by extracting the eigenvalues.
- It studies the each step of the workflow correlated with the efforts the found in the literature.
- It provides a set of paths that current approaches follow based on the Workflow defined.
- Finally, a general overview of feature extraction methods is given, and the future development direction is given according to the previous results.

Finally, it is worth noting that the main difference between this systematic review and other related work is the methodology proposed.The general procedure for applying ML technology in the field of Internet traffic classification is the paper of organization review.

Ⅳ.Organization： The remainder of this paper is organized as follows. The Ⅴ section briefly introduces the network of the highest frequency of 14 species in cyber attacks and show how these attacks work in network data. The Ⅵ section 9 of the most commonly used kind of machine learning algorithms, they were applied in the dataset. In accordance with the guidelines of the previous section, the Ⅶ section introduces the solution of the original data preprocessing. The Ⅷ part introduces some used to reduce or choose the method of feature extraction, report the method of feature extraction and feature extraction of importance weights, and find the characteristic and the rule of network attack. The Ⅸ section presents the scheme of algorithm evaluation, and analyze the results. The last part describes the prospect of the future.

Ⅴ. Types of Attacks in The Dataset:

All data in the dataset are tagged with 15 labels. One (Benign) of these tags represents normal network movements while the other 14 represent attacks. The benign record, formed using Mail services, SSH, FTP, HTTP, and HTTPS protocols represent a non-harmful / normal data stream on the network, created by simulating real user data. The names and numbers of these labels can be seen in Figure1 and Figure 2.



Label Name	Number
Benign	2359289
Faulty	288602
DoS Hulk	231073
PortScan	158930
DDoS	41835
DoS GoldenEye	10293
FTP-Patator	7938
SSH-Patator	5897
DoS slowloris	5796
DoS Slowhttptest	5499
Bot	1966
Web Attack – Brute Force	1507
Web Attack – XSS	652
Infiltration	36
Web Attack – SQL Injection	21
Heartbleed	11

Figure 1

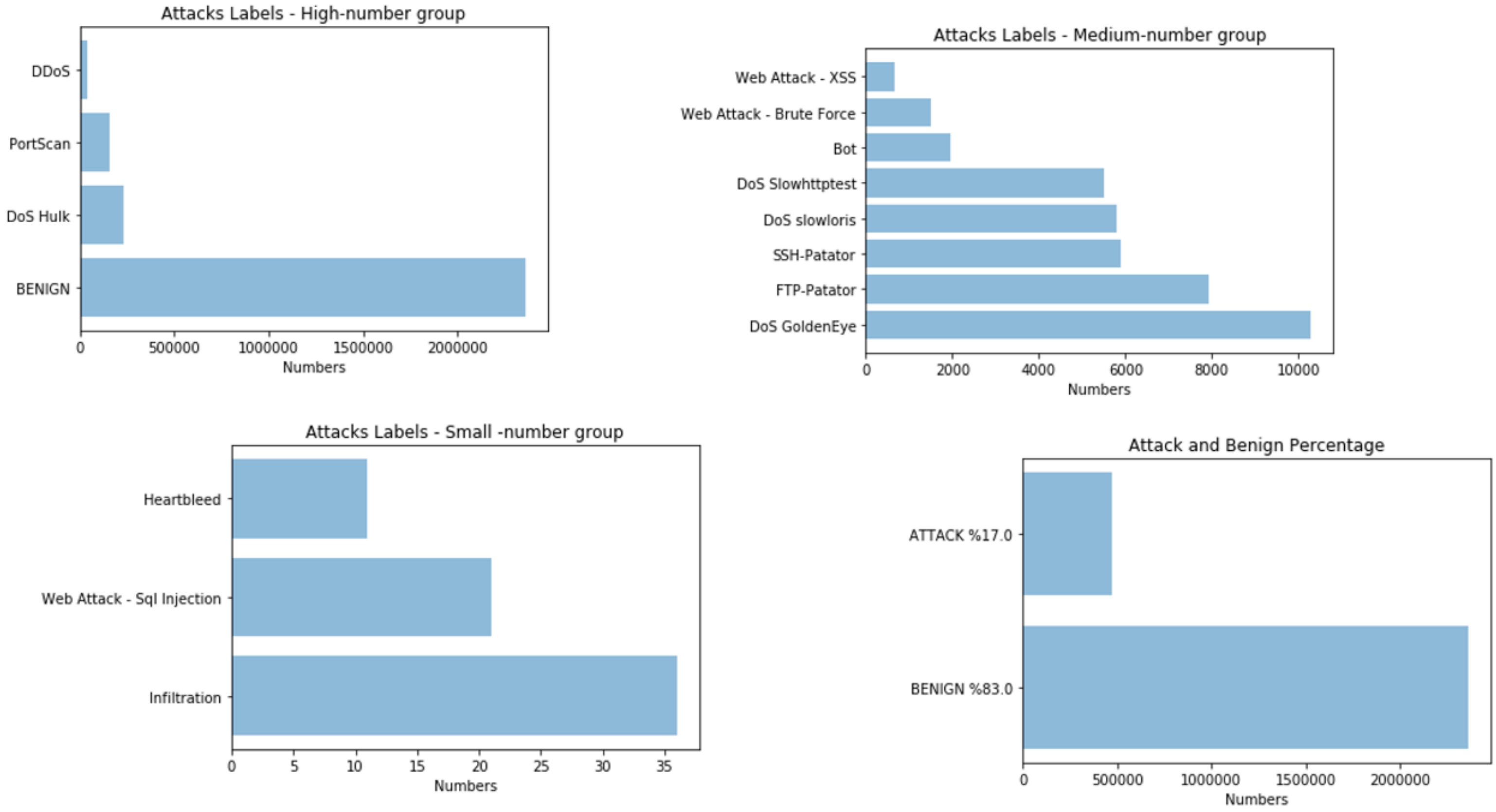


Figure 2

When examining the number of these attacks, it is clear that some are very high. For example, DoS Hulk attacks account for almost half of all attacks, while port scan attacks account for one-third of all attacks. The reason for this unbalanced distribution is the nature of these attacks. Both DoS and PortScan attacks cause excessive data and packet flow during an attack. Therefore, during these attacks, it is natural to see more intensive traffic from normal use and other types of attacks.

The attacks in the dataset can be explained as follows:

1. DoS HULK: one of the DOS attack tool, this tool using the UserAgent forged, to avoid attack detection, can be used to start 500 threads the HTTP GET targets in high frequency FLOOD at the request of the cow force every request is independent and can bypass the server-side cache measures, make processing request HULK is written in Python, change is also very convenient to GET the source code. Because this attack directly affects server load, it can disable the server in a short period of time, such as a minute or so. Hulk has the ability to hide real user agents and use different templates for each attack request. During the attack, it creates a TCP-SYN(Transport Control Protocol-Synchronization) flood and multiple HTTp-GET flood requests. To better understand this attack, look at the TCP-SYN flooding and HTTP-GET flooding methods in [20].

2. SYN Flood (semi-open attack) is a denial-of-service (DDoS) attack whose purpose is to make the server unavailable for legitimate traffic by consuming all available server resources. By repeatedly sending initial connection Request (SYN) packets, an attacker can overwhelm all available ports on the target server machine, causing the target device to fail to respond to legitimate traffic at all. How does SYN Flood attack work? Through the handshake process of TCP connection, SYN Flood attacks the work.Under normal circumstances, a TCP connection displays three different processes to connect.

a. First, the client sends the SYN packet to the server to start the connection.

b. The server responds to the initial packet with the SYN/ACK packet to confirm communication.

c. Finally, the client returns the ACK packet to confirm the packet received from the server. After completing the packet sending and receiving sequence, the TCP connection opens and can send and receive data.

3. HTTP-GET Flood: The principle of HTTP GET Flood attack is very simple. Attackers use attack tools or manipulate zombie hosts to launch a large number of HTTP GET messages to the target server and request URIs involving database operations or other URIs consuming system resources on the server, causing the server resources to be exhausted and unable to respond to normal requests. An attacker can send HTTP GET requests on each TCP connection or use the HTTP 1.1 feature to send multiple HTTP GET requests on each TCP connection.In addition, these requests do not require high network traffic.

4. PortScan: What is port scanning? Port scanning is when a hacker sends a set of port scanning messages in an attempt to break into a user's computer and master the type of computer network service. An attacker can use it to know where to find an attack vulnerability.In essence. The port scan sends messages to each port, one at a time.The type of response received indicates whether the port is being used and vulnerabilities can be found on the computer. "Port" has two meanings: one refers to the physical connection ports, such as hubs, switches, routers and other equipment interface, another is from the logical sense of visual metaphor, generally refers to Port in TCPgP agreement, such as Port 80 refers to the web service is refers to the FTP protocol, 2 l Port transport services, and TCP/IP protocol Port range is very wide, from 0 ~ 65535 belong to the scope of the logical Port.
Principle of port scanning: The principle of port scanning is to send probe packets (mainly for TCP/IP service port) to the target host, and judge the status of the service port through the data returned by probe packets. This information will be recorded and used to judge whether the port is closed or not. For example, the call socket function connect() can be used to connect to the target host and form a complete "triple handshake". If the port is in a listening state, the connect() function will be returned, which means the port is open. The opposite means that the service cannot be provided.Since most of the network access is based on the TCP transport protocol and UDP datagram protocol, it also provides the main scanning object for the attacker. TCPhP protocol, for example, consists of four protocol layers, namely, application layer, transport layer, Internet layer and interface layer. Most network services are also identified through TCP ports, further narrowing the scope of detectability. If an attacker wants to know the service status and content of the target host, he only needs to analyze it from the port number he receives. For example, if the port number 23 of the remote login protocol is detected, the intruder can establish a remote communication connection through the vulnerability by stealing the login account password.
Port scanning method:
a. TCPconnect() scanning mode.This is the most commonly used, most recent type of port scan and allows you to connect to any target computer port without requiring any permissions and saving unit access time by calling the operating system's conect() function.But there's a fatal problem with this approach. It's not so stealthy that it can be easily filtered out.
b. TCPSYN scanning method.This scanning is done via SYN packets and is technically a "semi-open" form. This kind of scanning method overcomes the disadvantage of poor concealment, and not easy to attract the attention of the computer monitoring system, and does not leave a record.But you also need root permissions to do this, which is also not easy to do.
c. IP segment scanning.IP segment scanning is a relatively new technology that can be easily implemented with a few web tools, and it's the random port scanning of everything Hacker likes.Scanning tools only need to determine the IP segment to be scanned, such as 192.168.1.1 ~ 192.168.1.200, you can find out the state of host passing through the router within the IP range.Because this method does not send packets directly through TCP probe, it is very covert.

5. DDoS: Distributed Denial of Service attacks (DDoS) refers to the multiple attackers at different locations to one or more targets at the same time attack, or an attacker controls are located in different position of the multiple machines and use these machines of victims at the same time to carry out attacks by attacking a point is Distributed in different places, this kind of attack is called a Distributed Denial of Service attack, an attacker can have more than one of them.

DDoS is a special form of denial-of-service attack based on DoS, which is a distributed and coordinated large-scale attack mode. Single DoS attack is usually adopts the one-to-one method, which USES the network protocol and some defects of the operating system, adopts the strategy of deception and camouflage to cyber attacks, restoring web server with a large number of requirements of information, network bandwidth or system resources consumption, lead to a network or system to bear that paralyzed and stop provide normal network services. Compared with DoS attacks launched by a single host, distributed denial of service (DDoS) attacks are group actions launched simultaneously with the help of hundreds or even thousands of hosts installed with attack processes after being invaded.

A complete DDoS attack system consists of four parts: attacker, master, agent and target. The master side and the agent side are used to control and actually launch attacks respectively, in which the master side only issues commands but does not participate in the actual attacks, and the agent side issues the actual attack packets of DDoS. The attacker has control or partial control over both the master and the agent computers. It will use various means to hide itself from others during the attack.Once the real attacker transmits the attack command to the master, the attacker can shut down or leave the network. The master publishes the commands to the various proxy hosts. This allows the attacker to avoid being tracked. Each attack proxy host will send a large number of service request packets to the target host, which are disguised and cannot identify its source. Moreover, the service requested by these packets often consumes a large amount of system resources, which makes the target host unable to provide normal services for users. It even causes the system to crash.

6. DoS Goldeneye: It is a Python-based DoS attack. The purpose of this attack is to consume the system resources of the victim, thus preventing legitimate users from receiving the service. Goldeneye is a multithreaded attack that can effectively launch an HTTP flood attack using multithreaded CPU and memory hardware. It does not encrypt packets during an attack, nor does it create a fake source IP (Internet Protocol) address (IP spoofing).It runs on all Linux, Mac, and Windows operating systems.

7. FTP-Patator: The FTP-Patator attack is carried out using Patator[28], a multithreaded tool written in a Python program. FTP(File Transfer Protocol) is a network protocol that provides the transfer of files between clients and servers over a network. To transfer files using FTP, you need to send a valid user name and password over the network.The FTP-Patator attack is a brute force attack to obtain user names and passwords [29].

A brute force attack is an encryption attack designed to obtain a password. In this attack, the attacker attempts to log in as a legitimate user by trying a possible username and password on the system. Software that automatically generates passwords is commonly used for this process. Because users tend to use meaningful and easy-to-remember passwords, passwords are generally weaker than those that are supposed to protect against violent attacks.

A large number of unsuccessful entry attempts in a short period of time is a characteristic of violent attacks. In this context, you can see a dense packet flow during a brute-force attack.In addition, failed entries do not contain large files, so bandwidth consumption and the number of bytes is low [29].

8. SSH-Patator: SSH-Patator attack is made using the Patator[28], a multithreaded tool written in the Python program. SSH (Secure Shell) is an encryption protocol that allows a variety of network services to be operated securely over a network in an insecure environment such as the Internet. The most common use of this protocol is to log in to a remote system [30]. In this type of attack, the attacker's goal is to gain remote access to the system and gain full control over it.

9. DoS Slowloris: Slowloris is an attack method proposed in 2009 by RSnake, a well-known Web security expert, that works by sending HTTP requests to a server at extremely low speed. Since The Web Server has a certain limit on the number of concurrent connections, if these connections are held in bad faith and not released, all the connections of the Web Server will be occupied by the bad faith connections, thus unable to accept new requests, leading to denial of service.

During this attack, a large number of incomplete TCP packets with SYN flags are observed. Package sizes are small, so bandwidth consumption and bytes count are low.

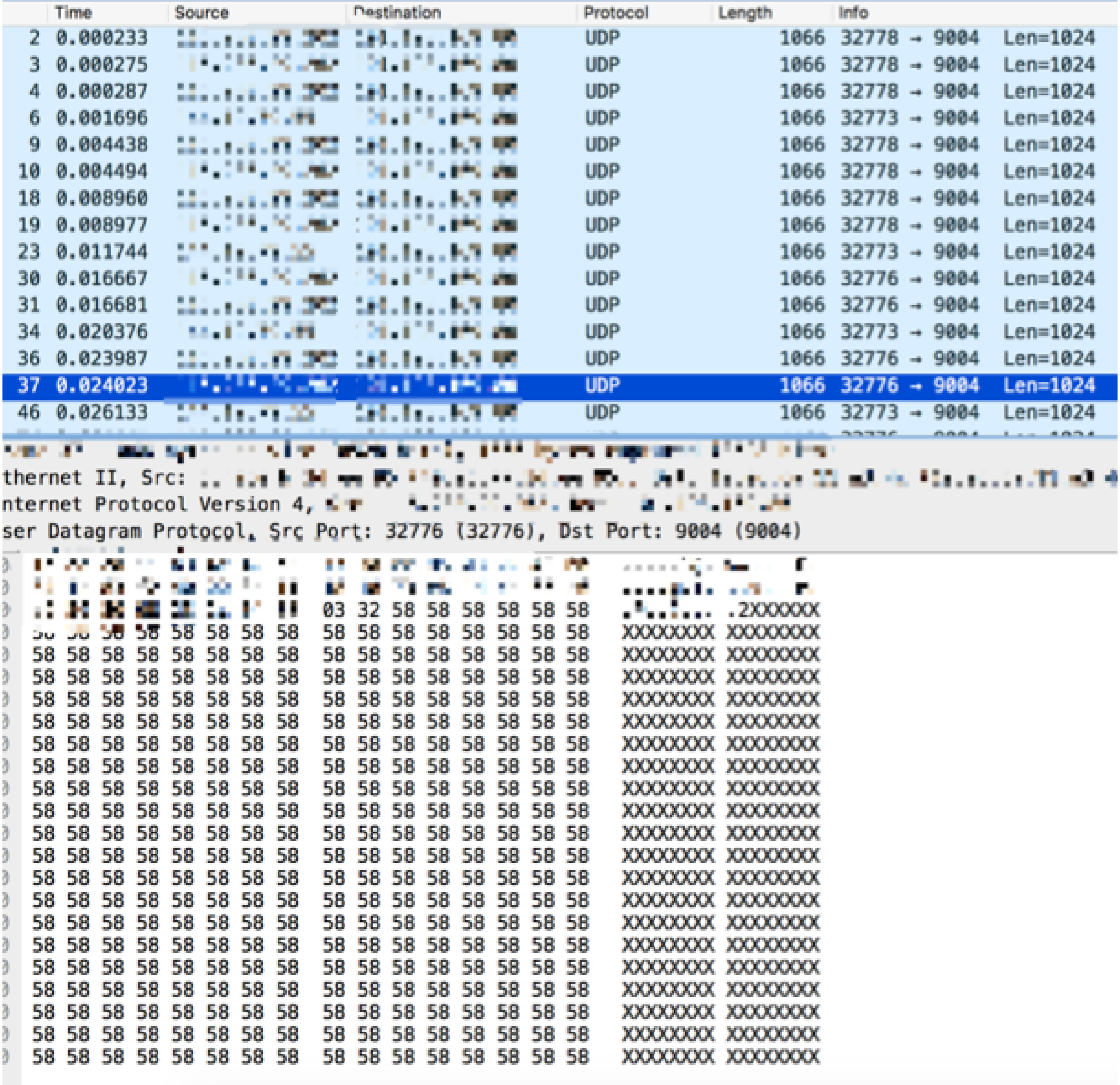
10. DoS SlowHTTPTest: It is a slow attack DoS attack tool that relies on HTTP protocol. The basic principle of the design is that the server will not process the request until it is completely received. If the client sends slowly or incomplete, the server will reserve connection resource pool for it, and a large number of concurrent such requests will lead to DoS.

During a SlowHTTPTest attack, the attacker sends a normal request to the server, but when the response is received from the server, it sets the window size too close to zero, thus slowing down the receiving process as much as possible. In this case, the server will have to allocate resources to process and store a small portion of the file and then most of the rest. In the event of an increase in these requests, the server is unable to meet the demand and ceases service. In addition to the attack, the attacker continues to send SYN and ACK packets to prevent a connection break due to a timeout [34].

This attack is easy to carry out because it does not require high bandwidth or hardware. Also, this attack is difficult to detect in Internet traffic because it looks like an innocent HTTP request that takes a long time [34]. However, if the window size of the connection is much smaller than the normal flow, you will be prompted for this attack.

11. Botnet: It refers to the use of one or more means of transmission, a large number of hosts infected bot program (zombie program), so as to form a one-to-many control network between the controller and the infected host.

Botnet implantation for the server is mostly to take advantage of the computer vulnerability of the other party to invade, or Trojan planting, and then to control the computer of the other party at will, such as deleting files, tampering and destroying, etc. In fact, the main purpose of Botnet implantation is to implant Botnet virus and to assemble Botnet in the form of activation. The botnet can be used as a secondary server to spread virus samples, and can also be used to carry out some DDoS attacks, CC attacks, etc. Some large-scale distributed attacks that require a large number of servers have a great impact. Figure 3 shows the abnormal traffic of the server that was attacked by elKnot DDoS virus.



```
     Time       Source      Destination  Protocol  Length  Info
 2   0.000233   [redacted]  [redacted]   UDP       1066    32778 → 9004  Len=1024
 3   0.000275   [redacted]  [redacted]   UDP       1066    32778 → 9004  Len=1024
 4   0.000287   [redacted]  [redacted]   UDP       1066    32778 → 9004  Len=1024
 6   0.001696   [redacted]  [redacted]   UDP       1066    32773 → 9004  Len=1024
 9   0.004438   [redacted]  [redacted]   UDP       1066    32778 → 9004  Len=1024
10   0.004494   [redacted]  [redacted]   UDP       1066    32778 → 9004  Len=1024
18   0.008960   [redacted]  [redacted]   UDP       1066    32778 → 9004  Len=1024
19   0.008977   [redacted]  [redacted]   UDP       1066    32778 → 9004  Len=1024
23   0.011744   [redacted]  [redacted]   UDP       1066    32773 → 9004  Len=1024
30   0.016667   [redacted]  [redacted]   UDP       1066    32776 → 9004  Len=1024
31   0.016681   [redacted]  [redacted]   UDP       1066    32776 → 9004  Len=1024
34   0.020376   [redacted]  [redacted]   UDP       1066    32773 → 9004  Len=1024
36   0.023987   [redacted]  [redacted]   UDP       1066    32776 → 9004  Len=1024
37   0.024023   [redacted]  [redacted]   UDP       1066    32776 → 9004  Len=1024
46   0.026133   [redacted]  [redacted]   UDP       1066    32773 → 9004  Len=1024

[redacted]
thernet II, Src: [redacted]
nternet Protocol Version 4, [redacted]
ser Datagram Protocol, Src Port: 32776 (32776), Dst Port: 9004 (9004)

[redacted]
[redacted]
[redacted]  03 32 58 58 58 58 58 58  [redacted] .2XXXXXX
58 58 58 58 58 58 58 58  58 58 58 58 58 58 58 58  XXXXXXXX XXXXXXXX
58 58 58 58 58 58 58 58  58 58 58 58 58 58 58 58  XXXXXXXX XXXXXXXX
58 58 58 58 58 58 58 58  58 58 58 58 58 58 58 58  XXXXXXXX XXXXXXXX
58 58 58 58 58 58 58 58  58 58 58 58 58 58 58 58  XXXXXXXX XXXXXXXX
58 58 58 58 58 58 58 58  58 58 58 58 58 58 58 58  XXXXXXXX XXXXXXXX
58 58 58 58 58 58 58 58  58 58 58 58 58 58 58 58  XXXXXXXX XXXXXXXX
58 58 58 58 58 58 58 58  58 58 58 58 58 58 58 58  XXXXXXXX XXXXXXXX
58 58 58 58 58 58 58 58  58 58 58 58 58 58 58 58  XXXXXXXX XXXXXXXX
58 58 58 58 58 58 58 58  58 58 58 58 58 58 58 58  XXXXXXXX XXXXXXXX
58 58 58 58 58 58 58 58  58 58 58 58 58 58 58 58  XXXXXXXX XXXXXXXX
58 58 58 58 58 58 58 58  58 58 58 58 58 58 58 58  XXXXXXXX XXXXXXXX
58 58 58 58 58 58 58 58  58 58 58 58 58 58 58 58  XXXXXXXX XXXXXXXX
58 58 58 58 58 58 58 58  58 58 58 58 58 58 58 58  XXXXXXXX XXXXXXXX
58 58 58 58 58 58 58 58  58 58 58 58 58 58 58 58  XXXXXXXX XXXXXXXX
58 58 58 58 58 58 58 58  58 58 58 58 58 58 58 58  XXXXXXXX XXXXXXXX
58 58 58 58 58 58 58 58  58 58 58 58 58 58 58 58  XXXXXXXX XXXXXXXX
58 58 58 58 58 58 58 58  58 58 58 58 58 58 58 58  XXXXXXXX XXXXXXXX
58 58 58 58 58 58 58 58  58 58 58 58 58 58 58 58  XXXXXXXX XXXXXXXX
58 58 58 58 58 58 58 58  58 58 58 58 58 58 58 58  XXXXXXXX XXXXXXXX
58 58 58 58 58 58 58 58  58 58 58 58 58 58 58 58  XXXXXXXX XXXXXXXX
```

Figure 3

12. Web Attack: Based on the data set, in this set of Web attacks, three different Web-based attacks were launched against the vulnerable PHP(hypertext preprocessor -- a scripting language)/ MySQL(an open source database management framework) Web application that was identified as the victim. These attacks are: Brute Force, XSS (Cross-Site Scripting) and SQL (Structured Query Language – a database management system) Injection attack.

13. Infiltration: The penetration concept used in this dataset is not a general attack, but a specific attack concept for a specific scenario. In this case, the network is subject to an information-gathering attack as virus files enter the system.
The attack scenario looks like this: the virus enters the system through a file the victim downloaded from Dropbox using Windows and a file the victim copied from a USB flash drive using Macintosh. In the next stage of the attack, the attacker exploits the vulnerability created by the virus to perform port scanning attacks on the network.

14. Heartbleed: Heartbleed is a security vulnerability that appeared in OpenSSL, an encryption library that is widely used to implement the Internet's Transport Layer (TLS) protocol. It was introduced into the software in 2012 and first revealed to the public in April 2014. As long as you are using a flawed OpenSSL instance, either the server or the client can be vulnerable. This problem is due to the lack of proper validation of inputs (lack of boundary checks) when implementing the HEARTBEAT protocol for TLS, and more data being read than should be allowed. Such bugs no longer exist in mainstream OpenSSL, and the current experiment needs to be done on the older Ubuntu12 system.
During the attack, a specially created heartbeat request is sent to the server. This requirement is virtually empty, but points out that it carries a lot of data. At this stage, due to a vulnerability in OpenSSL, the server will respond to this message by sending a block of memory of a specified length. These sections contain various confidential information, such as personal information, user names and passwords, and should not be sent. These packages can be up to 16 KILobytes in size, and the attacker can repeat this operation indefinitely.
During this attack, some anomalies in the network flow can be observed. The length of the incoming message is less than 20 bytes from the minimum message size of the heartbeat. The length of the outgoing message is kilobytes, which is too large for a heartbeat response. The size difference between outgoing and incoming messages is another way to understand an attack. In a normal heartbeat message, the outgoing and incoming messages have the same length. In the case of an attack, the length of the input message is too small and the length of the output message is too large.

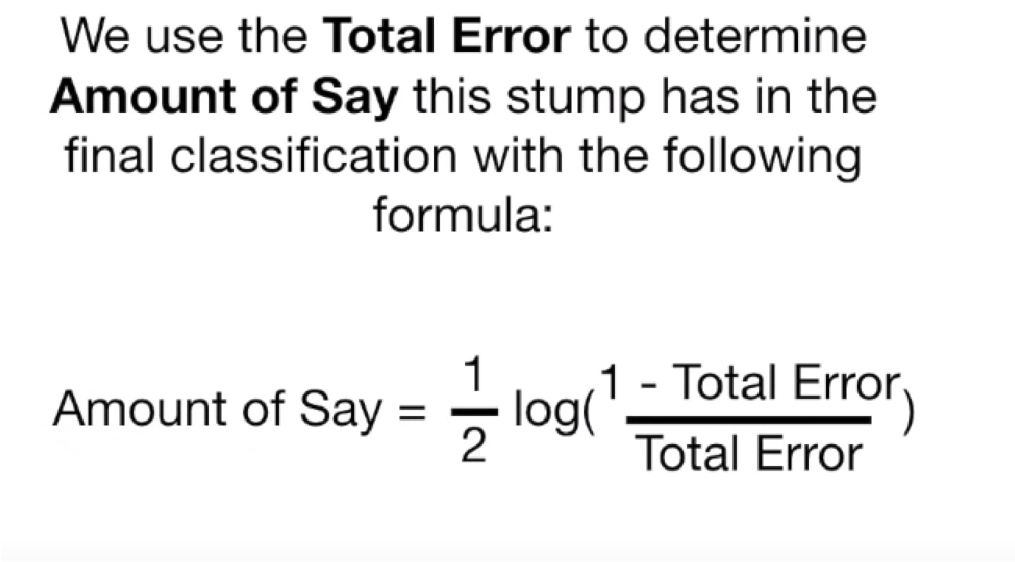
Ⅵ.Machine Learning Introduction

Machine learning (ML) technology is a very popular way to identify and classify patterns in different domains. Its main goal is to give computers the ability to learn automatically, that is, to extract knowledge from a process under certain conditions. Milliliters attempt to extract knowledge from a set of features or attributes that represent measurable attributes of a process or observed phenomenon. In this way, different models are trained to complete the learning process, namely, classification model, prediction model or clustering model. Their use depends on the nature of the problem. In this project, the following 9 supervised learning algorithms are selected for 14 data sets of tag-based network attacks:

1. Decision Tree: Decision tree is a data structure that can be used for classification and regression. The core components of a decision tree are root, internal, and leaf nodes. They used the Gini coefficient as the basis for the division. Here's how the decision tree works:
First, determine the root node of the decision tree, since it is the starting point of the decision tree, from which all internal nodes begin and extend. There are four steps to determine the root node. The first step is to calculate the characteristics of the training set one by one. Here, for an example of the list extracted from the left, we
need to rank the Feature1 values from smaller to larger. In the second step, we need to calculate the average of every two adjacent features. The third step is to calculate the Gini impurity score of each average value. The Gini coefficient reflects the probability of randomly picking two samples from a different sample set. The greater the Gini coefficient is, the greater the probability of randomly selecting two samples from different sample sets, that is, the smaller the sample purity is. Finally, each path starting from the root node, after two node screening, will obtain the highest purity classification, this node is the leaf node. Therefore, the process of spreading the branches and leaves of the decision tree is the process of gradually improving the classification purity.

2. Random Forest: After introducing the decision tree, I will introduce the random forest, because it is based on the decision tree, constructing multiple decision trees to generate a classification algorithm. First of all, there are two random sampling processes. Random forests require sampling of input data and columns. Row sampling adopts the method of putting back, that is, there may be duplicate samples in the sample set obtained by sampling. The new example must ensure that the number of rows and columns and the table header are the same as the original table. A newly created random list, we need to use it to generate these lists, for each function, the use data set to create the decision tree is made by random list before, so each use feature list is different.
The decision tree for each feature is trained from the root node. If the termination condition is reached on the current node, the current node is set to a leaf. If it is a classification problem, predict the type of output leaf node with the maximum number of samples of the current node. If it is a regression problem, the forecast outputs the average value of each sample value in the current node example. If the current node does not meet the termination condition, the ability to select the random feature is not put back, and the training continues. The termination conditions mentioned here are the number of N_estimators for the decision tree, the depth of the decision tree, max_depth, and so on, which can be set during modeling. In the process of fitting the random forest model, 1/3 of the original training data is not used because the training list is repeatable. This type of data is called an out-of-package data set and can be used as a test set.

3. AdaBoost: the full name is the adaptive boost, which means it is a kind of can adapt and improve the effect of classification algorithm, its adaptability is a basic classifier before the misclassification of sample weight will increase, and the correct classification of the sample weight will be reduced, and then use it to train the next basic classifier at the same time, in each iteration to join a new weak classifier, until you reach a predetermined small enough error rate or predetermined maximum number of iterations, to determine the final strong classifier error rate and the maximum number of iterations can be set in the modeling.
First, we need to initialize the weight distribution weights of the training data. If there are N training sample data, each training sample is initially assigned the same weight: weight =1/N. Through the calculation and selection method of ginI impurity in the decision tree, the weak classifier with the lowest error rate was selected as the first pile, in which ginI impurity was the error rate. Then we need to train the weak classifier. The specific training process is as follows: if a training sample point is accurately classified by a weak classifier, then the weight corresponding to the next training needs to be reduced. Conversely, if a training sample point is misclassified, its weight should be increased. The weighted updated sample set is used to train the next classifier, and the whole training process is iterative. The parameter that determines whether the weight value decreases or increases in the next iteration is Amount of Say. Figure 4 shows its calculation formula:

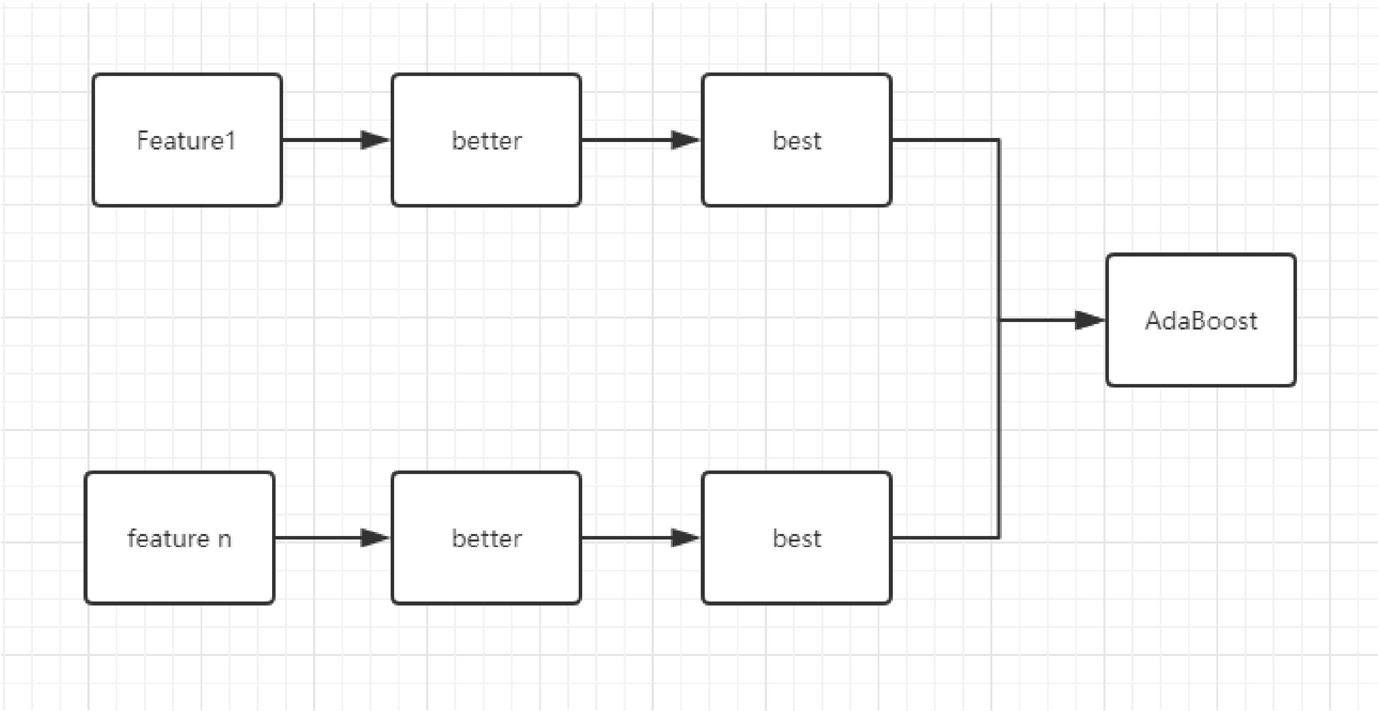


We use the **Total Error** to determine **Amount of Say** this stump has in the final classification with the following formula:

$$\text{Amount of Say} = \frac{1}{2}\log\left(\frac{1 - \text{Total Error}}{\text{Total Error}}\right)$$

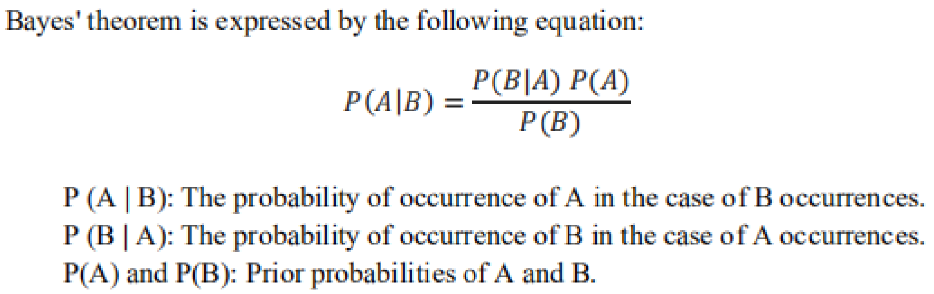
Fegure 4

Here, I use only the two features shown in fegure 5 to demonstrate the iterative evolution of each stump. Evolution completes each feature of the weak classifier will eventually form a strong classifier. The value of say of each weak classifier determines its weight in the final classifier.



Fegure 5

4. Gaussian Naive Bayes: The algorithm is used to adjust the value of feature, and its numerical distribution is fitted to the normal distribution.(The normal distribution curve is determined by the mean value and standard deviation of feature); In the test set, reference tag values make independent assumptions. The probability of independent events is calculated according to the following fegure 6. The large probability value is the output.



Bayes' theorem is expressed by the following equation:

$$P(A|B) = \frac{P(B|A)\ P(A)}{P(B)}$$

P (A | B): The probability of occurrence of A in the case of B occurrences.
P (B | A): The probability of occurrence of B in the case of A occurrences.
P(A) and P(B): Prior probabilities of A and B.

Fegure 6

5. K Nearest Neighbour: KNN is arguably one of the simplest classification algorithm, assuming that there are three characteristics in fitting, numerical characteristic of the three will be in 2 d figure shown in figure 7 of this kind of situation, the green dot in the graph represent the unknown of the test set point, a value of 3 K or choose green near the point of the recent three points, there are two blue dot and a red dot, the attribute of green dot as blue.

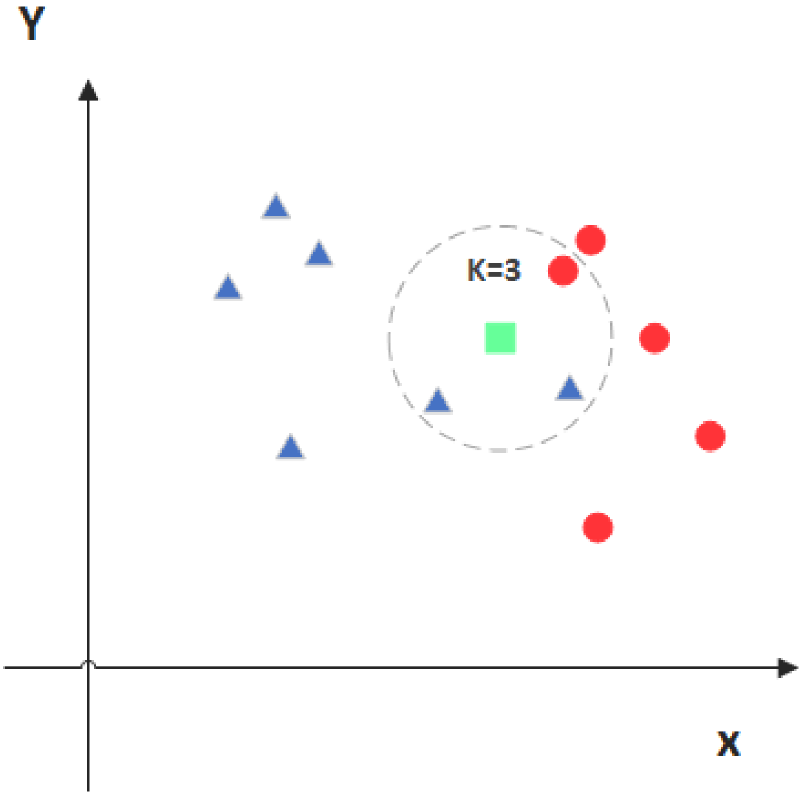


Figure 7

We know that K is important, so how do we know what K is? The answer is through cross-validation (the sample data will be divided into training data and the use of validation data in a certain proportion, such as 6:4 to split a part of training data and test data), from the selection of a smaller value K. And then calculate the variance of validation set, and finally found a more suitable K value calculation of variance of cross validation after you get below this figure would be easy to understand, as you increase K, general error rate will be reduced, because there are more samples can be for reference will be better classification effect, but note that when the K value is larger, the error rate is higher, it is easy to understand, for example, Look at the fegure 8, there are a total of 35 samples when you increased to 30 K, but meaningless basically choose K point can choose a larger point K, when it continues to increase/decrease error rate will rise, as shown in figure of K = 10.

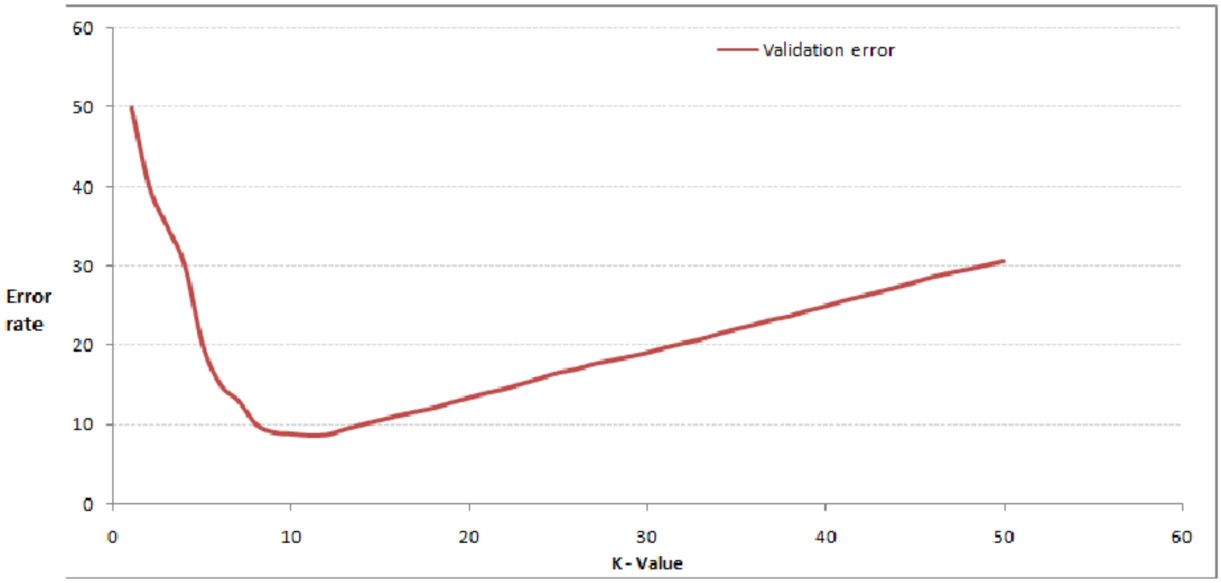


Figure 8

6. MLP: Multi-Layer Perceptron is a type of artificial neural network. Artificial neural networks (ANN) are a method of machine learning inspired by the way the human brain works. The purpose of this method is to mimic the characteristics of the human brain, such as learning, decision-making and acquiring new information.

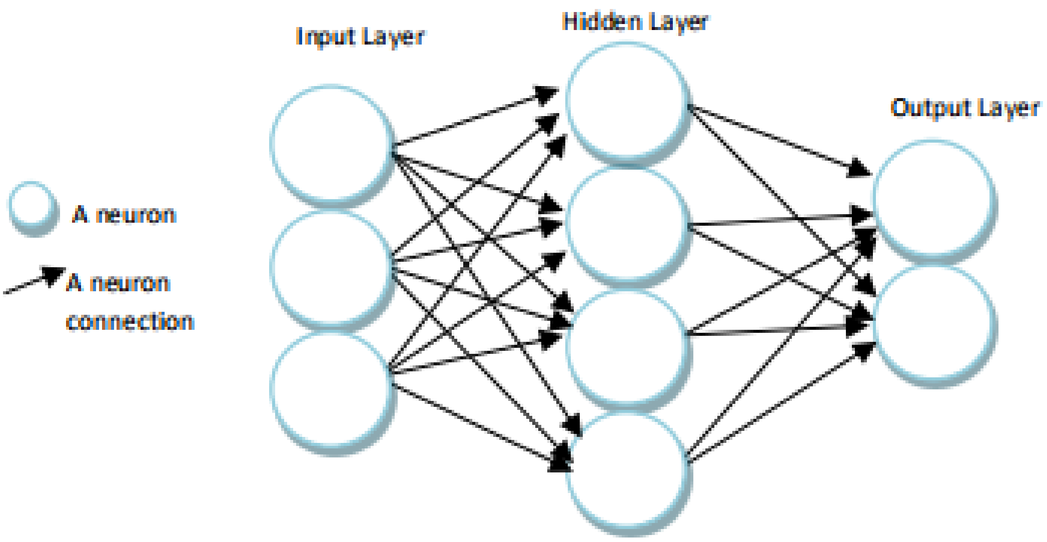


Figure 9

From the figure 9, The input layer is the stage where the MLP is responsible for receiving data.No information processing is performed at this level.Only the information received is transmitted to the next layer, the hidden layer.

In the hidden layer, the data sent from the input layer is processed and transmitted to the next layer, the output layer.

In the output layer, which is the last layer, each cell is tied to all the cells in the hidden layer, and the results of the processed data in the hidden layer are served at this stage.

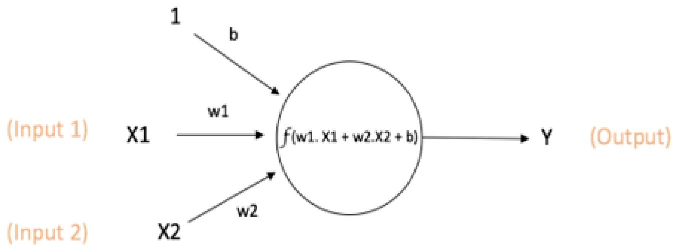
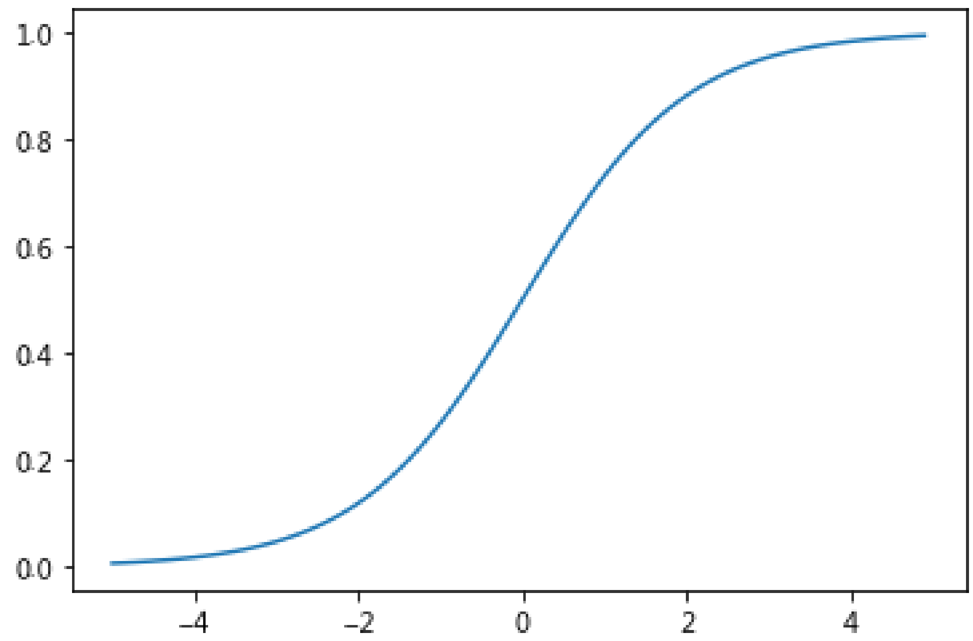


Figure 10

A neuron is the basic unit of an artificial neural network, generally called a node or unit. A single neuron receives input from other neurons or external sources, and then calculates the output. Each input has a weight, namely W, which depends on the relative importance of other inputs. Figure 10 is simple, a single neuron network to accept a weight of W1 and W2 input, enter 1 weights are b (called deviation) is the main purpose of bias for each neuron to provide a constant training, mainly through the activation function in the quadrant to move to the left or right, the mathematical expression of the output neurons function here: Y = f(w1*x1+w2*x2+b), f (x) is an activation function here is important to note that the activation function is not going to anything by name to activate.In the aforementioned neurons in the output, the activation function f (x) is a nonlinear function, because most of the earth's data is not linear, math, when activation function is linear, a double neural network approximation almost all of the functions, but if the activation function for the same activation function, f (x) = x, does not meet the nature, the entire network is essentially a single neural network in this case, I'll introduce you to the most commonly used activation function: Sigmoid.

The function of Sigmoid can transform the output between 0 and 1, which means that if it's a very large negative number it will output 0, and if it's a very large positive number it will output 1. The graphical representation of Sigmoid is shown here. Here is the formulas and image of Sigmoid:

$$f(x) = 1/(1+e)^{(-x)}$$



MLP structure of the main structure of the forward, also known as the feedforward structure, is the earliest inventions of the most simple neural network as shown in figure 11, it is composed of multilayer arrangement of multiple nodes (neurons) in the adjacent layer node has a connection or edge all connections have weight, we can see, the network is divided into three regions: the most on the left side of the input layer, hidden layer in the middle of the, and most the right side of the output layer, in short, the input layer is responsible for the external information is important to note that there is no calculation, only the information is passed to the hidden layerThe calculation of hidden layer is responsible for most, but in fact, there can be multiple hidden layer in the feedforward network, also can not output layer is responsible for the part of the calculation and the transfer of information to the outside world we say feedforward neural network is one of the most simple network, because: information just one-way movement, from the input layer to move forward, and then through the hidden layer to output layer, the network without loop or loop.

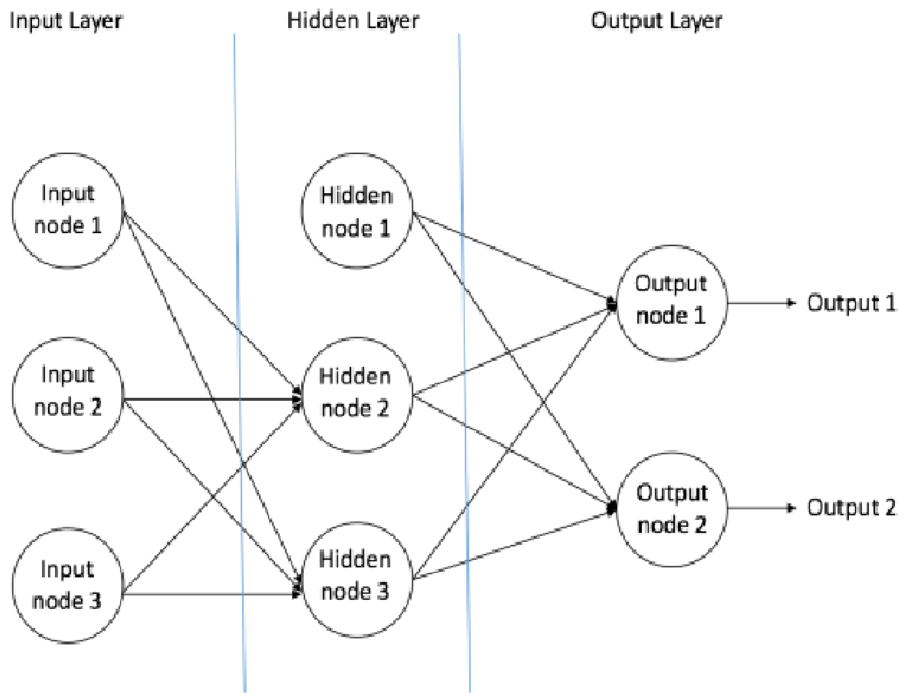


Figure 11

7. Quadratic Discriminant Analysis: The core method of QDA is linear classification using covariance. Covariance is defined as the covariance of any two random variables X and Y, denoted as Cov(X,Y), and mathematically defined as: Cov(X,Y)=E{[X– E(X)][Y-E(Y)]}. As X gets larger and larger,Y gets larger and larger, and then Cov(X,Y) > 0; As X goes down,Y goes down, so does Cov(X,Y) > 0; As the value of X increases, the value of Y decreases, and then Cov(X,Y) < 0;As X goes down,Y goes up, and Cov(X,Y) < 0.

8. Perceptron：The idea of perceptrons is simple, like we have a lot of boys and girls on a platform, and the model of perceptrons is to try to find a straight line that separates all the boys from all the girls. In three dimensions or higher dimensions, the perceptron model is trying to find a hyperplane that separates all the binary categories.If we can't find such a line. It means that the categories are linearly indivisible, and that means that the perceptron model doesn't fit the classification of your data. One of the biggest prerequisites for using perceptrons is that the data is linearly separable. This severely limits the use of perceptrons. But it's important because perceptrons are the basis for many other machine learning algorithms, like neural networks and support vector machines.

F (X) = Sign (W *X + B) is the model function of perceptron, X is the feature of training data set, F (X) is the label, sign is the symbol function, return value, if X is greater than 0, sign returns 1; Equals 0, returns 0; Less than zero, then return 1 x symbol determines the sign function return value assuming samples linearly separable, perception machine learning goal is for positive and negative samples can be completely separated from the separating hyperplane, namely to find w, b (for wx + b = 0 determines the separating hyperplane) so we need to define a learning strategy, namely define the loss function, and minimize the w by the training sample is the normal vector of the expression of vector (perpendicular to the surface normal vector), b is the hyperplane intercept the hyperplane will sample points divided into two categories, the positive and negative when (w) * x + b > 0, y = 1, when w*X + b < 0, y = -1.

9. Linear SVM algorithm: Support Vector Machines (SVM) is a dichotomous model. Its basic model is the linear classifier with the largest interval defined in the feature space, which is different from perceptron.SVM also includes kernel techniques, which make it a essentially nonlinear classifier. The learning strategy of SVM is interval maximization, which can be formalized as a problem for solving convex quadratic programming, which is also equivalent to the regularized hinge loss function minimization problem. The learning algorithm of SVM is the optimization algorithm for solving convex quadratic programming.

Learning is the basic idea of solving the VM correctly divided into training data set and geometric interval maximum separation hyperplane are shown in figure 12 below, [w * x + b = 0] for separating hyperplane, namely for linear separable datasets, such hyperplane has an infinite number (namely perception machine), but the geometric interval maximum separation hyperplane is the only one.

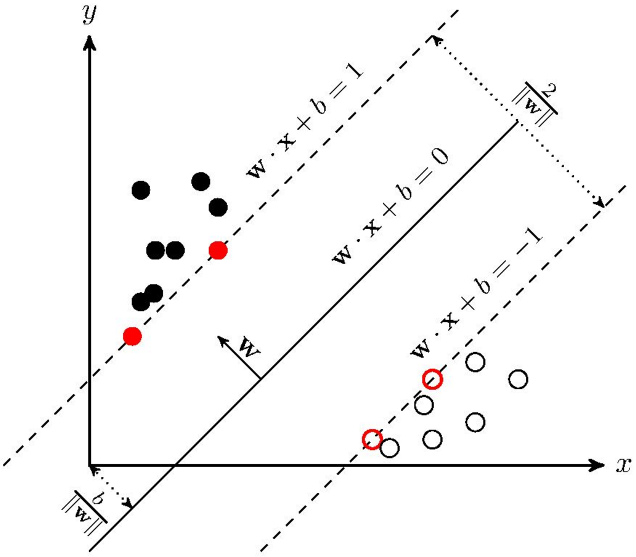

y
$\mathbf{w}\cdot\mathbf{x}+b=1$
$\frac{2}{\|\mathbf{w}\|}$
$\mathbf{w}\cdot\mathbf{x}+b=0$
$\mathbf{w}\cdot\mathbf{x}+b=-1$
$\mathbf{w}$
x
$\frac{b}{\|\mathbf{w}\|}$

Figure 11

VII. Data Preprocessing

Before you actually use the dataset, you may need to make some changes to it to make it more efficient. To this end, this section corrects some of the flaws in the data set used and edits some of the data. The dataset file contains 3119345 stream records. The distribution of these river records is shown in Table 1. When you examine these records, you can see that the 288602 records are incorrect/incomplete. The first step in the preprocessing process is to remove these unnecessary records.

Another error about the data set is in the columns that make up the characteristics. The dataset file consists of 86 columns that define stream properties such as stream ID, source IP, source port, and so on. However, the Fwd header length feature, which defines a forward data stream of the total bytes used, is written twice (columns 41 and 62). Removing the duplicate column (Column 62) corrects this error.
Another change that needs to be made to the dataset is to convert the attributes, including the class value and string value (stream ID, source IP, target IP, timestamp, external IP) into digital data for the machine learning algorithm. This can be done using LabelEncoder() in the Sklearn class. In this way, various string values that cannot be used in machine learning operations will result in integer values between 0 and N-1 that are more suitable for processing.

However, although the " Label" tag is a categorical feature, no changes have been made on it. The reason is that during the processing, the original categories are needed in order to classify the attack types in different forms and to try different approaches. Finally, some minor structural changes should be made to the dataset, including:

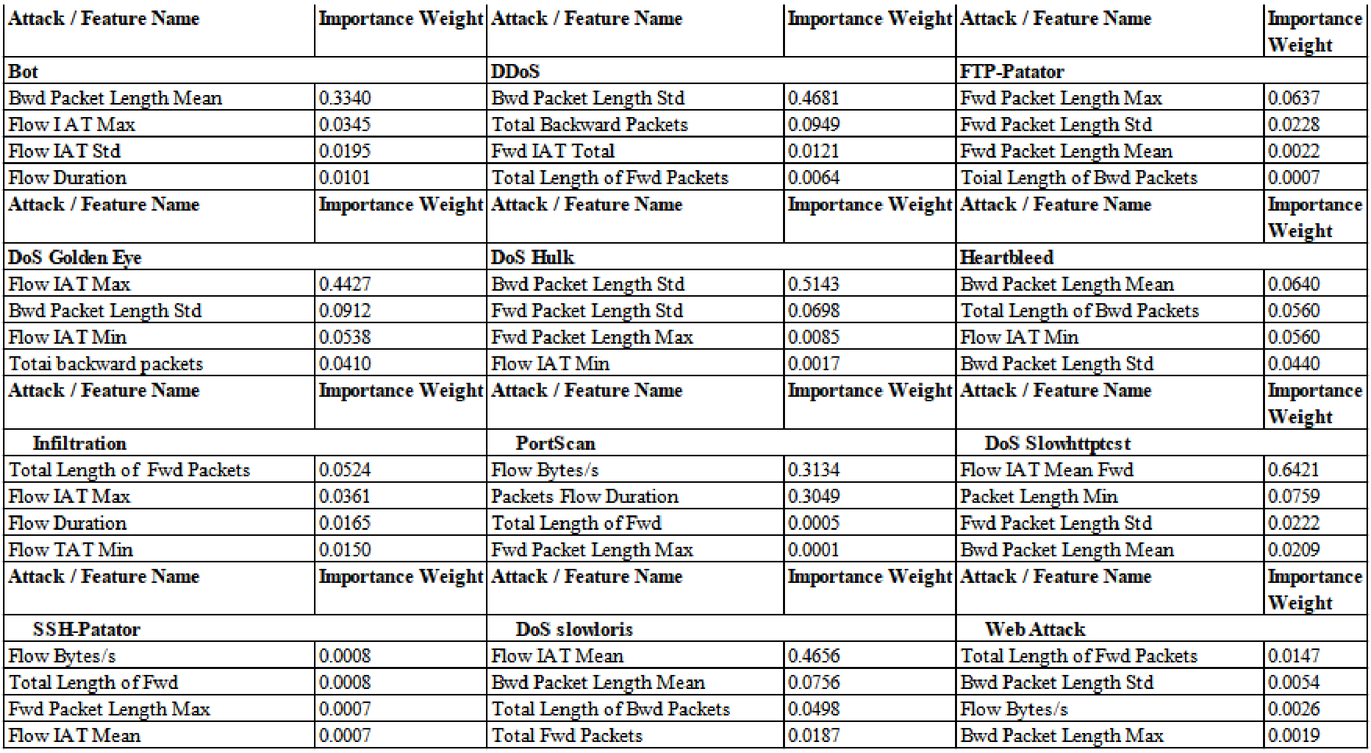
a. In the Label feature, the character "– " (Unicode Decimal Code &#8211) used to identify the web attack subtypes (Web Attack - Brute Force, Web Attack - XSS, Web Attack -SQL Injection) must be replaced with the character "-" (Unicode Decimal Code &#45), since utf-8, the default codec of Pandas library, does not recognize it. Otherwise, the Pandas library that will not recognize this character and it will fail.
b. "Flow Bytes/s", "Flow Packets/s" features include the values "Infinity" and "NaN" in addition to the numerical values, which can be modified to -1 and 0 respectively to make them suitable for machine learning algorithms.

Ⅷ. Feature Selection:

In this section, the characteristics of the dataset are evaluated to determine which characteristics are important for defining what kind of attack.
The feature weight of this project is extracted by random forest classifier.Feature screening of the two data sets will be performed as follows:

1. Feature Selection According to Attack Types: To do this calculation, a special file is created for every kind of attack by isolating the attack from other attacks. This file contains the entire stream identified as the attack and the data stream identified as randomly selected "Benign" (30% Attack, 70% Benign). The Random Forest Regressor[58] class of Sklearn is used when importance weights of features are calculated. This algorithm creates a decision-forest. In this decision forest, each feature is given a weight of importance as to how useful they are in the construction of the decision-tree. When the process is finished, these importance weights of features are compared and sorted [60]. The sum of the importance weights of all the properties gives the total importance weight of the decision tree. The comparison of the score of any feature to the score of the whole tree gives information about the importance of that feature in the decision tree. However, 8 features (Flow ID, Source IP, Source Port, Destination IP, Destination Port, Protocol, Timestamp, External IP) between 85 properties must not be included in the calculation, when the weight of importance is calculated. Although these features are used in classical approaches, it is possible that an attacker would prefer not to use well-known ports to escape control or to circumvent operating system constraints or he can use generated / fake IP addresses. Also, many ports are used dynamically, and many applications are transmitted over the same port. So, it can be misleading to use the port number [61]. In this context, while choosing the attribute importance of the attack, it will be much more effective to eliminate the misleading features such as IP address, Port number, Timestamp, use more generic and invariant attributes to define the attack. Because the shape of the data will give much more information aboutwhether or not it is an attack.

The distribution of features and four attributes with the most significance value for each attack can be seen from Table 1.



Attack / Feature Name	Importance Weight	Attack / Feature Name	Importance Weight	Attack / Feature Name	Importance Weight
Bot		**DDoS**		**FTP-Patator**	
Bwd Packet Length Mean	0.3340	Bwd Packet Length Std	0.4681	Fwd Packet Length Max	0.0637
Flow I AT Max	0.0345	Total Backward Packets	0.0949	Fwd Packet Length Std	0.0228
Flow IAT Std	0.0195	Fwd IAT Total	0.0121	Fwd Packet Length Mean	0.0022
Flow Duration	0.0101	Total Length of Fwd Packets	0.0064	Toial Length of Bwd Packets	0.0007
Attack / Feature Name	**Importance Weight**	**Attack / Feature Name**	**Importance Weight**	**Attack / Feature Name**	**Importance Weight**
DoS Golden Eye		**DoS Hulk**		**Heartbleed**	
Flow IAT Max	0.4427	Bwd Packet Length Std	0.5143	Bwd Packet Length Mean	0.0640
Bwd Packet Length Std	0.0912	Fwd Packet Length Std	0.0698	Total Length of Bwd Packets	0.0560
Flow IAT Min	0.0538	Fwd Packet Length Max	0.0085	Flow IAT Min	0.0560
Totai backward packets	0.0410	Flow IAT Min	0.0017	Bwd Packet Length Std	0.0440
Attack / Feature Name	**Importance Weight**	**Attack / Feature Name**	**Importance Weight**	**Attack / Feature Name**	**Importance Weight**
Infiltration		**PortScan**		**DoS Slowhttptest**	
Total Length of Fwd Packets	0.0524	Flow Bytes/s	0.3134	Flow IAT Mean Fwd	0.6421
Flow IAT Max	0.0361	Packets Flow Duration	0.3049	Packet Length Min	0.0759
Flow Duration	0.0165	Total Length of Fwd	0.0005	Fwd Packet Length Std	0.0222
Flow TAT Min	0.0150	Fwd Packet Length Max	0.0001	Bwd Packet Length Mean	0.0209
Attack / Feature Name	**Importance Weight**	**Attack / Feature Name**	**Importance Weight**	**Attack / Feature Name**	**Importance Weight**
SSH-Patator		**DoS slowloris**		**Web Attack**	
Flow Bytes/s	0.0008	Flow IAT Mean	0.4656	Total Length of Fwd Packets	0.0147
Total Length of Fwd	0.0008	Bwd Packet Length Mean	0.0756	Bwd Packet Length Std	0.0054
Fwd Packet Length Max	0.0007	Total Length of Bwd Packets	0.0498	Flow Bytes/s	0.0026
Flow IAT Mean	0.0007	Total Fwd Packets	0.0187	Bwd Packet Length Max	0.0019

Table 1

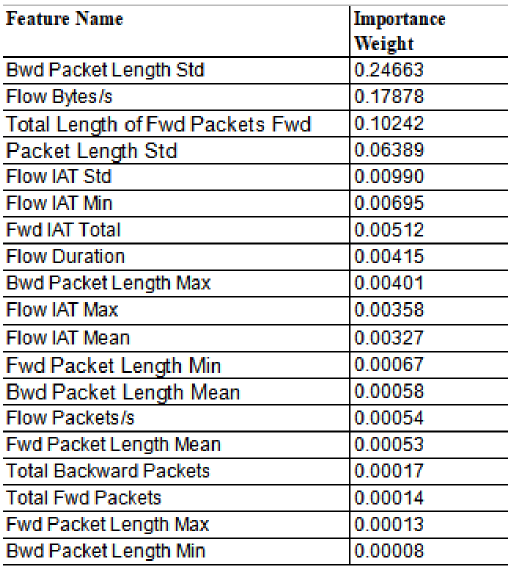
From the above table, we can see the four most important characteristics corresponding to each network attack. According to the previous description of the network attack, we can find some necessary links between the characteristics and the network attack:

On the one hand, for these twelve kinds of network attacks, the length of packet, the maximum traffic and the duration of traffic are significantly higher than those of normal traffic.

On the other hand, for the three attacks of SSH-Patator, Web Attack and PortScan, it is not difficult to find that their Flow Bytes/s are significantly higher than those of normal data Flow.

In conclusion, for conventional network attacks, data traffic and packet length and size are often the most important indicators for detecting anomalies.

2. Feature Selection According to Attack or Benign: Another approach in feature selection is to apply the Random Forest Regressor operation to the whole dataset by collecting all attack types under a single label: " attack" . So, the data in this file contains only the attack and benign tags. As a result of this operation, the feature list obtained is shown in Table 4 and graphics of feature in Figure 2.



Feature Name	Importance Weight
Bwd Packet Length Std	0.24663
Flow Bytes/s	0.17878
Total Length of Fwd Packets Fwd	0.10242
Packet Length Std	0.06389
Flow IAT Std	0.00990
Flow IAT Min	0.00695
Fwd IAT Total	0.00512
Flow Duration	0.00415
Bwd Packet Length Max	0.00401
Flow IAT Max	0.00358
Flow IAT Mean	0.00327
Fwd Packet Length Min	0.00067
Bwd Packet Length Mean	0.00058
Flow Packets/s	0.00054
Fwd Packet Length Mean	0.00053
Total Backward Packets	0.00017
Total Fwd Packets	0.00014
Fwd Packet Length Max	0.00013
Bwd Packet Length Min	0.00008

Table 2

When the data corresponding to all 14 types of network attacks are summarized in a data set, we can get the above 19 valid features (the remaining feature weights are too small to be considered). It is not difficult to find that 12 of the data packet values are displayed as exceptions, and 7 data traffic indicators are also displayed as exceptions.Therefore, it is not hard to say that the exception of network data is in the form of the exception of packets and traffic.

Ⅸ. Metrics for ML Evalution:

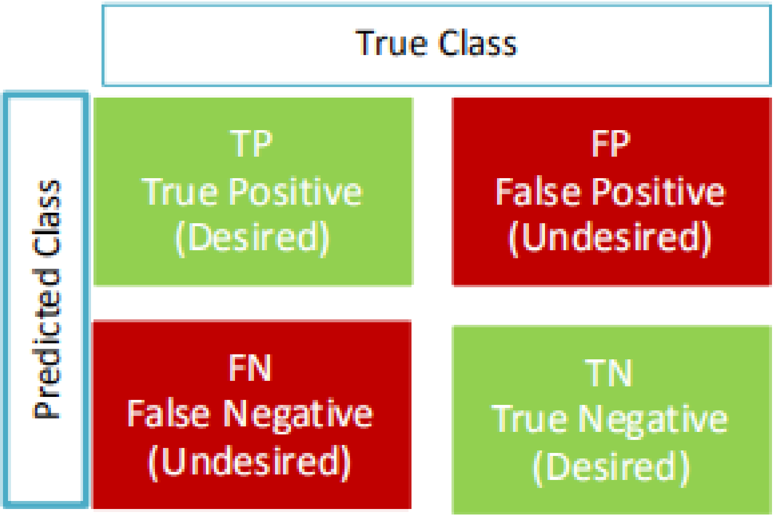
In order to evaluate network data effectively, high detection rate and low false positive rate are the key factors to be considered. Multiple metrics can be used for evaluation. It should be pointed out that the detection rate as the only evaluation index cannot reflect the real performance of the algorithm. Therefore, the results of this study were evaluated according to the four standards of accuracy, precision, F-measure and recall. All of these conditions are between 0 and 1. As it approaches 1, performance increases, and as it approaches 0, performance degrades.

1. Accuracy: The ratio of successfully categorized data to total data;
2. Recall (Sensitivity): The ratio of data classified as an attack to all attack data;
3.Precision: The ratio of successful classified data as the attack to all data classified as the attack;
4. F-measure (F-score/F1-score): The harmonic-mean of sensitivity and precision. This concept is used to express the overall success. So, in this study, when analysing the results, it will be focused, especially on the F1 Score.

In calculating these four items, the four values summarized below are used:

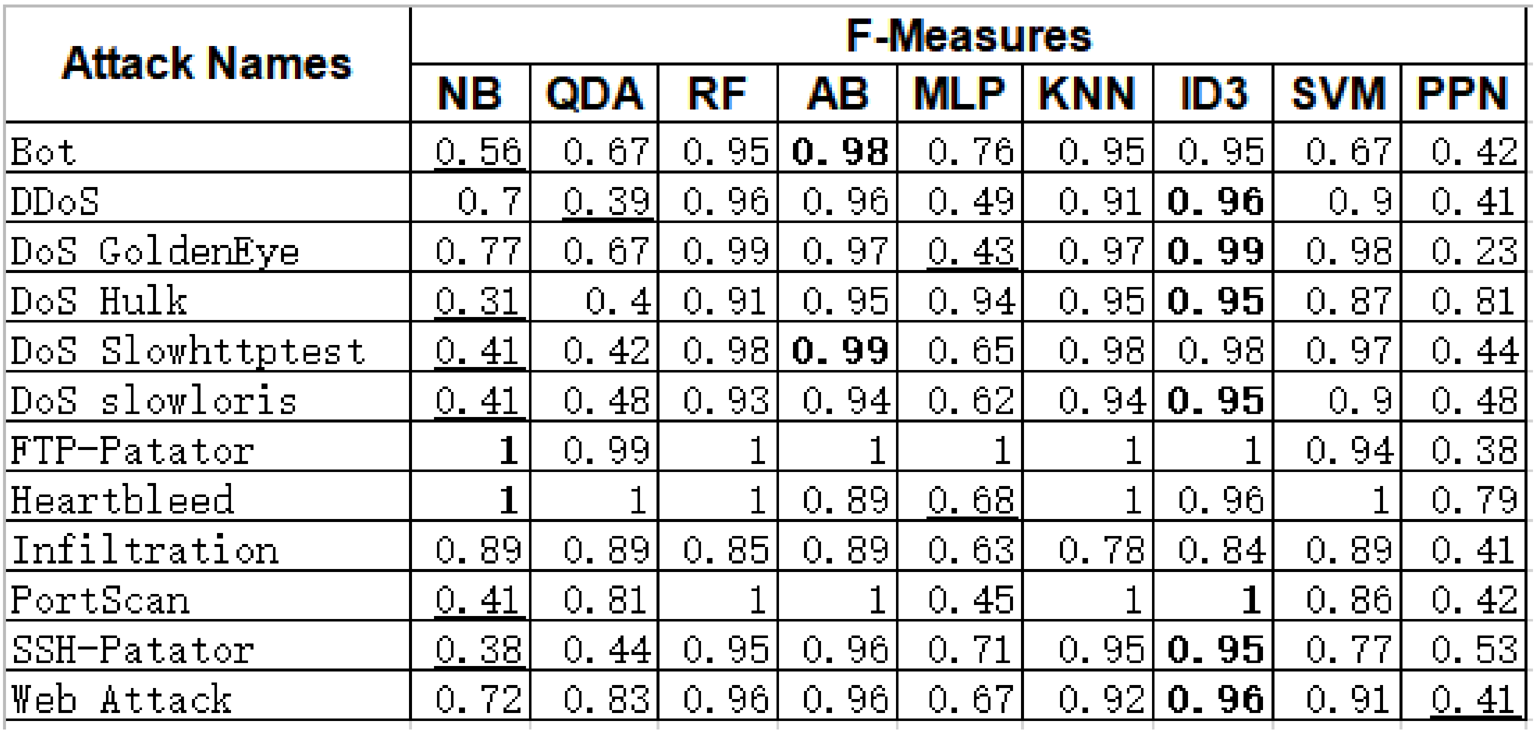
- True Positive (TP): Number of intrusions correctly detected, which is True Positive (Correct Detection). The attack data classified as attack；
- True Negative (TN): Number of non-intrusions correctlydetected,which is False Positive (Type-1 Error). The benign data classified as attack. ；
- False Positive (FP): Number of non-intrusions incor rectly detected, which is False Negative (Type-2 Error). The attack data classified as benign. ；
- False Negative (FN): Number of intrusions incorrectly detected, which is True Negative (Correct Rejection). The benign data classified as benign.

This distribution is presented by visualizing Confusion matrix in Figure 12.



Ⅹ.Evaluation of the ML Algorithms:

1. Using 12 Attack Types: Nine different machine learning methods were applied to 12 different types of attacks, and the results were
shown in Table 3.In the results shown in Table 3, the maximum and minimum grades are highlighted as follows: the maximum score is shown in bold and the minimum score is underlined.In the algorithm results, if the F-measure values are equal, in order to eliminate the equality, the following values are checked : accuracy, precision, recall and time respectively.



Attack Names	F-Measures								
	NB	QDA	RF	AB	MLP	KNN	ID3	SVM	PPN
Bot	0.56	0.67	0.95	**0.98**	0.76	0.95	0.95	0.67	0.42
DDoS	0.7	0.39	0.96	0.96	0.49	0.91	**0.96**	0.9	0.41
DoS GoldenEye	0.77	0.67	0.99	0.97	0.43	0.97	**0.99**	0.98	0.23
DoS Hulk	0.31	0.4	0.91	0.95	0.94	0.95	**0.95**	0.87	0.81
DoS Slowhttptest	0.41	0.42	0.98	**0.99**	0.65	0.98	0.98	0.97	0.44
DoS slowloris	0.41	0.48	0.93	0.94	0.62	0.94	**0.95**	0.9	0.48
FTP-Patator	**1**	0.99	1	1	1	1	1	0.94	0.38
Heartbleed	**1**	1	1	0.89	0.68	1	0.96	1	0.79
Infiltration	0.89	0.89	0.85	0.89	0.63	0.78	0.84	0.89	0.41
PortScan	0.41	0.81	1	1	0.45	1	**1**	0.86	0.42
SSH-Patator	0.38	0.44	0.95	0.96	0.71	0.95	**0.95**	0.77	0.53
Web Attack	0.72	0.83	0.96	0.96	0.67	0.92	**0.96**	0.91	0.41

Table 3

When looking at the results, it can be noted that the Random Forest, KNN, ID3 and Adaboost algorithms have achieved a success rate of more than 90% in almost all types of attack detection. In the four algorithms, the most successful was ID3, which completed seven of the 12 tasks and scored the highest. In fact, ID3 scored highest on at least one of the seven tasks (DDoS, DoS GoldenEye, DoS Hulk, PortScan, SSHPatator, and Web Attack). However, it has a shorter processing time and is superior to other algorithms. In the 12 tasks, the naive Bayesian algorithm with the smallest F-measure ranked last among the six tasks. However, it is also necessary to mention QDA here because QDA scores were very close to those of naive Bayes in almost all of these six tasks.

Figure 12 shows the description of these 12 types of network attacks by these 9 algorithms using F-score as the measurement standard. The above conclusions can be clearly seen and validated in the figure.

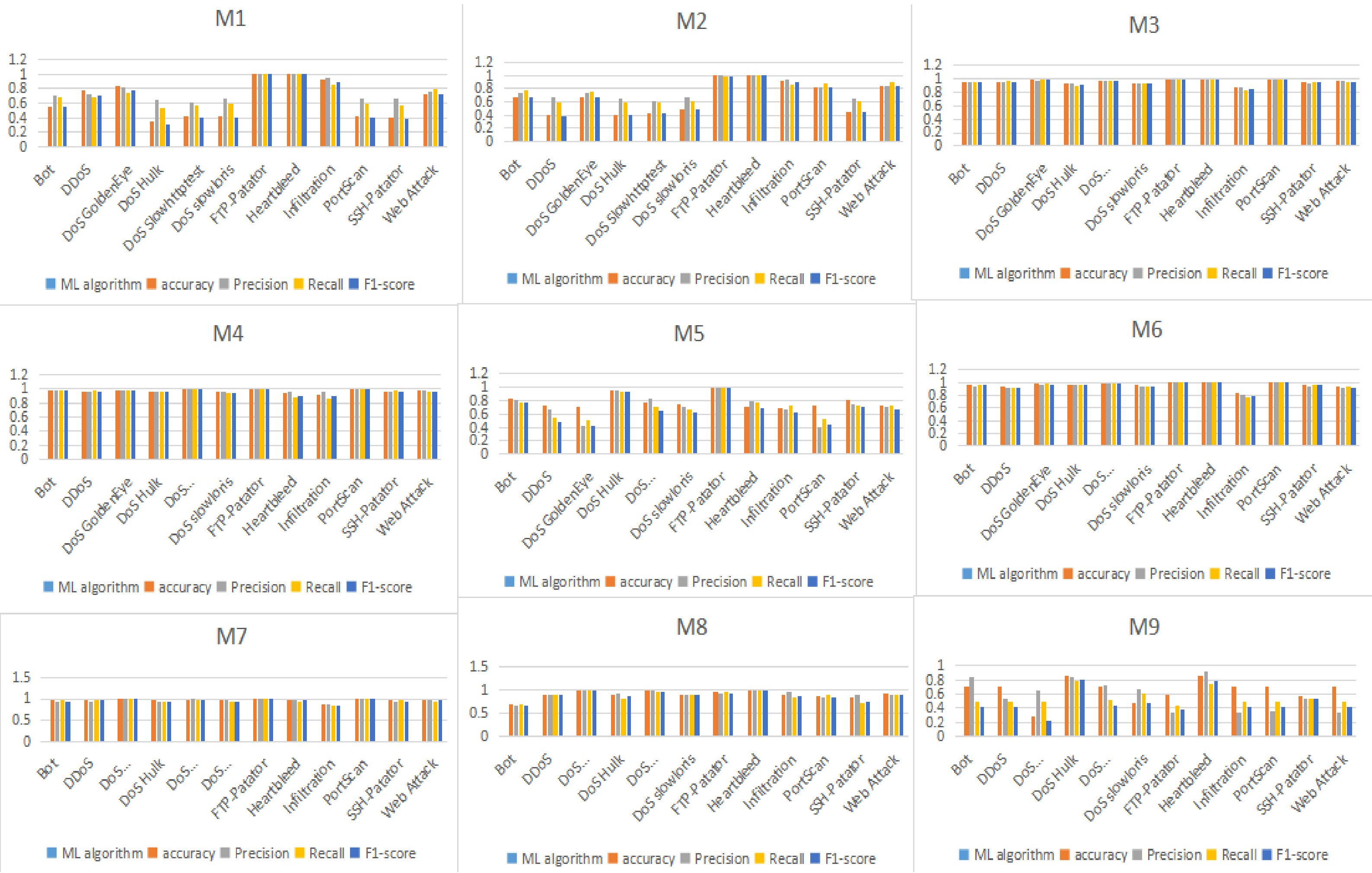
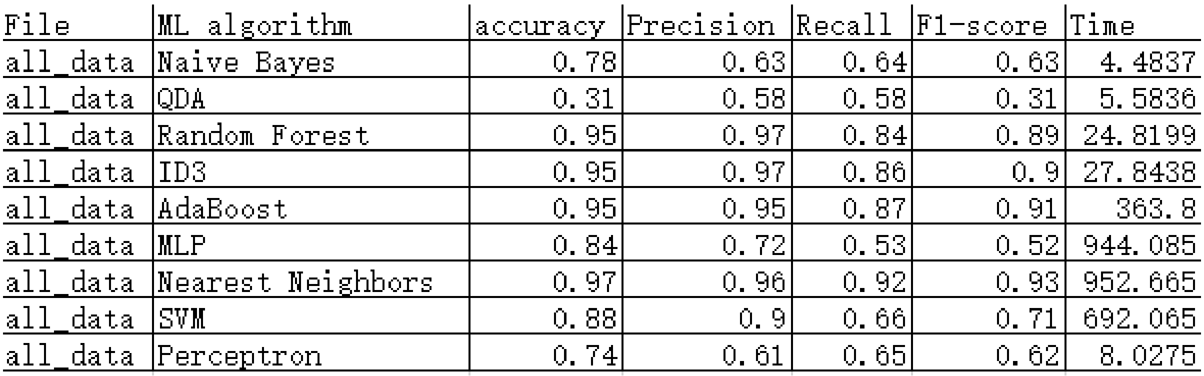


Figure 12

2. Using Two Groups: Attack and Benign: In this section, the entire dataset is used as a single dataset file. All attacks contained in this file are collected under a single generic name, "Attack." Nine different machine learning methods were applied to this data set.In this approach, two approaches are used, using the features created for the attack file in Approach 1, and Table 4 shows the results obtained using the 18 attributes extracted from ALL_data. As can be seen from the table, the algorithm with the highest performance is KNN, with an F1-score of 0.93. After that, AdaBoost and ID3 were only 0.90 and 0.91. However, ID3 is much faster than Adaboost, so it takes precedence over this feature. The algorithm with the lowest score is QDA, with a score of 0.31. The QDA score was about 0.4 points lower than the closest algorithms (naive Bayes and MLP). In this method based on tag value, it can be seen that KNN is the optimal algorithm.



File	ML algorithm	accuracy	Precision	Recall	F1-score	Time
all_data	Naive Bayes	0.78	0.63	0.64	0.63	4.4837
all_data	QDA	0.31	0.58	0.58	0.31	5.5836
all_data	Random Forest	0.95	0.97	0.84	0.89	24.8199
all_data	ID3	0.95	0.97	0.86	0.9	27.8438
all_data	AdaBoost	0.95	0.95	0.87	0.91	363.8
all_data	MLP	0.84	0.72	0.53	0.52	944.085
all_data	Nearest Neighbors	0.97	0.96	0.92	0.93	952.665
all_data	SVM	0.88	0.9	0.66	0.71	692.065
all_data	Perceptron	0.74	0.61	0.65	0.62	8.0275

Table 4

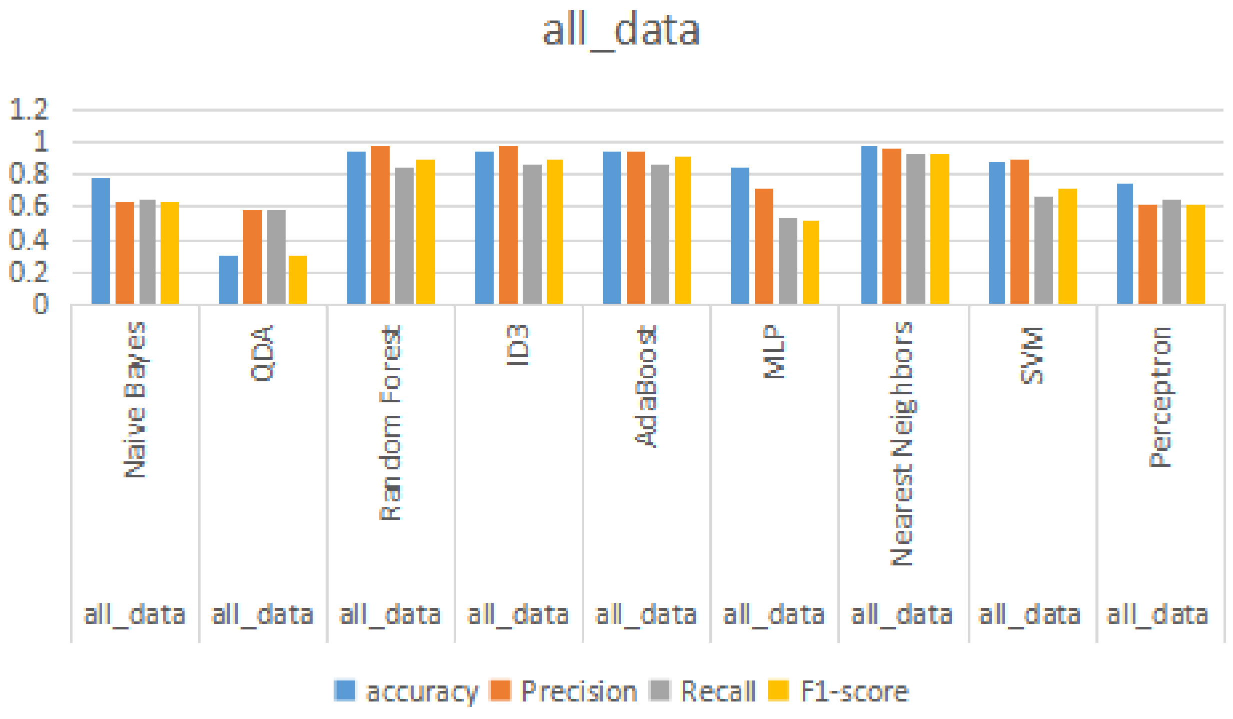


Figure 13

From Table 13, we can clearly see the comprehensive performance of each in the aggregate data set and the sensitivity of each network attack. To be sure, QDA is not a good general algorithm for detecting network attacks using machine learning.

XI. Conclusion and Future Work：

The increasing frequency of network and host intrusion has seriously affected the security and privacy of users. Researchers have studied a wide range of intrusion detection solutions. Based on the analysis of the principle of network attack, this paper looks for the commonness and correlation of network attack from the feature selection, and tries to use the machine learning method to play a role in intrusion detection. Further, we measured the performance of 9 commonly used supervised learning algorithms in network attack detection, and analyzed the most suitable and the least suitable detection algorithms. The analysis performed shows that if one algorithm performs well in detecting attacks, it may not perform as well in detecting other attacks. Therefore, nine algorithms were used to detect the data of each attack type, and a comprehensive evaluation was proposed.

Performance analysis of various machine learning algorithms has been done. The performance of single network attack and hybrid network attack data is compared. We have shown that even if an optimal AD hoc collection is sufficient to analyze an attack, it is not applicable to analyze other attacks. Therefore, because attack behavior varies, you need to define the optimal subset of characteristics and the appropriate technique for each type of attack. The difficulty of using machine learning techniques to detect low frequency attacks on network data sets has been described. It motivates researchers to work on other solutions. It provides future research directions for researchers to explore more effective attack detection solutions. The description of the existing literature is based on the data set of labels as the date to summarize our observation. Therefore, there are limitations in the selection of detection algorithm training. This article also does not use all techniques to evaluate performance to ensure repeatability of the results. This is still a limitation of my project and I am very eager to improve this as a future work.

In the future, I hope to have an in-depth understanding of the behavior of network attacks in terms of features . On this basis, I hope to have an in-depth understanding of the performance of machine learning algorithms, especially unsupervised learning algorithms that do not rely on tags, in detecting network anomalies.

[1] S. Agrawal and J. Agrawal, “ Survey on anomaly detection using data mining techniques,” Procedia Comput. Sci., vol. 60, pp. 708– 713, Dec. 2015.
[2] N. F. Haq et al., “ Application of machine learning approaches in intrusion detection system: A survey,” Int. J. Adv. Res. Artif. Intell., vol. 4, no. 3, pp. 9– 18, 2015.
[3] M. Ahmed, A. N. Mahmood, and J. Hu, “ A survey of network anomaly detection techniques,” J. Netw. Comput. Appl., vol. 60, pp. 19– 31,Jan. 2016.
[4] KDD. (1999). KDD Cup 1999 Data. [Online]. Available: http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html
[5] A. L. Buczak and E. Guven, “ A survey of data mining and machine learning methods for cyber security intrusion detection,” IEEE Commun. Surveys Tuts., vol. 18, no. 2, pp. 1153– 1176, 2nd Quart., 2015.
[6] S. Behal and K. Kumar, "Characterization and Comparison of DDoS Attack Tools and Traffic Generators: A Review," IJ Network Security, vol. 19, no. 3, pp. 383-393, 2017.
[7] A. Gharib, I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "An evaluation framework for intrusion detection dataset," in Information Science and Security (ICISS), 2016 International Conference on, 2016, pp. 1-6: IEEE.
[8] M. M. Najafabadi, T. M. Khoshgoftaar, C. Kemp, N. Seliya, and R. Zuech, "Machine learning for detecting brute force attacks at the network level," in Bioinformatics and Bioengineering (BIBE), 2014 IEEE International Conference on, 2014, pp. 379-385: IEEE.
[9] T. Ylonen and C. Lonvick, "The secure shell (SSH) protocol architecture," 2070-1721, 2005.
[10] “ Slow Read DoS Attack.” Istanbul Technical University, 07 Sep 2013.
[11] “ sklearn.preprocessing.LabelEncoder,” 1.4. Support Vector Machines - scikit-learn 0.19.1 documentati on.
[12] J. Brownlee, “ Feature Importance and Feature Selection With XGBoost in Python,” Machine Learning Mastery, 10-Mar-2018.
[13] R. Alshammari and A. N. Zincir-Heywood, "A flow based approach for SSH traffic detection," in Systems, Man and Cybernetics, 2007. ISIC. IEEE International Conference on, 2007, pp. 296-301: IEEE.