A Survey of Machine Learning Based Packet Classification

Yu Liu
The Institute for Computing, Information
and Cognitive Systems (ICICS),
University of British Columbia
Vancouver, BC V6T 1Z4
Canada
Yuliu98@mss.icics.ubc.ca

ABSTRACT
In this paper, I will categories and analysis different approaches to classify different Internet traffics using Machine Learning (ML) technic. The traffic classification can be used as an important tool to detect intrusion detection. And it also can be used by network operator to control the network. However it opens a topic related to protection of personal information. After realizing the advantage and disadvantage of packet classification, I will briefly introduce three classification methods and related researches. The classification methods are port-based, payload-based and statistical-based classification. And ML is a well-known technic used in statistical-based classification. After a brief introduction of ML, I will focus on analysis different researches related to traffic classification based on ML.

There exists a paper related to traffic categorization using ML [26]. But the researches mentioned in their paper are not up-to-date. My work extended their research to introducing different approaches to classify encrypted traffic such as Skype, GTalk, and SSH.

General Terms
Performance.

Keywords
Classification, DPI, Machine Learning, Traffic analysis, Application Identification

1. INTRODUCTION
In the 21st century, the number of Internet users increased dramatically. The users applied several Internet applications such as WWW, FTP, peer-to-peer-based software, web media, messaging, email, VOIP etc. This led to fast increments of Internet traffic. The classification of Internet traffic offers three main functions to the network administrator, internet service provider (ISPs), and governments: First, the packet classification can be used in the intrusion detection system (IDS) to detect the patterns of denial of service (DoS) or other malicious attacks. It can also be used by the administrator to identify and control the network applications when needed. Second, it can be used by the ISPs to monitor the network flows, diagnose the network to find faults, properly allocate the bandwidth to applications, and ensure the performance of the applications and services running on the networks. Third, it can be used by governments to do "Lawful Inspection" (LI) of the payload of packets, to obtain user information. Just like how telephone companies offer to monitor telephone calls to the government, ISPs provide the LI services to the governments. [1, 2, 3]

We know the importance of the characterization of Internet traffic. Now we need to understand the barriers for packet classification. Internet traffic characterizing has been a challenge over the past few years. [4] It requires in-depth understanding of the sophisticated network protocol structure, because there are many various types of traffic for the ISPs, as well as a large volume of stream flows. With the bandwidth and number of services increasing, users can perform much more complicated activities than before. A broadband user can perform tasks such as VoIP, shopping and banding online, peer-to-peer-based file and video sharing among peers, and much more complicated functions that were previously known by dial-up users.[5] The complexity will increase when using different wireless technologies such as the 4G Long Term Evolution (LTE) system and the Wi-Fi system.[6]

1.1 Port-Based Classification
The simplest way to classify Internet traffic is by using UDP or TCP port numbers. The reason is that some traffic uses well known port numbers, and the port numbers can be found on Internet Assigned Numbers Authority (IANA) [7]. For example, HTTP uses port 80, POP3 uses port 110, and SMTP uses port 25. We can set up rules to classify the applications that are assigned to the port numbers. However, many researches claim the port-number-based classification is not sufficient. [8, 9, 10, 11, 12] Moore and Papagiannaki claimed the accuracy of port-based classification is around 70% during their experiment. [13] Moreover, Madhukar and Williamson claimed in their research that the misclassification of port-based classification is between 30% and 70%. [14] The main reason for choosing static port numbers is to make the packet more able to go through the server firewalls. Many recent applications try to avoid the detection of firewall by hiding the port numbers. Some of the other applications use dynamic port numbers instead of static ones. And servers which share the same IP address will use un-standard port numbers.

1.2 Payload-Based Classification
Another approach to classify packets is to analyze the packet payload or use deep packet inspection (DPI) technology. They classify the packets based on the signature in the packet payload, and it has been touted as the most accurate classification method, with 100% of packets correctly classified if the payload is not encrypted [13]. The signature is unique strings in the payload that distinguish the target packets from other traffic packets. Every protocol has its distinct way of communication that differs from other protocols. There are communication patterns in the payload of the packets. We can set up rules to analyze the packet payload to match those communication patterns in order to classify the application. For example, according to [15], "MAIL FROM","RCPT TO" and "DATA", as in Figure 1, are the commands that appear in the payload of SMTP packets. Therefore, we can create rules to match the plain text in the packet payload to classify SMTP packets. The problems include: users

may encrypt the payload to avoid detection, and some countries forbid doing payload inspection to protect user information privacy. Furthermore, the classifier will experience heavy operational load because it needs to constantly update the application signature to make sure it contains the signature of all the latest applications.

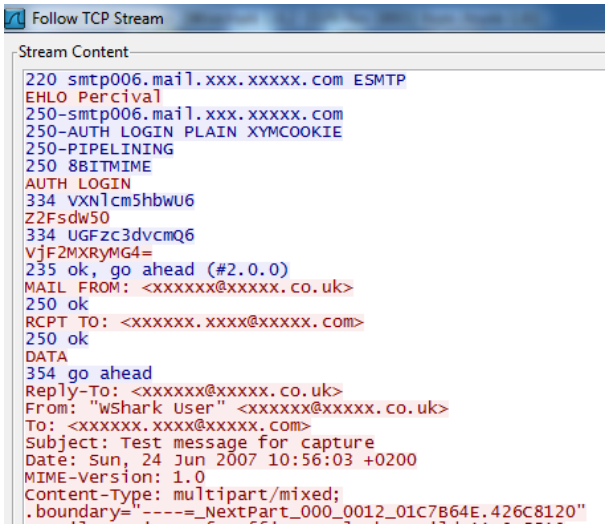

Figure 1- SMTP TCP Stream

1.3 Statistical-Based Classification

Due to the limitation of port-based and payload-based classification, the recent research focuses on the use of transport and flow layer behavior statistics for packet classification. [16, 17, 18] This approach uses a set of sample traffic trace to train the classification engine to identify future traffic based on the application flow behaviors, such as packet length, inter-packet arrival time, TCP and IP flags, and checksum. Their target is to classify traffic with similar patterns into groups, or classify traffic into individual application. However, the accuracy of classifying encrypted traffic using a statistical-based approach is relatively low, varying from 76% to 86% with false positive rate between 0% and 8% base on different rule settings. [18, 19, 20, 21] Many researchers use Machine Learning (ML) to perform statistic-based classification. The reason to choose ML is because it can automatically create the signatures for the application and automatically identify the application in the future traffic flow. Another reason to choose ML is it has the ability to automatically select the most appropriate features to create the signature. The ML technique consists of many steps: 1) Define the features associate with the traffic. The features may include packet size or inter-packet arrival time. 2) Assign a application type to instances. 3) Choose sample application traces to train the classification engine to generate rules, and uses the ML algorithms to classify future traffic. A lot of research exists about the relationship between the statistical properties and traffic types. Vern Paxson studies the statistical relationship between data bytes, duration and some TCP applications such as Telnet, NNTP, SMTP, and FTP. [22] Christian Dewes *et al*. studied the relationship between flow duration, packet inter-arrival time and packet size with Internet chat room traffic. Some other researchers [24, 25] studied the packet length and packet inter-arrival times by analyzing the traces from different applications.

2. BACKGROUD OF MACHINE LEARNING

In this section, I will introduce some fundamental concepts related to machine learning. First, I will introduce the classification metrics, followed by a short history of using ML in traffic classification. Then I will focus on the input out of ML technology and different types of learning methods.

2.1 Classification Metrics

Classification engines take unknown trace files as input, and then it identifies the existence of the targeted type in the trace file. The output should be 'yes' if the traffic belongs to a target type, and 'no' if the traffic does not belong to a target type. The key to distinguishing a good classification technique and a bad one is the classification accuracy. In this paper, we will consider the following metrics: false positive, false negative, true positive, true negative recall, and precision. Thuy T.T. Nguyen and Grenville Armitage have given their definitions [26].

- False Negatives: Percentage of targeted type incorrectly classified as other.
- False Positives: Percentage of other types of traffic classified as targeted type.
- True Positives : Percentage of traffic correctly classified as targeted type
- True Negatives : Percentage of other traffic correctly not classified as targeted type
- Recall: Percentage of traffic correctly classified as targeted type.
- Precision: Percentage of those instances that truly have targeted type, among all those classified as targeted type.

2.2 History of Machine Learning

In 1992, Shi [27] described machine learning as making a machine to independently learn new knowledge and skills, and to make a machine that uses the knowledge it has learned. ML is a useful and powerful tool to discover useful patterns in large data sets. It automatically learns the difficult patterns from large data sets and makes correct decisions based on the learned rules.

ML has been used in many fields, such as search engine, medical diagnostics, face recognition, marketing, sales diagnostics, text-based recognition, image recognition and so on. In 1990, Bernard Silver used the machine learning technique to maximize the call completion in a circuit-switched telecommunication network [28]. This work is the first to use ML in the telecommunication network. In 1994, Jeremy Frank completed the first research using ML in traffic flow classification for IDS. Since then, the ML begins to be used in many other fields, especially in packet classification.

2.3 The Input and Output of ML Technology

In general, the purpose of ML technology is to find patterns in the sample data sets.

The input of ML is equivalent to the datasets of instances. An instance is the individual sample dataset, which has similar

features or attributes. An instance can be explained as the packets from the same flow. In another word, users can choose the trace file that contains the target type flows.

The output of ML is the patterns and rules that ML learned from the sample dataset. The output can be different when using different ML approaches.

2.4 Different Type of Learning

Witten and Frank defined four types of learning approaches in [29].

- Classification
- Clustering
- Association
- Numeric prediction

Classification is the method to train the machine with sample datasets, which means the traffic type is known to the user, in order to build classification rules. Then, the machine uses the rules to classify unknown datasets. Clustering is the method to find similar patterns among different traffic types, and group the traffic that has similar patterns in the clusters. This method does not require supervision. Association is a way to detect the relationships between attributes. Numeric prediction is a way to find the total number features appearing in the dataset. This method is useful when finding important features or attributes. This method is supervised learning.

The main difference between supervised learning and unsupervised learning is that supervised learning needs training datasets to train the machine, whereas unsupervised learning does not require a training phase.

There are only two methods, classification and clustering, that are wildly used in the packet classification. Hence, I only discuss those two approaches in the following section.

2.4.1 Classification

Classification learning is a supervised approach. It is a method to classify the sample dataset into different traffic type based on the rules learned during the training phase. The rules are generated by the learning algorithm from the training phase. The training phase is using the traces only containing a target traffic type as input for learning algorithm, and then the learning algorithm produce a set of rules to classify future target traffic type from large amounts of unknown traffic.

The classification learning is trying to find the relationship between input dataset and output rules. The output the classification learned can be represented by a decision trees or classification rules.

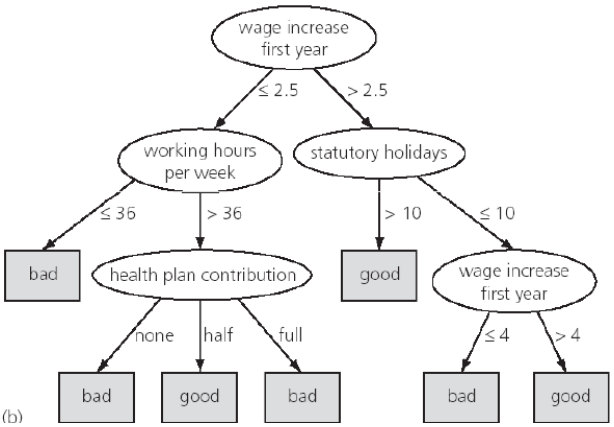

Figure 2-Sample Decision Tree for Labour Data

As shown in Figure 2, every packet needs to compare its value with a particular attribute. Every packet will go through the decision tree and be classified based on the results in the leaf node.

```
If tear production rate = reduced then recommendation = none
```

Figure 3-Sample Classification Rule from Contact Lens Problem

The classification rule consists of two parts, antecedent or precondition and consequent or conclusion. The precondition can be seen as a path in the decision tree, but unlike the decision tree, the tests are logically 'AND'ed together. This means the packet has to pass all the tests in order to be classified as a targeted traffic type. The conclusion is the targeted traffic type, similar to the leaf node in decision tree.

The classification learning consisted of two main steps.

- Training: This step utilizes dataset training, in which the traffic type is known to the user, to form a set of rules to distinguish the target traffic type with other types.
- Classifying: The classification is using the rules from the previous step to classify unknown datasets.

There is a way to improve the training process. The user trains the machine with traffic belonging to the target traffic type, as well as traffic not belongs to target traffic. This will enhance the rules generated from the training phase.

There exists many classification learning methods. The main difference is they use different training algorithm and different optimization algorithms in their research.

2.4.2 Clustering

Unlike Classification learning, clustering is an unsupervised approach and does not have a training phase. The goal of clustering is to group the packets that have similar patterns. There are three situations when grouping the packets.

- The packet can be put into a single group if the group is exclusive.

- The packet can be put into many groups if the packet matches the patterns in multiple groups.
- The group can be probabilistic, so the packet can belongs to a group with a fixed probability

The clustering methods consist of three main methods: k-means algorithm, incremental clustering, and probability-based clustering. K-means is a method to groups N packets into K groups. The packet with the nearest mean to a group should be classified into that group. Incremental clustering is a method to form a hierarchical structure of groups. The group can be divided into sub-groups in incremental clustering. The probability-based approach is the probability of a packet being assigned to a group [29].

2.4.3 Comparison between Classification and Clustering

Packet classification is the process of associating the packets, which are mixed with random packets, with an application that generates the packets. The most challenging task is finding the relationship between the source packet and the packet generated by the targeted application.

The classification approach needs a training phase to associate the patterns with the application. The training phase needs a sample dataset that has been classified into targeted traffic. Hence, the classification approach works better when classifying one application or groups of applications. However, this approach has a limitation. The classifier has to be trained with all of the patterns appearing in the traffic generated by the application. So the performance of the classification approach greatly depends on the training phase. If the training phase covers all of the possibilities, then the accuracy is high. And if the training phase did not cover all possibilities, then the accuracy is low.

An advantage for the cluster approach is it can recognize the patterns in the dataset automatically. However, this approach can only characterize the packets into different groups. It can't label the patterns without the help of the user. Another advantage of the cluster approach is that it is easier to classify previously unknown applications. It cannot directly identify the application, but it can detect new types of traffic to help users identify new applications. Another disadvantage is that the classification can only identify a group of applications sharing the same patterns. One application can be classified into multiple groups. The worst case is that the application is classified into none of the groups. In the case of the packet, they can be classified into different groups. It is difficult to map the groups with the original application.

3. THE IMPLEMENTATION OF MACHINE LEARNING-BASED PACKET CLASSIFICATION

In this section I will describe and categorize different research approaches using machine learning to classify traffic. There are four main categories: the clustering approach, the classification learning approach, the hybrid approach, and comparison.

3.1 Clustering Approach

3.1.1 Expectation Maximization-Based Flow Clustering

McGregor et al. [31] has done the first research to using expectation maximization algorithm to classify traffic. This approach classifies traffic into different applications using similar patterns. This work focuses on the following traffic types, FTP traffic, HTTP traffic, DNS traffic, IMAP traffic and NTP traffic. The Auckland-VI trace file they used is generated over 6 hours. The flows in the trace file are full-flow, except the flows are over 6 hours long.

The traffic file was divided into different groups using expectation maximization algorithm. Then, the classification rules are created based on the groups. The rules are used to identify the features that have the lowest influence on the classification, and then remove those features. This process needs to be repeated several times.

3.1.2 Simple K-Means Based Clustering

Bernaille et al. [16] proposed a method to use simple k-means algorithm to classify the TCP flows into different applications. The goal of this method is to classify traffic as early as possible. Early detection is achieved by inspecting only the first few packets of a flow. The reason to check only the first few packets is that the distinct communication commands and messages exchanged are usually in the first few packets.

Different from other approaches, the training phase of the k-means based approach is done offline. The trace file is generated over one hour, containing several application traffics. The trace file is classified into different groups based on the packet size of the first few packets. The number of groups can be generated using K-means algorithms. The goal is to find the minimum Euclidean distance between the flow and defined groups. The output of this approach is a little bit different. It consists of two parts: the descriptions of groups and the applications associated with the groups.

They claim that using the first 5 packets of a flow can correctly identify more than 80% of the TCP flow. However, the classifications of POP3 flows are not accurate, and they were classified as NNTP and SMTP.

This is a relatively fast and easy approach. However, the classification cannot be accurate if the first few packets in the flow are missing. The other disadvantage is that the classification accuracy will drop significantly if the application did not form a group in the training phase, for example POP3 flows were not correctly classified.

3.1.2.1 Similar approaches

Erman et al [32] proposed a similar approach using simple k-means algorithm to grouping the flows, and uses Euclidean distance to calculate the distance between two flows. But they focus on the popular protocol such as P2P and web traffic, and they only use uni-directional flow information to create the groups.

In the training phase, they used the traces that have been labeled based on the signature and payload information. They also use k-means to get the minimum number of groups. Three types of training traces were considered in their approach: the trace from the client to server, the trace from the server to client, and a mixed trace.

The results show that the flow accuracy and byte accuracy are better when the k increased from 25 to 400. In general, this approach has the highest accuracy when classifying the trace from the server to client, 95% of flow accuracy and 79 byte accuracy. The mixed trace has 94% of flow accuracy and 67% byte accuracy. The client to server trace has high flow accuracy, which is 94%, but it has the lowest byte accuracy which is 57%.

This algorithm is using the information from TCP protocol. Therefore it can only classify TCP traffic. The statistic patterns include the following categories: flow duration, number of packets in the flow, and number of bytes. The duration is the time difference between the first packet and the last packet in each flow. The number of packets can be obtained using the last sequence number and the acknowledgement number. The number of bytes can be found in in ACK packets. In this research, they assume the packet loss rate is 0. And they have shown that the accuracy of classification will be affected by packet loss rate, significantly. The number of bytes and flow duration has less impact on the classification accuracy.

3.2 Classification Learning Approach

3.2.1 Signature-based approach

Roughan et al. [33] have done research about using quadratic discriminant analysis, nearest neighbours and linear discriminate analysis to classify different applications. The authors used various features to set up classification rules. The features are categorized into five categories. In *packet level* feature, the author uses features such as packet size and root mean square size. For *flow level* features, they use features such as mean flow duration and mean number of packets. In *connection-level*, the features may include advertisement window sizes as well as the features in the flow-level. *Intra-flow /connection* features include inter arrival time, latencies loss rate and so on. For multi-flow category, the features are more complicated than other categories. This category is more useful for P2P applications which use multiple connections to the system end-system to download the files. Among all of the features mentioned in their paper, they use duration and average packet length as the most valuable features.

The goal of their approach is to find the feature vector to generate the rules for classifier with a given number of classes, features, and training datasets. They considered 3 types of classes: 3 classes, 4 classes, and 7 classes. They evaluate their approach based on the error rate. The author listed their testing results in Table 1. From Table 1, we know that 7-class has the highest error rate, varying from 0.94 to 0.126 whereas 3-class has the lowest error rate, changing from 0.025 to 0.034 when applying different algorithms. The 4-class is more stable compared to the other two approaches, and the error rate stays around 0.056.

Table 1: The cross-validation results

Algorithm	Error Rate		
	4 class	3 class	7 class
LDA	5.6%	3.4%	10.9%
1-NN	7.9%	3.4%	12.6%
3-NN	5.1%	2.5%	9.4%
5-NN	5.6%	2.5%	9.9%
7-NN	5.6%	2.8%	9.7%
15-NN	6.2%	3.4%	11.4%

3.2.2 Naive Bayes Estimator-Based Approach

Moore and Zuev [34] used the Naïve Bayes estimator to classify traffic into different applications. Different from other approaches, they used the dataset that has been classified to make their testing results more accurate. They selected 248 features in the training phase. They pre-defined the classification type associated with different applications. The classification type includes: BULK, MULTIMEDIA, GAMES, ATTACK, P2P, WWW, SERVICES, MAIL, DATABASE, and WWW. Their focus is not on individual applications but the category of applications.

They use *trust* and *accuracy by bytes* to evaluate their testing results. Trust represents how well you can trust the classification. Accuracy-By-Bytes is the percentages of flow bytes that were correctly classified. They used four approaches to test their results. When they used the simple Naïve Bayes method, they claimed an average of 65.26% of their flow was being successfully classified, and 83.93% of the bytes were correctly classified. WWW and MAIL had the highest truest, over 90%, whereas ATT and P2P had the lowest, below 5%. Later, they used other technics, such as kernel density estimation, FCBF pre-filtering, to improve the performance. The simple Naïve Bayes method performs the lowest in the average flow classification success rate, staying below 65%. The Naïve Bayes method, with kernel density estimation technique after FCBF pre-filtering, performed the best with a minimum 93.73% average success rate. In general, their approach performed better when classifying WWW and MAIL with average trust rate over 90%. However, the classification for P2P traffic had the lowest trust rate, which ranged from 4.96% to 53.50%.

3.2.2.1 Similar Approaches

Nguyen and Armitage [35] used a sliding window to divide the flow into many sub-flows. They used the features calculated from the sub-flows to train the classifier. Their approach also used Naïve Bayes, but they targeted the game traffic. They claim their approach can quickly detect game traffic and save resources because they only use the sliding window to select the recent limited number of packets. The advantage of their research is that the classifier can classify traffic at any time. It does require a full flow to detect targeted traffic type.

They chose inter-arrival time, packet length variation, and packet length in each direction as the features. These features were calculated by the packets in the sliding window. The following procedure is the scenario to calculate the features. They get multiple sub-flows of targeted traffic types. The sub-flows should have statistical value, which means sub-flows need to be taken at different times. Then, the features will be generated from each sub-flow. After getting the features, the classifier will be trained using the features of sub-flows.

They trained the classification engine with three methods. One method is using a full-flow and a given window size. Another way is using individual sub-flows to train the classifier. The latter way is using sub-flow to train the engine. The results were evaluated using recall and precision. From their testing results, it shows that the highest precision is more than 98% and highest recall is more than 95%.

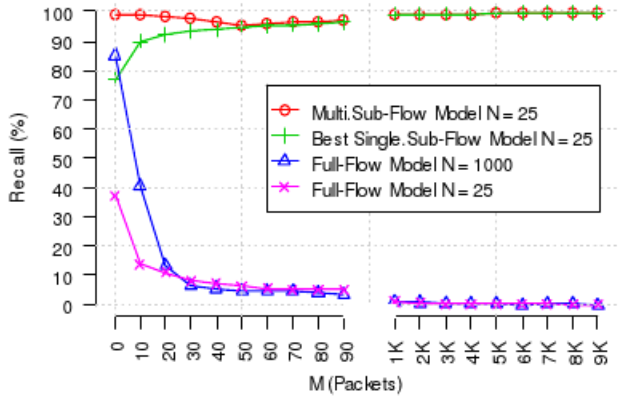

Figure 4-Recall Comparison of Full-Flow and Sub-Flow Training of the Classifier

However, the work is mainly focusing on the game traffic classification. It fails to evaluate the performance with popular traffic type such as P2P, SSH, WWW and so on.

3.2.3 Fingerprints-Based approach

Crotti et al. [36] proposed a classification method using packet length, the order of packet and inter-arrival time. These features have been used by other researchers, but the difference is the structured features are called *fingerprints*. The fingerprint is a more efficient and structured way to organize the features. They used normalized threshold algorithms in their research.

In the training phase, they use the pre-labeled sample dataset to train the engine to create the protocol fingerprints, which is a PDF vector. The procedure of classification starts with generating the *anomaly score* of the unknown flow. The anomaly score is used to describe the "distance" of the unknown traffic with the PDF of the target protocol. It shows that the packets in the flow that have a higher score will have a higher possibility to belong to a targeted protocol.

Table 2-Hit Ratios and False Positive Ratio

Protocol	Hit Ratio	FP
HTTP	91.76%	6.38%
SMTP	94.51%	3.06%
POP3	94.58%	3.08%
Other	90.64%	N/A

The author listed their test results in Table-2. From the table, we can see that the overall accuracy is more than 90% with the false positive rate lower than 6.5%. They perform relatively better than other approaches in classifying individual protocol.

Although the performance is better, their approach still has some issues needing to be addressed. Their testing process only considers the situation where packets are not lost or reordered in the flow. Also, it did not consider the situation where the traces lost the first few packets of a flow.

3.2.3.1 Similar Approaches

Patrick Haffner et al. [37] proposed a similar approach to use fingerprint to train the classification engine, but they used three popular ML algorithms: Naïve Bayes, AdaBoost, and Maximum entropy, to generate the application signatures. Then, they compared the performance of the three approaches. They claim their research is the first in application level classification. The most interesting point is their approach identified encrypted packets, such as SSH and HTTPs. This approach identified the encrypted using the first few packets of plaintext handshake. This pointed out that their approach will fail to classify encrypted packets if the flow misses the first few packets.

During the training phase, they use pre-classified datasets to train the classifier. In their approach, they only use the first 64 Bytes in the payload stream to generate the signature. The results from the training algorithms are the signature of the application and can be used to classify targeted applications.

They use error rate, precision and recall to evaluate the performance of three algorithms. Their application type consist of FTP, SMTP, POP3, IAMP, HTTPS, HTTP and SSH. From the results in Figure-5, we know that AdaBoost algorithms perform better among all of the methods. In general, their error rate, precision and recalls are 0.51%, 99% and 94% respectively. It also shows that Naïve Bayes has the lowest performance and its error rate is 4 to 12 times larger than AdaBoost on the 8/2400 training dataset.

However, their approach still has the problem of losing the first few packets of flow. The accuracy of classifying SSH and HTTPs packets will decrease significantly when missing the first few packets or changing the encryption algorithm.

Application	Training set	Training User Time	Algorithm	Error Rate in %	Precision	Recall	|w|
ftp control	8hr	4h53m17.86s	AdaBoost	0.016	0.996	0.971	612
smtp	8hr	7h33m58.07s	AdaBoost	0.031	0.998	0.999	480
pop3	8hr	5h44m36.53s	AdaBoost	0.039	0.995	0.999	356
imap	8hr	12m2.16s	AdaBoost	0.000	1.000	0.999	189
https	8hr	7h28m39.37s	AdaBoost	0.258	0.992	0.946	271
http	8hr	1h0m17.06s	Maxent	0.508	0.990	0.999	5666
ssh	8hr	20m54.00s	AdaBoost	0.001	1.000	0.866	74

Figure 5-Best Classification Results

3.3 Comparison

3.3.1 Comparison of C4.5, Support Vector Machine, Naïve Bayesian and RIPPER

Riyad Alshammari and A. Nur Zincir-Heywood [39] proposed a ML-based classification for encrypted traffic. They only considered SSH and Skype in their experience, but they claim their system can be used to classify any type of encrypted traffic. They only use the statistic value in flow level. Also, they claim they are the first researchers to consider the robustness of classifying encrypted traffic. The reason to choose Skype and SSH is that Skype is a well-known application that generates robust encrypted packets. And SSH is one of the most common encrypted traffic sources.

The authors employed 4 different approaches, Support Vector Machine, Naïve Bayesian, RIPPER and C4.5. The features only consider the statistics of flow level features (Table-3). The detailed training process for each algorithm can be found in [40, 41, 42 43, 44].

Table 3-Flow Based Features Employed

Protocol	Duration of the flow
Number of packets in forward direction	Number of bytes in forward direction
Number of packets in backward direction	Number of bytes in backward direction
Min forward inter-arrival time	Min backward inter-arrival time
Standard deviation of forward inter-arrival times	Standard deviation of backward inter-arrival times
Mean forward inter-arrival time	Mean backward inter-arrival time
Max forward inter-arrival time	Max backward inter-arrival time
Min forward packet length	Min backward packet length
Max forward packet length	Max backward packet length
Standard deviation of forward packet length	Standard deviation of backward packet length
Mean backward packet length	Mean forward packet length

They evaluate their performance by detection rate and false positive rate. Their results are based on testing the 5 algorithms with 3 public traces, Dalhousie, AMP & MAWI, DARPA99. From their testing results, it shows that overall C4.5 and RIPPER have a better performance. But C4.5 has the best performance when classifying Skype traffic. The DR of Skype traffic can reach as high as 98%, with around 8% FPR. In this research, the lowest DR for C4.5 is 83.7%, with 1.5% FPR. In the best situation, the DR can reach 97% with 0.8% FPR.

There used many different public traces in their research to make the results more reliable. However, this approach is focusing on classify SSH and Skype packets, and did not consider the classification for WWW, FTP, P2P, other encrypted application such as GTalk, IRC encrypted traffic. Reader cannot compare their results with other testing results directly.

3.3.2 Comparison of C4.5, Bayesian Network, Naïve Bayes using Discretisation (NBD), Naïve Bayes using Estimation (NBK), and Naïve Bayes Tree

Williams et al. [45] compared and analysis 5 machine learning algorithms: Naïve Bayes using Discretisation (NBD), Naïve Bayes using Estimation (NBK), C4.5, Bayesian Network, and Naïve Bayes Tree. They suggested that the performance of ML algorithms is not only determined by accuracy alone, but also computational speed and the time to map classification rules. They state that the redundant feature sets can be a problem for classification efficiency. They focus on choosing the more valuable feature sets to maximize the packet classification efficacy.

They used three public NLANR datasets in their experiments. They selected 22 flow features, which they call it "*full feature set*". And they use Correlation-based Feature Selection (CFS) and Consistency-based Feature selection (CON) to find reduced feature sets.

Their results are evaluated with accuracy, precision, recall, computational performance. Using the full 22 features, most of the algorithms have more than 95% accuracy except NBK, which achieves more than 80%. Using the reduced feature sets of 8 (CFS) and 9 (CON), the overall accuracy almost stay the same as the results using 22 features. The most significant variations are the accuracy for NBD and NBK, between 2% to 2.5%, with 9 (CON) features. The computational performances of these algorithms vary significantly. The paper shows that C4.5 algorithm is the fastest among all algorithms. The highest speed can reach 54,700 per second. The slowest speed is NBK algorithm, which followed by NBTree, BayesNet, NDB and C4.5. For building time, they found NBTree has the slowest speed. The rest the algorithms have similar performance with NBK performed a little bit better.

Although they found the C4.5 perform better in general, but they did not give the classification accuracy for individual applications. Instead, they compared the accuracy between different algorithms. This comparison may not accurate when the datasets are not balanced.

3.4 Hybrid Approach

Jeffrey Erman et al. [38] proposed a method to use the statistics of flow information to classify applications. Their approach is different from others. It uses both pre-classified and unclassified training datasets in the experiment. And they use a *semi-supervised* method to train the classification engine.

They listed three advantages of their proposed approach. First, the classifier can be trained quickly and accurately with few pre-classified training datasets and many unclassified traffic sources. Second, their approach is able to recognize new applications without predefined rules, and it can adapt the behavioral changes for the existing applications. Third, this approach offers the functionality to enhance the classifier via the operator.

Their classification method consists of two steps. They first use the k-means clustering approach to group the training datasets. The datasets consist of pre-classified traffic and unclassified traffic. Then, they use the output from the clustering algorithm to map to the known traffic types.

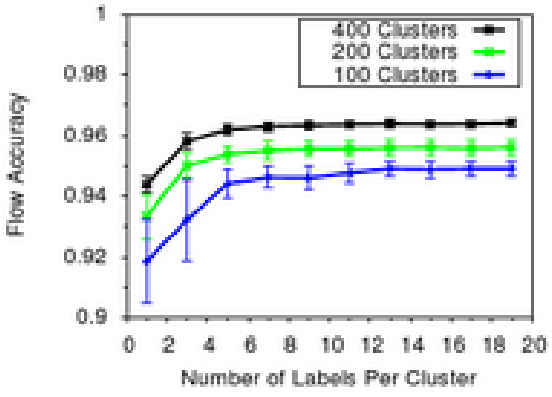

Figure 6-Selective Labeling of Flows

The testing result (Figure-5) from their research shows that the flow accuracy of their research can achieve 94% with 2 pre-classified flows per group and 400 groups. The accuracy of classification reached margin when using more than 5 pre-classified flows per group. The authors mentioned that this approach is a relatively fast training approach. And it can handle unknown traffic at any time during the classification process. Also, the classifier can be enhanced by network operator. However, we can only determine the accuracy of their approach from their research results. They did not demonstrate those advantages of their approach.

4. CONCLUSION

This survey paper is introducing the development of packet classification technics. Packet classification started using port number since more than 70 % of the traffic went through a static port number. But later, the application starts to use dynamic port number instead of static ones. The classification technic began to use packet payload inspection to classify traffic. Packet inspection approach has the highest classification accuracy, as high as 100% in the best scenario. However, the performance will decrease significantly when classifying encrypted packets. The researches started to use statistical features in packet header or flow content to create rules, fingerprint or signature to classify Internet traffic. Machine learning is an important technic for statistical based classification. The first research using machine learning in classification is in 1994, and it has been almost 20 years. During the 20 years, the search begins with classify applications into general groups of applications. Then the research starts to classify individual application or protocol using machine learning. Currently, around 70% of traffic is Peer-to-Peer traffic. Furthermore, applications like Skype, BitTorrent and GTalk encrypted their packets to prevent user information leak from deep packet inspection. Another reason to encrypt their traffic is that the Peer-to-Peer applications want to bypass the firewalls to share resources. The port based and payload based classification will not work properly with encrypted traffic. Hence many recent research papers focused on encrypted packets classification using machine learning. The benefit of using machine learning is that the machine will learn the application behaviors in the training phrase and generate classification rules to classify future flows. Machine learning process can be done without constantly updating the classification rules manually. Some early papers (e.g. [31, 32]) using machine learning to classify packet into categories of application. Now the research begins to let machine to create rules automatically to adapt the changes in application behavior [38].

Machine learning uses different Machine learning technic has two main categories, supervised and unsupervised. The main difference between supervised method and unsupervised method is that supervised method usually has two phases, training phase and classification phase whereas unsupervised method does not require a training phrase. The reason for that is unsupervised method is a method to calculate and group application with similar behaviors or patterns. They are not used to directly classify individual application. In [38] the authors use clustering to grouping the datasets to simplify their future work.

There are interest research results from the papers. The evaluation matrix starts with false positive, false negative, true positive, true negative, recall and precision. In recent research, authors start to take computational performance into account as evaluation matrix. Their approach has shown that the performance efficiency changes dramatically using different algorithms. From their testing results, they show that C4.5 algorithm perform better in term of computational performance as well as classification accuracy.

The research shows that they share some common issues. Most of the research is based on the full-flow based sample dataset. The high accuracy is based on a closed testing environment. Their accuracy may change dramatically using the traces from real Internet. Currently, this is not a standard method to compare the classification accuracy. Some of them compare the accuracy in the application level. Other people compare the classification accuracy between different algorithms. There are no standard application types or protocol types to evaluate the classification accuracy. Most of them use WWW, HTTP, and FTP as application type. But they fail to address the applications using encryption. Some research will include SSH, Skype, and GTalk in their research. But they fail to compare the classification of encrypted application with non-encrypted application.

Further work needs to be done to evaluate the performance of classifying P2P file sharing applications.

.

5. REFERENCES

[1] Snort - The de facto standard for intrusion detection/prevention, http://www.snort.org, as of August 14, 2007.

[2] Bro intrusion detection system - Bro overview, http://bro-ids.org, as of August 14, 2007.

[3] V. Paxson, "Bro*: A system for detecting network intruders in real-time*," Computer Networks, no. 31(23-24), pp. 2435–2463, 1999.

[4] Azzouna, Nadia Ben and Guillemin, Fabrice, *Analysis of ADSL Trafficon an IP Backbone Link*, IEEE IEEE Global TelecommunicationsConference 2003, San Francisco, USA, December 2003.

[5] Cho, Kenjiro, Fukuda, Kenshue, Esaki, Hiroshi and Kato, Akira, *The Impact and Implications of the Growth in Residential User-to-User Traffic*, ACM SIGCOMM 2006, Pisa, Italy, September 2006.

[6] Balachandran, Anand; Voelker, Geoffrey M.; Bahl, Paramvir and Ragan,P. Venkat, *Characterizing user behavior and network performance in a public wireless LAN*, Proceedings of the 2002 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems, pp. 195-205, 2002.

[7] Internet assigned numbers authority (IANA), http://www.iana.org/assignments/port-number (last accessed October, 2009)

[8] Karagiannis, Thomas; Broido, Andre; Brownlee, Nevil; Claffy, K.C. and Faloutsos, Michalis, *Is P2P dying or just hiding?*, IEEE Global Telecommunications Conference, November 2004.

[9] Karagiannis, Thomas; Broido, Andre; Faloutsos, Michalis and Claffy, K.C., *Transport Layer Identification of P2P Traffic*, Internet Measurement Conference (IMC '2004), October 2004.

[10] Moore, Andrew W. and Papagiannaki, Konstantina, *Toward the Accurate Identification of Network Applications*, Passive and Active Measurement Workshop (PAM 2005), March 2005.

[11] Madhukar, A. and Williamson, C., *A Longitudinal Study of P2P Traffic Classification*, 14th IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems, September, 2006.

[12] A. W. Moore, K. Papagiannaki, *Toward the accurate identification of network applications*, in: Passive and Active Network Measurement: Proceedings of the Passive &Active Measurement Workshop, 2005, pp. 41–54.

[13] A. Madhukar, C. Williamson, *A longitudinal study of p2p traffic classification*, in: MASCOTS '06: Proceedings of the 14th IEEE International Symposium on Modeling, Analysis, and Simulation, IEEE Computer Society, Washington, DC, USA, 2006, pp. 179–188. doi:http://dx.doi.org/10.1109/MASCOTS.2006.6.

[14] J. Klensin, *SIMPLE MAIL TRANSFER PROTOCOL*, IETF RFC 821, April 2001; http://www.ietf.org/rfc/rfc2821.txt

[15] T. Karagiannis, K. Papagiannaki, M. Faloutsos, *BLINC: multilevel traffic classification in the dark*, in: SIGCOMM '05: Proceedings of the 2005 conference on Applications, technologies, architectures, and protocols for computer communications, ACM Press, New York, NY, USA, 2005, pp. 229–240.

[16] L. Bernaille, R. Teixeira, I. Akodkenou, A. Soule, K. Salamatian, *Traffic classification on the fly*, SIGCOMM Comput. Commun. Rev. 36 (2) (2006) 23–26.

[17] J. Erman, M. Arlitt, A. Mahanti, *Traffic classification using clustering algorithms*, in: MineNet '06: Proceedings of the 2006 SIGCOMM workshop on Mining network data, ACM Press, New York, NY, USA, 2006, pp. 281–286.

[18] P. Haffner, S. Sen, O. Spatscheck, D. Wang, *ACAS: automated construction of application signatures*, in: MineNet '05: Proceeding of the 2005 ACM SIGCOMM workshop on Mining network data, ACM Press, New York, NY, USA, 2005, pp. 197–202.

[19] C. V. Wright, F. Monrose, G. M. Masson, *On inferring application protocol behaviors in encrypted network traffic*, J. Mach. Learn. Res. 7 (2006) 2745–2769.

[20] C. Wright, F. Monrose, G. M. Masson, *HMM profiles for network traffic classification*, in: VizSEC/DMSEC '04: Proceedings of the 2004 ACM workshop on Visualization and data mining for computer security, ACM Press, New York, NY, USA, 2004, pp. 9–15.

[21] A. D. Montigny-Leboeuf, *Flow Attributes For Use In Traffic Characterization*, CRC Technical Note No. CRC-TN-2005-003.

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

[23] C. Dewes, A. Wichmann, and A. Feldmann, "*An analysis of Internet chat systems*," in ACM/SIGCOMM Internet Measurement Conference 2003, Miami, Florida, USA, October 2003.

[24] T. Lang, G. Armitage, P. Branch, and H.-Y. Choo, "A synthetic traffic model for Half-life," in Proc. Australian Telecommunications Networks and Applications Conference 2003 ATNAC2003, Melbourne, Australia, December 2003.

[25] T. Lang, P. Branch, and G. Armitage, "A synthetic traffic model for Quake 3," in Proc. ACM SIGCHI International Conference on Advances in computer entertainment technology (ACE2004), Singapore, June 2004.

[26] Thuy T.T. Nguyen and Grenville Armitage. "*A Survey of Techniques for Internet Traffic Classification using Machine Learning*," IEEE Communications Survey & tutorials, Vol. 10, No. 4, pp. 56-76, Fourth Quarter 2008.

[27] Z. Shi, *Principles of Machine Learning*. International Academic Publishers, 1992.

[28] B. Silver, "*Netman: A learning network traffic controller*," in Proc. Third International Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems, Association for Computing Machinery, 1990.

[29] I. Witten and E. Frank, *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations (Second Edition)*. Morgan Kaufmann Publishers, 2005.

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

[31] A. McGregor, M. Hall, P. Lorier, and J. Brunskill, "*Flow clustering using machine learning techniques*," in Proc. Passive and Active Measurement Workshop (PAM2004), Antibes Juan-les-Pins, France, April 2004.

[32] J. Erman, A. Mahanti, M. Arlitt, and C. Williamson, "*Identifying and discriminating between web and peer-to-peer traffic in the network core*," in WWW '07: Proc. 16th international conference on World Wide Web. Banff, Alberta, Canada: ACM Press, May 2007, pp. 883–892

[33] T. Auld, A. W. Moore, and S. F. Gull, "*Bayesian neural networks for Internet traffic classification*," IEEE Trans. Neural Networks, no. 1, pp. 223–239, January 2007.

[34] A. Moore and D. Zuev, "*Internet traffic classification using Bayesian analysis techniques*," in ACM International Conference on Measurement and Modeling of Computer Systems (SIGMETRICS) 2005, Banff, Alberta, Canada, June 2005.

[35] T. Nguyen and G. Armitage, "*Training on multiple sub-flows to optimise the use of Machine Learning classifiers in real-world IP networks*," in Proc. IEEE 31st Conference on Local Computer Networks, Tampa, Florida, USA, November 2006.

[36] M. Crotti, M. Dusi, F. Gringoli, and L. Salgarelli, "Traffic classification through simple statistical fingerprinting," SIGCOMM Comput. Commun. Rev., vol. 37, no. 1, pp. 5–16, 2007.

[37] Haffner P., Sen S., Spatscheck O., Wang D., "ACAS: Automated Construction of Application Signatures", Proceedings of the ACM SIGCOMM, pp.197-202, 2005

[38] J. Erman, A. Mahanti, M. Arlitt, I. Cohen, and C. Williamson, “Semisupervised network traffic classification,” ACM International Conference on Measurement and Modeling of Computer Systems (SIGMETRICS) Performance Evaluation Review, vol. 35, no. 1, pp. 369–370, 2007.

[39] R. Alshammari and A. N. Zincir-Heywood, "*Machine learning based encrypted traffic classification: Identifying ssh and skype*," in Computational Intelligence for Security and Defense Applications, 2009. CISDA 2009. IEEE Symposium on, July 2009, pp. 1-8.

[40] Burges C. J. C., “A Tutorial on Support Vector Machines for Pattern Recognition”, Data Mining and Knowledge Discovery, 2(2): 1-47, 1998.

[41] Alpaydin E., “Introduction to Machine Learning”, MIT Press, ISBN: 0-262-01211-1.

[42] George H. John and Pat Langley Estimating Continuous Distributions in Bayesian Classifiers. Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence. pp. 338-345, Morgan Kaufmann, San Mateo, 1995.

[43] J R Quinlan, “C4.5: Programs for Machine Learning”,Morgan Kaufmann Publishers,isbn=1-55860-238-0, 1993

[44] Cohen W. W., “Fast effective rule induction”, Proceedings of the 12th International Conference on Machine Learning, pp. 115-123 , 1995.