Real-time Encrypted Traffic Identification using Machine Learning

Chengjie Gu, Shunyi Zhang and Yanfei Sun
Institute of Information Networks Technology, Nanjing University of Posts and Telecommunications, Nanjing, China
Email: jackiee.gu@gmail.com, {dirzsy, sunyanfei}@njupt.edu.cn

***Abstract*—Accurate network traffic identification plays important roles in many areas such as traffic engineering, QoS and intrusion detection etc. The emergence of many new encrypted applications which use dynamic port numbers and masquerading techniques causes the most challenging problem in network traffic identification field. One of the challenging issues for existing traffic identification methods is that they can't classify online encrypted traffic. To overcome the drawback of the previous identification scheme and to meet the requirements of the encrypted network activities, our work mainly focuses on how to build an online Internet traffic identification based on flow information. We propose real-time encrypted traffic identification based on flow statistical characteristics using machine learning in this paper. We evaluate the effectiveness of our proposed method through the experiments on different real traffic traces. By experiment results and analysis, this method can classify online encrypted network traffic with high accuracy and robustness.**

***Index Terms*—P2P, machine learning, encrypted traffic, traffic identification**

I. INTRODUCTION

Accurate network traffic identification would assist network administrators effectively on many network tasks such as managing bandwidth and ensuring security. The demand for bandwidth management methods that optimize network performance and provide QoS guarantees has increased substantially in recent years. Peer-to-Peer (P2P) applications have dramatically grown in popularity over the past few years, and now constitute a significant share of the total traffic in many networks. Therefore, accurate and online identification of network traffic plays important roles in many areas such as traffic engineering, QoS, and intrusion detection etc [1].

Skype is an encrypted P2P (Peer to Peer) VoIP application. Skype is widely known for its broad range of features, including free voice and video conferencing, and its ability to use P2P technology to overcome common firewall and NAT problems. Originally developed by the entrepreneurs who created the pioneering Web applications Kazaa, Skype ended 2008 with 405 million user accounts a 47% increase from 2007. According to TeleGeography Research, Skype users spent 33 billion minutes talking to people in other countries, representing 8% of all international voice traffic in 2008. Moreover, Skype usage hit an all-time peak on March 30, 2009, when more than 17 million users were online at the same time.

The simplest approach to traffic identification is port-based traffic identification which consists in examining the port numbers in TCP headers. However, this approach becomes increasingly inaccurate when Skype application use non-standard ports to by-pass firewalls or circumvent operating systems restrictions. Moreover, ports can be dynamically allocated as needed. Payload-based analysis technique to classifying network traffic is to inspect the payload of every packet. This technique can be extremely accurate when the payload is not encrypted. However, encrypted applications such as Skype imply that the payload is opaque. Substantial attention has been invested in data mining techniques and machine learning algorithms using flow features for traffic identification. Traffic identification method based on flow statistics using machine learning shows effective performance in this field [2].

In this paper we investigate how encrypted Skype traffic can be accurately and rapidly identified from observing the statistical properties of a small sequence of any part of a flow. Indeed, covering a collection of such different encrypted behavior makes it difficult to distinguish Skype from non-Skype traffic. Thus, the goal of this work is to develop a model that distinguishes Skype from non-Skype traffic without using IP addresses, port numbers or payload information. We believe that this will not only enable our model to generalize from one network to another well but also potentially will enable us to employ such an approach for the identification of other encrypted applications. In order to identify encrypted Skype traffic, four different machine learning algorithms will be employed. These are C4.5, Support Vector Machine, Naive Bayesian and Random Forest.

We take two aspects to improve the accuracy and speed of the machine learning methods for Internet traffic identification. 1) In order to achieve early detection, we allow the classifier to classify traffic flows early in the connection using the first p packets of flow. 2) We choose optimal algorithm method which can obtain high accuracy with faster computational time.

The remainder of this paper is organized as follows: Section Ⅱ reviews some related work in this area. In

Supported by National High-Tech Research and Development Plan (863 of China (No.2009AA01Z212, No.2009AA01Z202), National Natural Science Foundation of China (No. 61003237)

doi:10.4304/jsw.6.6.1009-1016

Section Ⅲ, we discuss the encrypted Skype application. Section Ⅳ proposes the real-time encrypted traffic identification methodology using machine learning, and feature selection will also be presented in this section. Section Ⅴ gives the empirical traces collection. Section Ⅵ gives the experimental results and analysis. Finally, we conclude our paper and discuss future work in Section Ⅶ.

II. RELATE WORK

The earliest and simplest approach to traffic identification is port-based traffic identification which consists in examining the port numbers in TCP headers [3]. This solution of the well-known ports classifies the traffic according to the ports registered in the IANA [4]. However, the new P2P applications use different strategies to camouflage their traffic in order to evade detection. So the port-based method is no longer reliable.

In order to deal with the disadvantages of the above method, payload-based identification method is proposed to inspect the packet payload [5]. Several payload-based analysis techniques have been proposed to inspect the packets payload searching for specific signatures [5-8], some researchers shows that identification is extremely accurate. Although this solution does can achieve high identification accuracy, it can't work with encrypted traffic or newly P2P applications. On the other hand, signature searching in the payload of every packet produces a high consume of resources [9].

Other alternatives to solve problems of payload-based traffic identification include methods based on the host-behavior that can classify the traffic according to information extracted from the interactions of the end-hosts [10].The host-behavior-based approach is developed to capture social interaction observable even with encrypted payload [11]. However, this method such as BLINC can't classify exactly the applications. It could suppose a problem with applications that are theoretically from di□erent groups but with similar behavior.

At the same time, traffic identification method based on flow statistics shows effective performance in this field. Substantial attention has been invested in data mining techniques and machine learning algorithms using flow features for traffic identification [12-16]. Machine learning technique which is a powerful tool in data separation in many disciplines aims to classify data based on either a priori knowledge or statistical information extracted from raw dataset. This method can be well suited with Internet traffic identification, as long as the traffic classified into categories that exhibit similar characteristics in parameters. Nguyen et al. [12] provided context and motivation for the application of ML techniques to IP traffic identification, and reviewed some significant works. Machine learning algorithms are generally divided into supervised learning and unsupervised learning. Supervised learning requires training data to be labeled in advance and produces a model that fits the training data. Moore et al. [13] used a Naive Bayes classifier which was a supervised machine learning approach to classifying internet traffic. But only 65% accuracy rate, which was not good enough to classify Internet traffic. Williams et al. [14] conducted a comparison of five machine learning algorithms which were widely used to classify empirical study of Internet traffic. Among these algorithms, C4.5 achieved the highest accuracy in their results. Auld [15] proposed supervised machine learning based on a Bayesian neural network to classify the traffic with higher accuracy and better stability, but it was not capable for real-time applications. Ma et al. [16] used C4.5 decision tree to classify Internet traffic. This method could identify traffic of different types of applications with high accuracy, by collecting some features at the start of the flow.

Unsupervised learning essentially clusters flows with similar characteristics together [17-21]. The advantage is that it does not require training, and new applications can be classified by examining known applications in the same cluster. McGregor et al. [17] used unsupervised EM (Expectation Maximization) algorithm to cluster flows described by features, but the method could only classify groups of traffic with similar properties. Zander et al. [18] extended this work by using an EM algorithm called Auto Class, and found the optimal feature subset for classifying traffic. Erman et al. [19] compared the performance of unsupervised machine learning algorithms in traffic identification. Since our main focus is on evaluating the predictive power of a trained traffic classifier rather than on detecting new applications or flow clustering. Also, Erman et al. [20] evaluated the performance of two clustering algorithms, namely K-Means and DBSCAN, in Internet traffic identification. The result indicated that K-Means was one of the quickest and simplest algorithms for clustering of Internet flows. Bernaille et al. [21] used a simple K-Means clustering algorithm to perform identification by using only the first five packets of the flow, aiming at applying on the real-time identification.

III. SKYPE OVERVIEW

In recent years, the popularity of VoIP-telephony has progressively grown and the majority of network operators have started offering VoIP-based phone services. Skype is widely known for its broad range of features, including free voice and video conferencing, and its ability to use P2P technology to overcome common firewall and NAT problems. Skype users can speak to other Skype users for free, call traditional telephone numbers for a fee, receive calls from traditional phones, and receive voicemail messages. It is a versatile method of synchronous and asynchronous communication.

Skype system is also an encrypted P2P VoIP network. Skype is related to KaZaA which is a famous P2P filesharing system and it consists of the ordinary nodes, super nodes and servers. The Skype P2P network organizes participants into two layers: super nodes, and ordinary nodes. Such networks have been the subject of recent research.Typically, super nodes maintain an overlay network, while ordinary nodes pick one super nodes to associate with; super nodes also function as ordinary nodes and are actually elected from ordinary

nodes. Ordinary nodes issue queries through the super nodes with which they are associated.

Skype communication consists of three components: Skype client login, buddy lookup and file/voice/video communication. Firstly, Skype client will login on to Skype and the login process could be divided into four steps: scanning super nodes, connecting with super nodes, connecting to updateservers and login on servers. Then, Skype clients need to lookup super nodes to obtain the buddy's IP address before they conduct file transfers, chat services, voice and video communications. The lookup can be classified into distribution lookup and concentration lookup. With distribution lookup, a Skype client sends requests to three super nodes which are known alive. The super node which responds to the request will return the buddy's IP address or return the super nodes which might know the buddy's IP address. Before obtaining the IP address of interest, the lookup can repeat at most 6 rounds. The maximum of number of super nodes included in the search process is 18. If the Skype client doesn't obtain the buddy's IP address using distribution lookup, it will use concentration lookup which asks the servers to find the IP address.

IV. REAL-TIME ENCRYPTED TRAFFIC IDENTIFICATION USING MACHINE LEARNING

A. Real-time Encrypted Traffic Identification Methodology

In our real-time encrypted traffic identification methodology, a flow is defined to be as a series of packet exchanges between two hosts, identifiable by the 5-tuple {source address, source port, destination address, destination port, transport protocol}, with flow termination determined by an assumed timeout or by distinct flow termination semantics. For each flow, network monitors can record statistics such as duration, bytes transferred, mean packet interarrival time, and mean packet size. Let $X = \{x_1, x_2, \cdots, x_n\}$ be a set of flows. A flow instance x_i is characterized by a vector of attribute values, $x_i = \{x_{ij} \mid 1 \le j \le m\}$, where m is the number of attributes, and x_{ij} is the value of the j^{th} attribute of the i^{th} flow. Also, let $Y = \{y_1, y_2, \cdots, y_q\}$ be the set of traffic classes, where q is the number of classes of interest. The y_i can be classes such as "HTTP", "Streaming", and "Peer-to-Peer". Therefore, our goal is to learn a mapping from a *m*-dimensional variable X to Y.

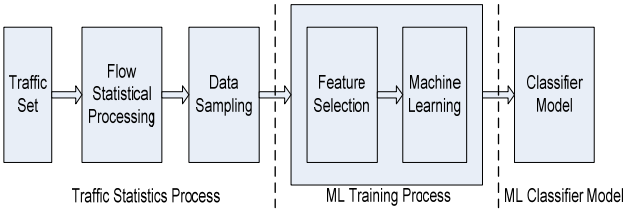

Figure 1. Training the supervised machine learning classifier

Fig.1 illustrates the sequence of events involved in training a supervised ML traffic classifier. The flow statistics processing module involves calculating the statistical properties of these flows as a prelude to generating features. An effective module is data sampling, designed to narrow down the search space for the ML algorithm when faced with extremely large training datasets. The sampling module extracts statistics from a subset of instances of various application classes, and passes these along to the classifier to be used in the training process. As noted in next sub-section, a feature selection step is desirable to limit the number of features actually used to train the supervised ML classifier and thus create the classifier.

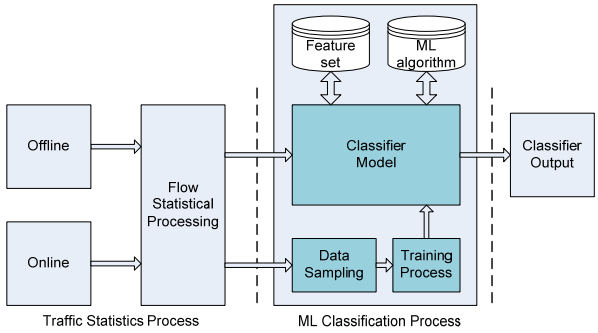

Figure 2. Real-time encrypted traffic identification using ML algorithm

As illustrated in Fig.2, the real-time encrypted traffic identification methodology includes two steps: off-line ML modeling and on-line ML identification. Firstly, according to flow information from preprocessing module, flow statistics is computed in terms of each selected feature and stored them to the corresponding database when reaching some milestone. Secondly, training and testing sets are uniformly sampled for the specific ML identification. Once the datasets is ready, the system carries out feature selection to eliminate the redundant and irrelevant features, resulting in optimal feature subset. Finally, the flow is trained using the selected ML classifier, and evaluated by performance metrics. In default, each experiment is repeated on 10 independently sampled datasets to eliminate bias. If satisfied, ML modeling is finished and ready for the traffic identification, otherwise the process is repeated. In the stage of on-line ML identification, the output model produced by ML modeling is applied to classify the captured traffic. Eventually, the identification output would be applied to different network activities.

Our proposed method should learn traffic character from identified known traffic using machine learning, which can help to identify unknown and encrypted applications intelligently and automatically. For a range of network activities, this traffic classification should meet the key criteria, such as low complexity, real time, high accuracy, early detection and robustness to provide QoS guarantees according to all kinds of Internet application levels with minimum manual intervention. At the same time, it is practicable and scalable to facilitate online real-time identification on high speed links for large traffic volumes with low overheads and low computational complexity.

B. Feature Selection

Prior to the ML modeling, feature selection can be executed off-line, regardless of its high complexity. Feature selection is an important step to machine learning which is the process of choosing a subset of original features. The process removes irrelevant and redundant features to improving algorithm performance. Many flow statistics can be calculated from a flow, but not all features provide good discrimination between the different applications. Using such features can decrease the accuracy of the classifier. We start with 23 candidate features as illustrated in Table 1.

TABLE 1 STATISTICS OF INITIAL FEATURE SET

Num	Feature description	Abbreviation
1	Duration of the flow	duration
2	Number of packets in forward direction	fpkts
3	Number of packets in backward direction	bpkts
4	Number of bytes in forward direction	fbytes
5	Number of bytes in backward direction	bbytes
6	Minimum forward packet length	minfpktl
7	Mean forward packet length	meanfpktl
8	Maximum forward packet length	maxfpktl
9	Minimum backward packet length	minbpktl
10	Mean backward packet length	meanbpktl
11	Maximum backward packet length	maxbpktl
12	Minimum forward inter-arrival time	minfiat
13	Mean forward inter-arrival time	meanfiat
14	Maximum forward inter-arrival time	maxfiat
15	Minimum backward inter-arrival time	minbiat
16	Mean backward inter-arrival time	meanbiat
17	Maximum backward inter-arrival time	maxbiat
18	Minimum of active flow	minaf
19	Mean of active flow	meanaf
20	Maximum of active flow	maxaf
21	Minimum of idle flow	minif
22	Mean of idle flow	meanif
23	Maximum of idle flow	maxif

We use the Correlation-based Feature Selection (CFS), which is computationally practical and outperforms the other filter method in terms of identification accuracy and efficiency. Correlation-based Feature Selection uses an evaluation heuristic that examines the usefulness of individual features along with the level of inter-correlation among the features. High scores are assigned to subsets containing attributes that are highly correlated with the class and have low inter-correlation with each other. We use a Best First search to generate candidate sets of features from the feature space, since it provides higher identification accuracy than Greedy search. Best First search is similar to greedy search in that it creates new subsets based on the addition or removal of features to the current subset. However, it has the ability to backtrack along the subset selection path to explore different possibilities when the current path no longer shows improvement. The 10 flow features that were chosen as illustrated in Table 2. In the rest of the paper we use this set of features as a basis for our classifiers.

TABLE 2 STATISTICS OF OPTIMAL FEATURE SUBSET

Num	Feature description	Abbreviation
1	Number of packets in forward direction	fpkts
2	Number of packets in backward direction	bpkts
3	Mean forward packet length	meanfpktl
4	Maximum forward packet length	maxfpktl
5	Mean backward packet length	meanbpktl
6	Maximum backward packet length	maxbpktl
7	Mean of active flow	meanaf
8	Maximum of active flow	maxaf
9	Mean of idle flow	meanif
10	Maximum of idle flow	maxif

C. Machine Learning Algorithm

ML is prevailing in traffic identification because of its being independent from the port and payload information. In order to identify encrypted Skype traffic, three different machine learning algorithms are deployed. These are C4.5, Support Vector Machine (SVM), Naive Bayesian and Random Forest.

C4.5 is a decision tree based identification algorithm. A decision tree is a hierarchical data structure for implementing a divide-and-conquer strategy. It is an efficient non-parametric method that can be used both for identification and regression. In non-parametric models, the input space is divided into local regions defined by a distance metric. In a decision tree, the local region is identified in a sequence of recursive splits in smaller number of steps. A decision tree is composed of internal decision nodes and terminal leaves. Each node m implements a test function $f_m(x)$ with discrete outcomes labeling the branches. This process starts at the root and is repeated until a leaf node is hit. The value of a leaf constitutes the output. In the case of a decision tree for identification, the goodness of a split is quantified by an impurity measure. A split is pure if for all branches, for all instances choosing a branch belongs to the same class after the split. One possible function to measure impurity is entropy, equation 1.

$$I_{\mathrm{m}} = -\sum_{i=1}^{K} p_m^i \log_2 p_m^i \qquad (1)$$

If the split is not pure, then the instances should be split to decrease impurity, and there are multiple possible attributes on which a split can be done. Indeed, this is locally optimal, hence has no guarantee on finding the smallest decision tree. In this case, the total impurity after the split can be measured by equation 2.

$$I'_{m} = -\sum_{j=1}^{n} \frac{N_{mj}}{N_m} \sum_{i=1}^{K} p_m^i \log_2 p_{mj}^i \qquad (2)$$

Support Vector Machines (SVMs) are a set of machine learning methods used for regression and identification problems. They belong to a family of generalized linear classifiers. A special property of this family of classifiers is to simultaneously minimize the empirical identification error and maximize the geometric margin. In this case, data is represented by a vector of n attributes or n features. The overall identification problem then takes the form of determining whether this data can be separated by n-1 dimensional hyper-plane. Assuming our data is linearly sepa- rable; we should find a hyperplane that separates our feature vectors. This is a typical form of linear classifier. There are many linear classifiers that might satisfy this property. However, we are additionally interested in establishing the maximum separation/margin between the two classes. If such a hyperplane exists, the hyperplane is clearly of interest and is known as the maximum-margin hyperplane and such a linear classifier is known as a maximum margin classifier. The feature vectors from which the distance to the hyperplane is measured, or the vectors at either side of the margin, are known as the support vectors.

Naive Bayesian is a statistical classifier based on Bayess theorem that gives its conditional probability a given class. This identification method analyses the relationship between instance of each class and each attributes to acquire a condi- tional probability for the relationships between the attribute values and the class. Naive Bayesian classifier assumes the values of the input features are independent and have no effect on a given class. This assumption, conditional independence, is made to simplify the computations and consider to be naive. Naive Bayesian can be used for identification in straightforward process by computing the probability of occurrence for each class prior probability, and computing the probability of occurrence of instance in a given class. Moreover, Naive Bayesian has managed to achieve good results even though when conditional independence assumption is violated.

Random Forest is a classifier consisting of a collection of tree-structured classifiers. Ensemble identification methods train several classifiers and combine their results through a voting process. RF is a general example for ensemble methods using tree-type classifiers. It is also the name of a specific implementation by Leo Breiman. The classifier uses large number of decision trees. To classify a new object, the object is sent to each tree in the forest. Each tree gives a identification for the object and the forest chooses the identification having the most votes. Each tree in the forest is grown as follows. First, choose N samples randomly with replacement from the original training dataset for growing the tree. Second, select m variables randomly out of total M variables independently for each node and use the best split on the selected m variables to split the node. The value of m is held constant during the forest growing. Third, each tree is grown to maximum depth without pruning. Finally, vote the trees to get predictions.

V. EMPIRICAL TRACES COLLECTION

In our experiments, the performance of the different machine learning algorithms is established on two different network data sources, Handmade_Set was simply labeled by the payload information or port characteristics through manual identification in our laboratory; Univetsity_Set was collected from Nanjing University of Posts and Telecommunications.

Unlike the usual way to obtain traces, we set a local experimental network with around 100 hosts to generate traffic manually to get Handmade_Set. Let each host run the specific application (HTTP, MAIL, FTP, DATABASE, P2P, GAME, etc.) at the same time. Since the applications run in the host is predetermined, it is easy to classify and categorize the traffic flow by the IP address. Table 3 summarizes the applications in our experiments. This set can be used as base truth to evaluate the accuracy of the classifier.

TABLE 3 STATISTICS OF HANDMADE_SET

Type of flow	Num of flow	Percent(%)
WWW	1000	12.5
MAIL	1000	12.5
FTP	1000	12.5
DATABASE	1000	12.5
SERVER	1000	12.5
SKYPE	1000	12.5
PPLIVE	1000	12.5
GAME	1000	12.5
Total	8000	100

To facilitate our work, we collected traces from the Internet link of Nanjing University of Posts and Telecommunications. Campus trace was 20 1-hour traces, was collected over a span of six months from April 10, 2009 to October 10, 2009 in all academic units and laboratories on the campus. Over 29 different applications were identified in the University_Set. To simplify the presentation, we choose the eight applications the same as Handmade_Set. Table 4 summarizes the applications found in the 20 1-hour University _Set traces.

TABLE 4 STATISTICS OF UNIVERSITY _SET

Type of flow	Num of flow	Percent(%)
WWW	4606712	72.55
MAIL	561994	8.85
FTP	611786	9.63
DATABASE	528681	8.32
SERVER	2876	0.04
SKEPY	10897	0.17
PPLIVE	13698	0.22
GAME	13453	0.22
Total	6350097	100

VI. EXPERIMENTAL RESULTS AND ANALYSIS

A. Evaluation Metrics

To measure the performance of our proposed method, we use three metrics: *accuracy*, *precision* and *recall*. In this paper, *TP*, *FP*, and *FN* are the numbers of true positives, false positives, and false negatives, respectively. True Positives is the number of correctly classified flows, False Positives is the number of flows falsely ascribed to a given application, and False Negatives is the number of flows from a given application that are falsely labeled as another application.

Precision of an algorithm is the ratio of True Positives over the sum of True Positives and False Positives or the percentage of flows that are properly attributed to a given application by this algorithm.

$$precision = \frac{TP}{TP + FP} \times 100\% \quad (3)$$

Recall is the ratio of True Positives over the sum of True Positives and False Negatives or the percentage of flows in an application class that are correctly identified.

$$recall = \frac{TP}{TP + FN} \times 100\% \quad (4)$$

Accuracy is the ratio of the sum of all True Positives to the sum of all the True Positives and False Positives for all classes. We apply this metric to measure the accuracy of a classifier on the whole trace set. The latter two metrics are to evaluate the quality of identification results for each application class.

$$accuracy = \frac{\sum_{i=1}^{n} TP_i}{\sum_{i=1}^{n} TP_i + \sum_{i=1}^{n} FP_i} \times 100\% \quad (5)$$

B. Impact of Machine Learning Algorithm on Identification Accuracy

In this subsection, we investigate the accuracy of the classifier generated by C4.5, Support Vector Machine (SVM), Naive Bayesian and Random Forest learning algorithms for distinguishing Skype traffic from non-Skype traffic in a given traffic trace. To do so, we employ traffic traces captured on our Handmade_Set and University_Set. We evaluated the aforementioned learning algorithms using traffic flow 23 candidate features. To this end, we have used Weka which is an open source tool for data mining tasks. We employed Weka with its default parameters to run all algorithms on our data sets. In this case TP will reflect the number of encrypted Skype flows correctly classified whereas FP will reflect the number of non-Skype flows incorrectly classified. Naturally, a high TP rate and a low FP would be the desired outcomes.

Table 5 shows the C4.5 based classification approach is much better than other machine learning algorithms employed in identifying the encrypted Skype flow on the two datasets. C4.5 achieves 93.1% TP and 0.9% FP on HANDMADE_SET traces, 94.5% TP and 1.8% FP on the UNIVERSITY_SET traces. In our implementation, we chose C4.5 algorithm as a basis for our classifiers.

TABLE 5 IMPACT OF MACHINE LEARNING ALGORITHM ON IDENTIFICATION ACCURACY

	C4.5		SVM		NB		RF	
	TP	FP	TP	FP	TP	FP	TP	FP
	HANDMADE_SET							
Skype	**0.931**	**0.009**	0.902	0.013	0.882	0.019	0.913	0.016
Non-Skype	0.947	0.013	0.926	0.022	0.903	0.035	0.922	0.087
	UNIVERSITY_SET							
Skype	**0.945**	**0.018**	0.918	0.034	0.093	0.047	0.932	0.017
Non-Skype	0.962	0.016	0.932	0.071	0.0911	0.093	0.954	0.087

C. Impact of Feature Selection on Identification Accuracy

Feature selection can optimize for higher learning accuracy with lower computational complexity by removing irrelevant and redundant features. There appears to be a very good trade-off between feature space reduction and loss of accuracy.

We examine the impact of feature selection with C4.5 algorithm, in terms of precision using University_Set. Cross-validation testing is performed using the full feature set and the CFS subset. We obtain the precision rates across the all classes after test. Fig.3 compares the identification accuracy when using the CFS subset and the full feature set. Therefore, we can choose the optimal feature subset to classify the network traffic instead of fullset. In the rest of the paper we use this optimal subset of features as a basis for our classifiers. It can provide a dramatic decrease in the number of features required, with the best subset providing similar mean accuracy.

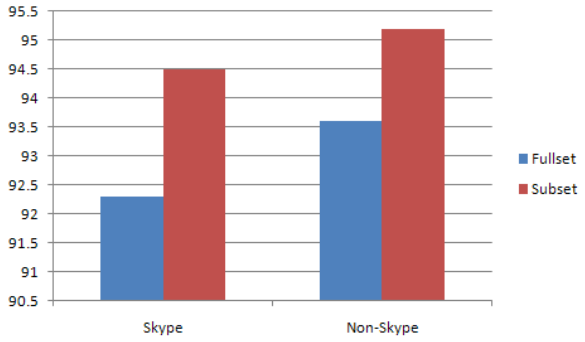

Figure 3 Impact of feature selection on identification accuracy

D. Impact of Number of Packets for Statistics on Identification Accuracy

Unlike offline identification where all discriminating flow statistics are available a priori, in the real-time identification we only have partial information on the flow statistics. In order to classify the network applications associated with a flow as early as possible, we follow the idea presented in Bernaille et al. [21] and conduct experiments to determine the appropriate packet

number *p*. The statistics information of several packets in each flow could distinguish network traffic from Internet traffic accurately with the least *p*.

For our experiments, we classified network traffic using flows from Handmade_Set and Univetsity_Set respectively. In our implementation, we choose C4.5 algorithm using the optimal feature subset to test the suitable parameter of number of packets.

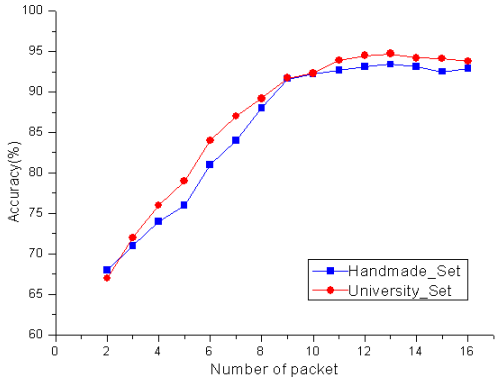

Figure 4 Impact of number of packet for statistics on identification accuracy

The experimental result shown in Fig.4 indicates that C4.5 algorithm could achieve better accuracy when we choose 12 packets for statistics. It is noticeable that the statistics of the first 12 packets could classify traffic with over 93% high accuracy in different traces. At the same time, the identification accuracy improves marginally using more than 12 packets. Considering our goal to detect network traffic as fast as possible with high accuracy, we choose 12 packets for statistics.

E. Identifying encrypted Skype from P2P traffic

Skype is an encrypted P2P VoIP application which has the samiliar characteristic with the Xunlei, PPlive, PPStream, BitTorrent, Kougou, eDonkey. The purpose of this subsection is to verify whether the proposed method is robust enough to classify Skype traffic from P2P traffic. We choose the P2P applications from 29 different applications identified in the University_Set to test our identification method. The main applications in our experiments include Xunlei, PPlive, PPStream, BitTorrent, Kougou, eDonkey and Unkonwn P2P. Table 6 shows statistics of P2P flow fromUniversity_Set.

TABLE 6 STATISTICS OF P2P FLOW FROM UNIVERSITY_SET

Application	Num of flow	Percent(%)
Skype	10897	0.83
Xunlei	447263	33.86
PPlive	288086	21.81
PPStream	109884	8.32
BitTorrent	186794	14.14
Kougou	99636	7.54
eDonkey	67398	5.11
Unkonwn P2P	110835	8.39
Total	1320793	100

With the method proposed in this paper, the results in Fig.5 show that Skype precison and recall in University_Set are 94.51% and 83.63% respectively, which indicates that Skype can be effectively identified. Specifically, the experimental results show that precision and recall is 94.81% and 83.26% for Kougou application respectively. At the same time, the average identification accuracy of unknown P2P traffic is 86.28%, which indicates that the methods can classify encrypted P2P traffic with considerable accuracy. Thus the methods are robust enough to classify Skype traffic based on the fact that the P2P applications share the similar characteristics of peer behavior, connection feature and transportation statistics.

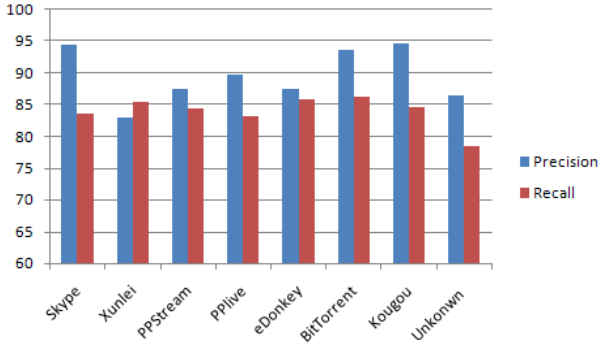

Figure 5 Identification accuracy of P2P traffic

VII. CONCLUSION

Accurate traffic identification helps to identify the application utilizing network resources, and facilitate the QoS guarantee for different applications. However, as many newly-emerged P2P applications use dynamic port numbers and encrypted techniques, it causes the most challenging problem in network traffic identification. Our work mainly focuses on how to build a real-time Internet traffic identification system based on flow statistics. We propose a real-time encryped traffic identification using machine learing. The proposed real-time encrypted traffic identification methodology can identify encrypted Skype applications using flow character-based approach with high accuracy and low overheads. Furthermore, we need more experiments to find out which features are suitable for improving the encrypted traffic identification accuracy.

ACKNOWLEDGMENT

This research is funded by National High-Tech Research and Development Plan (863) of China (No.2009AA01Z212, No.2009AA01Z202), National Natural Science Foundation of China (No. 61003237), Natural Science Foundation of Jiangsu Province (No.BK2007603), High-Tech Research Plan of Jiangsu Province (No.BG2007045). We would like to give great thanks to the reviewers for their helpful comments and feedbacks considering of manuscript improvement.

REFERENCES

[1] Soysal, Murat,Schmidt, Ece Guran. "Machine learning algorithms for accurate flow-based network traffic classification: Evaluation and comparison". *Performance Evaluation*, 2010, 67(6):451-467

[2] B.Marco, Mellia, Antonio Pescape and LucaSalgarelli. "Traffic classification and its applications to modern networks". *Computer Networks*, 2009,53(6):759-76.

[3] Karagiannis T, Roido A, Aloutsos M, Laffy K. "Transport layer identification of P2P traffic". *In Proceedings of the 2004 ACM SIGCOMM Internet Measurement Conference*, ACM: New York, 2004:121–134.

[4] Internet Assigned Numbers Authority (IANA). http://www.iana.org/assignments/port-numbers, August 28, 2010.

[5] Haffner P, Sen S, Spatscheck O, Wang D. "ACAS: Automated Construction of Application Signatures". *In SIGCOMM'05 Workshops*, Philadelphia, PA, 2005: 197–202.

[6] Moore AW, Papagiannaki K. "Toward the accurate identification of network applications". I*n Passive and Active Measurement Workshop*, Boston, MA, 2005: 41–54.

[7] T. Karagiannis, A. Broido, N. Brownlee, K. Claffy, and M. Faloutsos. "Is P2P dying or just hiding". *In IEEE Global Telecommunications Conference*, 2004:1532-1538,

[8] S. Sen, O. Spatscheck, and D. Wang. "Accurate, scalable in-network identification of p2p traffic using application signatures". *In Proceedings of the 13th international conference on World Wide Web,* ACM New York, NY, USA, 2004: 512-521

[9] Constantinou F, Mavrommantis P. "Identifying known and unknown peer-to-peer traffic". *In IEE NCA'06 Conference*, 2006: 93–102.

[10] T. Karagiannis, K. Papagiannaki, and M. Faloutsos. "BLINC: multilevel traffic classification in the dark". *In Proceedings on Applications, technologies, architectures, and protocols for computer communications*, ACM New York, NY, USA, 2005: 229-240.

[11] T. Karagiannis, K. Papagiannaki, N. Taft, and M. Faloutsos. "Profiling the end Host". *Lecture Notes Computer Science*, 2007

[12] T. Nguyen and G. Armitage. "A Survey of Techniques for Internet Traffic Classification using Machine Learning". *IEEE Communications Surveys and Tutorials*, 2008,11(3):37-52

[13] A. W. Moore and D. Zuev. "Internet traffic classification using bayesian analysis techniques". *In Proceedings of ACM SIGMETRICS international conference on Measurement and modeling of computer systems*, 2005:50-60

[14] N. Williams, S. Zander, and G. Armitage. "A preliminary performance comparison of five machine learning algorithms for practical ip traffic flow classification". *ACM SIGCOMM Computer Communication Review*,2006, 30(5):5-16

[15] T. Auld, A. W. Moore, and S. F. Gull. "Bayesian neural networks for internet traffic classification". *IEEE Transaction on Neural Network*, 2007,18(1):223-239

[16] Yongli Ma, Zongjue Qian, Guochu Shou, Yihong, Hu. "Study of information network traffic identification based on C4.5 algorithm". *2008 International Conference on Wireless Communications, Networking and Mobile Computing*, 2008

[17] A. Mcgregor, P. Lorier M. Hall, and J. Brunskill. "Flow clustering using machine learning techniques". *In Passive and Active Network Measurement*, 2004: 205-214,.

[18] S. Zander, T. Nguyen, and G. Armitage. "Automated traffic classification and application identification using machine learning". *In Proceedings of the IEEE Conference on Local Computer Networks*, 2005:250-257

[19] J. Erman, A. Mahanti, M. Arlitt, I. Cohen, and C. Williamson. "Offline/real-time traffic classification using semi-supervised learning". Technical report, University of Calgary,2007.

[20] J. Erman, M. Arlitt, and A. Mahanti. "Traffic classification using clustering algorithms". *In Proceedings of SIGCOMM workshop on Mining Network Data*, 2006: 281-286,

[21] Bernaille L, Teuxeira R, Akodkenous I, Soule A, Slamatian K. "Traffic classification on the fly". *ACM SIGCOMM Computer Communication Review* 2006; 36: 23-26.

Chengjie Gu was born in Anhui, China, in 1985. He received his M.S. degrees from Beijing Jiaotong University in China in 2009. He now is a PhD candidate of computer networks in Nanjing University of Posts and Telecommunications. His current research interests include peer-to-peer networks, network traffic classification and cognitive network.

Shunyi Zhang was born in Jiangsu, China, in 1944. He is Professor of Nanjing University of Posts and Telecommunications. He also is a member of the director board for the branch of Chinese Institute of Electronics, and Chinese Institute of Telecommunications. His current interest is in computer communication, IP technology, and next generation network.

Yanfei Sun was born in Shangdong, China, in 1976. He is a PhD and associate Professor of Posts and Telecommunications. His current interest is in computer communication, IP technology, and cognitive network.