TELKOMNIKA Indonesian Journal of Electrical Engineering
Vol. 12, No. 2, February 2014, pp. 1551 ∼ 1560
DOI: http://dx.doi.org/10.11591/telkomnika.v12.i2.3877

Hierarchical Real-time Network Traffic Classification Based on ECOC

Yaou Zhao[1], Xiao Xie[2], and Mingyan Jiang*[3]
[1,3]School of Information Science of Engineering, Shandong University, Jinan, China
[2]School of Information Science of Engineering, University of Jinan, Jinan, China
Corresponding author, e-mail: yaou_zhao@sina.com[1], jiangmingyan@sdu.edu.cn[3]

Abstract

Classification of network traffic is basic and essential for many network researches and managements. With the rapid development of peer-to-peer (P2P) application using dynamic port disguising techniques and encryption to avoid detection, port-based and simple payload-based network traffic classification methods were diminished. An alternative method based on statistics and machine learning had attracted researchers' attention in recent years. However, most of the proposed algorithms were off-line and usually used a single classifier. In this paper, a new hierarchical real-time model was proposed which comprised of a three tuple (source ip, destination ip and destination port) look up table(TT-LUT) part and layered milestone part. TT-LUT was used to quickly classify short flows which need not to pass the layered milestone part, and milestones in layered milestone part could classify the other flows in real-time with the real-time feature selection and statistics. Every milestone was a ECOC(Error-Correcting Output Codes) based model which was used to improve classification performance. Experiments showed that the proposed model can improve the efficiency of real-time to 80%, and the multi-class classification accuracy to 91.4% on the data sets which had been captured from the backbone router in our campus through a week.

Keywords: *Hierarchical Real-time Model, Network Traffic Classification, ECOC*

1. Introduction

In the recent years, with the rapid development of the Internet, a variety of new applications have sprung up (e.g. streaming, gaming, peer-to-peer (P2P)). Going by recent measurement studies, P2P now accounts for 50%-70% of the Internet traffic [1, 2]. Network bandwidth is faced with great challenge from new applications which have affected people's normal work and normal activities online especially in the peak surfing time such as 8:00 pm or 10:00 am. Network security and network diagnostic monitoring are also becoming focus. To provide a safe and efficient network environment, accurate categorization and identification of network traffic are significant and beneficial for network operators.

In the past years, the method of identifying network traffic through port number was a norm. With the evolution of new applications especially P2P, port-based identification of network traffic had been no longer accurate, because many applications did not communicate with standardized ports, and some used temporal dynamic ports or overlaid well-known ports such as Web, FTP to avoid the detection by firewalls or some other network management tools. Now many researches showed that port-based identification of network traffic performed poor. e.g. less than 70% accuracy [3, 4].

Relying on the inspection of packet contents, payload-based techniques were proposed to address the aforementioned drawbacks of port-based classification. Packets of many network applications contained characteristic signatures which could be used to classify the network flow. However, several limitations still existed in payload-based technique. First, these techniques only identified traffic for which signatures were available and were unable to classify any other traffic. Second, payload-based or deep packet inspection techniques needed large capacity and strong processing ability. Third, if the applications used encryption, payload-based techniques would fail and out of use. Fourth, the user's privacy would be affected if packet content had been inspected.

Received July 11, 2013; Revised September 6, 2013; Accepted October 3, 2013

Transport layer statistic techniques were motivated by the limitations and disadvantages of port-based and payload-based approaches [5]. In the last few years, machine learning based approaches for Internet traffic classification had been concerned by researchers. Generally, they could be divided into three categories. First, unsupervised approaches in which Cluster [6, 7] and EM algorithms [8] were common. The advantage of these methods were that it did not require labeled data set, and new applications could be classified by examining known applications in the same cluster. The second was semi-supervised approaches which allowed classifiers to be designed from training data which consisted of only a few labeled and many unlabeled flows. Jeffrey Erman firstly used semi-supervised leaning techniques for the traffic classification [9]. The Third approach was supervised methods such as Decision Tree [10], Neural Network [11, 12].

However, the currently published machine learning approaches for network traffic classification generally employed one classifier. It was difficult for one classifier to classify multi-categories at one time, so integrated method would necessary to improve one classifier's low generalization ability.

Moreover, It's important to monitor and classify network traffic in real-time, so the online model is needed and helpful to managers. Erman also proposed "milestone packet" idea to solve network traffic classification, and Jingjing Zhao [13] proposed real-time feature selection. They achieved some results, but the performance were not satisfied.

In our paper, a new hierarchical real-time model based on three tuple (source ip, destination ip and destination port) look up table (TT-LUT) and layered milestone ECOC model was proposed for improving the performance. Our main contributions were as follows:

1. Classifying flows through three tuple (source ip address, destination ip address and destination port) was proposed in our paper. As we found that many five tuple (source ip address, destination ip address, source port, destination port and protocol) flows in the same application type always hadthe same three tuple.

2. A real-time feature subset which can be calculated before the whole flow was finished was basic and crucial to hierarchical real-time study model. In this paper, nine features which could be calculated in real-time were used.

3. The whole machine learning classification procedure was layered into many milestones according to the number of packets had received in a flow. In every milestone, there was a classifier trained by feature subset updated in real-time. As a result, network traffic classification could be realized in real-time in layered milestone part.

4. ECOC(Error-Correcting Output Codes) based model was proposed as the multi-classification algorithm. Every milestone was a ECOC-based model which could increase the average classification accuracy of the milestone, and also the overall accuracy of the whole model. Experiments showed that this hierarchial real-time model work well.

The rest of this paper is organized as follows. Section 2 describes the design of hierarchical real-time network traffic classification based on ECOC. Section 3 introduces the experiments. Section 4 concludes this paper and outlines the future work.

2. Design of Hierarchical Real-time Network Traffic Classification Based on ECOC

2.1. Real-time Feature Selection

Feature selection is actually most important in machine learning. Good feature selection method cannot only improve the accuracy of algorithms, but also improve the computational performance. However most of the features selected cannot be used in online traffic classification since some features can not be calculated before the flow that has been finished. Therefore the selection of real-time feature subset is crucial to realize the online traffic classification which is also the key part of online traffic classification.

In the hierarchical real-time classification model, features used in this paper have two characteristics:

Table 1. Feature selection table

Num	Feature
1	Total volume in bytes
2	Number of upload packets
3	Number of download packets
4	Volume of upload packets
5	Volume of download packets
6	Flow time duration
7	Mean of upload packets' volume
8	Mean of download packets' volume
9	Volume of upload packets' header

1. Real-time features can be calculated on the fly and only partial information of the flow is statistic. For example, the maximum packet inter-arrival time must be obtained when the flow has finished, while the total volume of the flow can be calculated on the fly.

2. Features that used in real-time are not selected by algorithms, instead they are selected through the rule that whether they can be calculated in real-time. Statistic information is then used in the following machine learning algorithms immediately.

Table 1 shows the statistical features of flow which are taken into account in this paper as they can be calculated in real-time and are most often used in other experiments [9]. Previous work done by Zander had estimated the influence of the different attributes on the outcome of the learning [14].

2.2. ECOC-based Classifier Model

A ECOC represents a general framework based on a coding and decoding (ensemble strategy) technique to handle multi-class problems. One of the most well-known properties of the ECOC is that it improves the generalization performance of the base classifiers [15]. Moreover, the ECOC technique has demonstrated to be able to decrease the error caused by the bias and the variance of the base learning algorithm [16, 17].

In this paper, supervised artificial neural network (ANN) classifier which is trained by a set of real-time features (see Table 1) is taken as the sub-classifier of the NN ensemble (NNE). ANN can be integrated with many methods. The most common method is multi-voting. ECOC is also a form of voting with multiple classifiers. The error correcting codes can correct errors between the outputs and add redundancy to the outputs, so it can improve the generalization ability of NNE. Therefor ECOC-based classifier is an integration of many NN with ECOC.

Following the method proposed in [13], for 5 classes network traffic classification problem, the length of ECOC code is 7 and the minimum Hamming distance is 4. The proposed ECOC-based method includes two parts: training part and testing part (see Figure 1). The details of this method are shown as follows:

- Training Part

1. Create an $N_c \times n$ ECOC matrix *M*.

2. Each class is assigned to one row of *M*.

3. Each column of *M* is assigned to a binary learner. The binary classifier is trained by NN and optimized by particle swarm optimization (PSO) algorithm. The input of the binary classifier are the nine real-time features.

- Testing Part

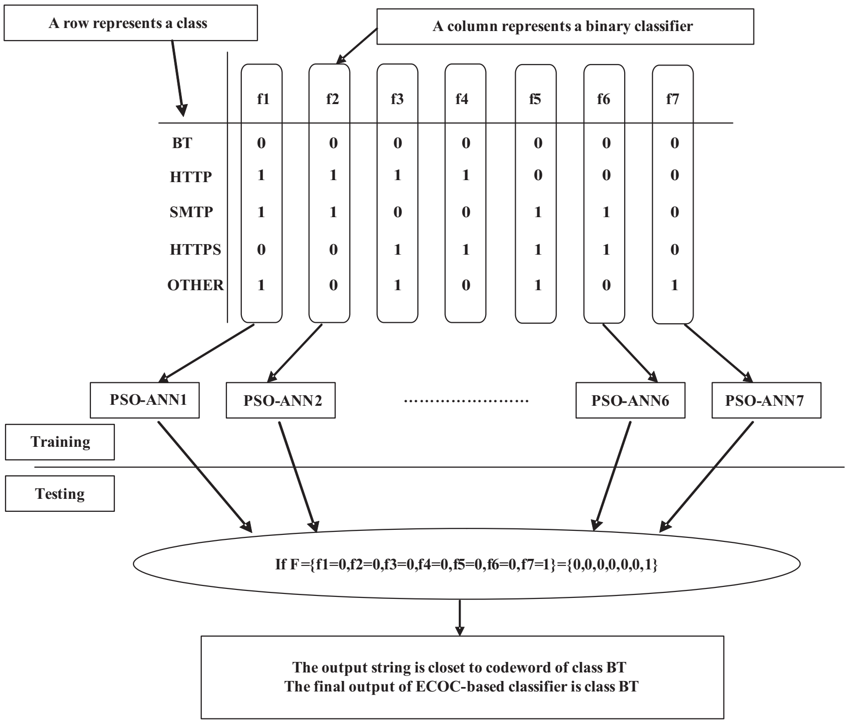


Figure 1. Training and testing phases of the ECOC-based odel

1. Apply each of the *n* classifiers to the test sample.
2. Combine the *n* outputs to form a binary string.
3. Compare the output binary string with the codeword and classify it to the class with the nearest codeword.

2.3. The importance of Three Tuple

Network traffic classification always uses five tuple to distinguish a new flow. Now most of the classification researches are based on five tuple and are TCP flows. In our paper, just TCP flows are the focus of our research. First, TCP trace accounts for a significant fraction of the overall trace. Second, start and end of TCP trace can be identified through three handshake and finish hand signal, it's convenient for us to statistic the features.

In Jeffery Erman [9], their results show that the presence of mice and elephant flows in the Internet complicates the design of classification model. This characteristic is often referred to as the elephant and mice phenomenon (a.k.a. the vital few and trivial many rule), most flows (mice flows) only have a small number of packets, while a very few flows (elephant flows) have a large number of packets. A noticeable attribute of elephant flows is that they contribute a large portion of the total traffic volume despite being relatively few in the number of flows. According to our observation, many flows are mice flows which have the same three tuple, but the volume of these flows is small proportion of all packets. Mice flows account most of all flows. If we visit a web page, there are a number of http connections with the same three tuple. Three tuple plays an important role in network traffic classification. Operating system randomly assigns source port to the application software or the connection. We find that for the same three tuple, source port number always increases. From statistics of Auckland University 's Data Set and packets we captured on our campus' backbone router, we find that many flows for BT, HTTP, SMTP, HTTPS, and OTHER in five tuple have the same three tuple (shown as table 2). Therefore, we assume that if flows have the same three tuple, they will have the same application type.

Table 2. The number of three tuple and five tuple on our campus dataset and Auckland dataset

	Campus		Auckland	
	Three Tuple	Five Tuple	Three Tuple	Five Tuple
BT	485	1902	-	-
HTTP	524	5141	629	11015
SMTP	376	756	373	748 7
HTTPS	199	1967	192	1936
OTHER	78	709	90	996

2.4. Hierarchical Real-time Classification Model

Hierarchical real-time classification model (as described in figure 2) is composed of two parts which are three tuple look up table (TT-LUT) part (part one) and layered milestone part (part two). The two parts are sequentially connected. Layered milestone part is composed of many milestones. Every milestone is a ECOC-based classifier. A packet milestone is reached when the count of the flow's total number of packets reaches a specific value. In this paper, five milestones which are 8 packets, 16 packets, 32 packets, 64 packets, and 128 packets are configured as figure 2 shows. b8 represents that flows have finished before milestone 8, or the count of the flow's total number of packets is less than or equal to 8. So as b16, b32, b64 and b128. a128 represents that flows have finished after milestone 128, or the count of the flow's total number of packets is more than 128. All flows are classified through part one and then part two. This model has dramatically increased network traffic classification's real-time characteristic.

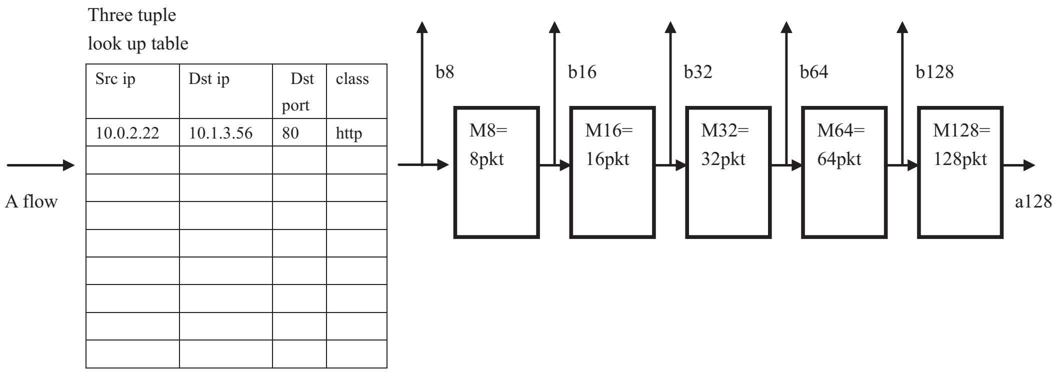


Figure 2. Hierarchical real-time classification model

As shown above, many five tuple with the same application type always have the same three tuple. We utilize this characteristic of flows to realize real-time network traffic classification. See figure 2, if the three tuple of a new flow can be found in TT-LUT, this flow will get a TT-LUT classification result that TT-LUT provides. And then the flow also need to pass through the layered milestone part to get a layered milestone classification result. Milestone 8 is the first classifier in part two, b8 flow has finished before passing it and can not be classified in real-time in layered milestone part. In this case, the final classification result of b8 flow is TT-LUT result and the real-time efficiency is not affected. For b16, b32, b64, b128 and a128 flows, their classification results will be the nearest milestone's result that passed. For example, the final class of b16 flow is calculated by milestone 8, a128 is calculated by milestone 128. Although for b16, b32, b64, b128 and a128 flows, there have been a TT-LUT classification result, we still use the result of the layered milestone as the final result, because in this way a high classification accuracy will be got. After getting the flow's final classification result, we need to update the classification result of TT-LUT with the result of the layered milestone if they are different from each other.

If the three tuple of a new flow can not be found in TT-LUT, it can't get the classification result of TT-LUT. The flow will then go through the layered milestone part to get a layered milestone classification result. In this case, b8 flow can not be classified in real-time, because it has finished before reaching the milestone 8. For real-time efficiency calculation, b8 flow is looked as a non real-time flow. For classification accuracy calculation, b8 flow's final classification result is m8 result. For b16, b32, b64, b128 and a128 flows, their real-time characteristics will not be affected because they have passed at least one milestone classifier, and their final classification results will be the one that the nearest milestone they have passed provides. After getting the flow's final classification result, we need to create a new item with the three tuple and the flow's final classification result in TT-LUT. The hierarchical real-time model allows us to classify flows in real-time. We use the same feature set for all milestones.

3. Results and Discussion

3.1. Data Set

We used the payload-based classifier developed in earlier works to establish ground truth for our traces. Several network traffic traces were used for training and evaluating our classification model. TCP traces were captured at the backbone router of our campus network (see Figure 3). We captured 5 kinds of traffic traces for our experiments which were BT, HTTP, HTTPS, SMTP and OTHER applications through a week. The training data set is composed of 2500 samples where the number of every application is 500. The test data set is composed of 500 samples where the number of each application is 100. All the samples are randomly selected from all the flows. For hierarchical real-time classification model, training and testing samples must be accounted through milestone as shown in table 3, because some of the flows have finished before they reach the milestones. In all the samples there are many flows have finished before m8 and there are many flows with the same three tuple. For example, in training samples, there are 500 BT samples before m8, and for m16 only 135 BT samples. In the 500 BT samples there are only 242 different three tuple.

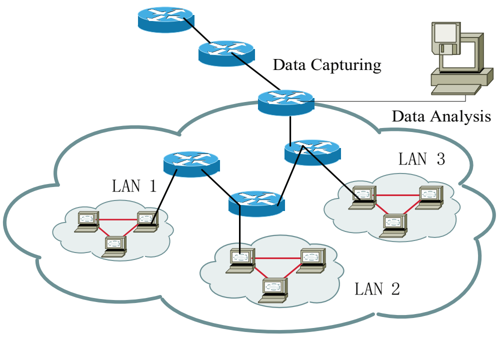


Figure 3. The topology of network traffic captured environment

Table 3. Training and testing data

	Training Data					Testing Data				
	m8	m16	m32	m64	m128	m8	m16	m32	m64	m128
BT	500	135	114	38	32	100	21	18	12	10
HTTP	500	142	98	55	20	100	21	16	9	2
SMTP	500	469	369	112	63	100	94	71	29	15
HTTPS	500	262	146	99	57	100	43	27	19	9
OTHER	500	69	51	25	10	100	3	3	3	1
SUM	2500	1077	778	329	182	500	182	135	72	37

3.2. Algorithm Effectiveness

The overall effectiveness of hierarchical algorithm is calculated by overall accuracy and single accuracy.

The number of correctly classified connections in a class is refereed to as the True Positives (TP). The overall accuracy and single accuracy are thus defined as follows:

$$overall\ accuracy = \frac{\sum TP\ for\ one\ class}{total\ number\ of\ flows} \tag{1}$$

$$single\ accuracy = \frac{TP\ for\ one\ class}{total\ number\ of\ flows\ in\ this\ class} \tag{2}$$

Number of flows that can be calculated in real-time is refereed to as RN, so the real-time efficiency is calculated by real-time efficiency which is described as follows:

$$realtime\ efficiency = \frac{RN}{total\ number\ of\ flows} \tag{3}$$

Several experiments have been done and the results are shown as follows. In the following section, we will discuss our experiment in two important factors: accuracy and real-time efficiency.

3.3. ECOC-based classifier Experiments

As shown in table 4, every milestone classifier is an ensemble of seven binary classifiers through ECOC, and they are trained by samples in the according milestone, like m16 will be trained and tested by 1077 and 182 samples respectively as listed in table 3. The experiments made about how ECOC affects the classifier's performance show that the accuracy of each sub-classifier and the length of ECOC code are important. So in our experiments for ECOC-based classifier, best trained binary classifiers are selected to integrate, and the code length is seven. For example, m8 classifier ensemble is composed of seven binary classifiers whose average classification accuracy is more than 94%, and then the overall accuracy is 91.6% as listed in table 4. However, the overall accuracy for m16 and m32 are just 84.6% and 85.1% because in these two milestones the number of m16 and m32 samples decreases quickly, and the classifiers cannot be trained completely. Figure 4a shows the single accuracy of all type of flows in ECOC-based classifier in a milestone. From experiment results we have done, we know that ECOC-based method can increase the overall accuracy of the classifier from average 70% to 90% for multi-class classification, and the average single accuracy of all application types is about 90% [2]. Layered milestone part is composed of five milestone classifiers, m8, m16, m32, m64 and m128. All of them are implemented through ECOC-based classifier ensembles and have a high overall accuracy and single accuracy than other normal machine learning algorithms.

Table 4. The overall accuracy of the ECOC-based classifier in a milestone

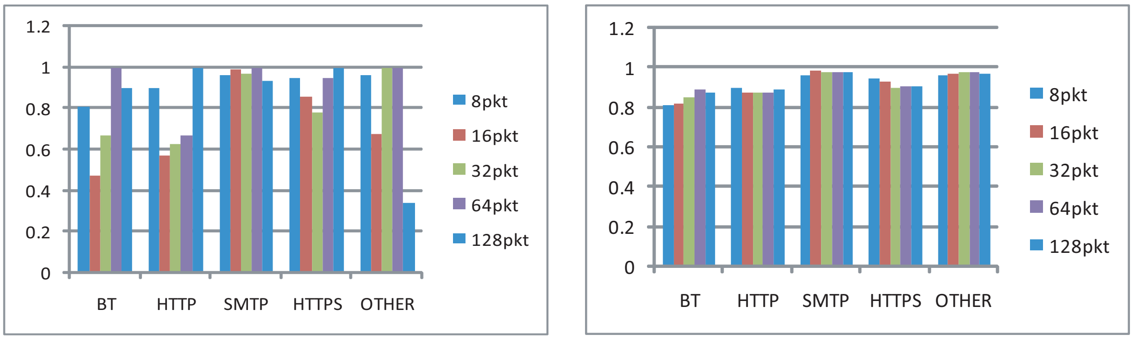
	m8	m16	m32	m64	m128
f1	0.952	0.862	0.948	0.944	0.972
f2	0.952	0.862	0.948	0.944	0.972
f3	0.922	0.912	0.925	0.986	0.972
f4	0.934	0.906	0.925	0.986	0.945
f5	0.944	0.961	0.844	0.931	0.972
f6	0.951	0.950	0.881	0.958	0.945
f7	0.982	0.978	0.992	0.944	0.972
overall accuracy	0.916	0.846	0.851	0.944	0.944

3.4. Layered Milestone Part Experiments

As layered milestone part plays an important role in hierarchical real-time network traffic classification, we'll analyze this part without TT-LUT in details. We chose the best five milestone classifiers m8, m16, m32, m64 and m128 and then combined them to analyze the overall accuracy and single accuracy of the whole layered milestone part. As shown in table 5, the overall accuracy of the whole layered milestone part are listed according to which milestone the flow has passed. the overall accuracy is increasing with the number of milestones it has passed. Of course, the single accuracy of all five application types are also increasing in every milestone as show in Figure 4b. Since the more milestones the flow has passed, the more feature information will be obtained. So in the layered milestone part, the classification result of the following milestone can correct the errors that in the forward milestone, and the overall accuracy of layered milestone part has increased to 92.6%.

Table 5. The overall accuracy of the layered milestone part

milestone has passed	overall accuracy
m8	0.916
m16	0.918
m32	0.918
m64	0.928
m128	0.926



(a) (b)

Figure 4. (a)The single accuracy of all types of flows in the ECOC-based classifier in a milestone. (b)The single accuracy of all types of flows in the layered milestone part.

3.5. Hierarchical TT-LUT & Layered Milestone Experiments

There are two parts in our hierarchical real-time model for network traffic classification which are TT-LUT part and the layered milestone part. For the proposed model, the first important thing is to create the TT-LUT table. Record is added to a TT-LUT table when there is a new three tuple. And the flow's milestone result will be padded into TT-LUT table with the three tuple's class. If the three tuple's TT-LUT result is different from the milestone's result, the TT-LUT's result will be modified to the milestone's result.

In our hierarchical real-time model TT-LUT mainly affect the b8 flow's result, as the flow has finished before m8. If there is a same three tuple in TT-LUT, the b8 flow's class will be gotten immediately. Otherwise, we suppose the b8 flow's result is m8. For b16, b32, b64, b128 and a128 flows, their final classification results are milestone results rather than the TT-LUT results.

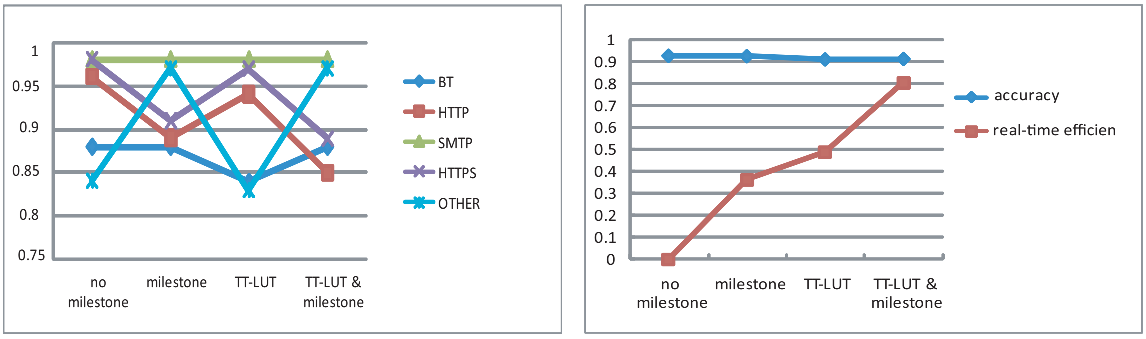
In this paper, four experiments have been done which are traditional off-line classification method. They are the methods without layered milestones or TT-LUT (no milestone), layered milestone classification method without TT-LUT as described in the above (milestone), TT-LUT based methods without layered milestones and the hierarchical real-time model with TT-LUT and layered milestone (TT-LUT & milestone). No milestone method has been discussed in our past paper [2] and the Milestone method we have discussed is on the above. TT-LUT method is in fact based on no milestone method whose idea is similar to TT-LUT & milestone. Difference between them is that the result in TT-LUT & milestone will be corrected by the result of the following milestone.

Figure 5b describes the overall accuracy of the four methods, and figure 5a describes the single accuracy. From the figure, we find that no milestone method has the highest overall accuracy about 92.8%, and the TT-LUT method has the lowest overall accuracy about 91.2%, the proposed hierarchical real-time method has a overall accuracy about 91.4%, and milestone method is about 92.6%. We can see that our proposed method has a similar classification accuracy with other methods.

3.6. Real-time performance

Figure 5b also describes the real-time efficiency of the four methods. As figure 5b shows, the real-time efficiency of no milestone method is 0%, as it's a total off-line method. the real-time efficiency of milestone method is 36.4% since b8 flows can't be classified in real-time and there are many b8 flows in samples. So the real-time efficiency is low. The TT-LUT method which is firstly work off-line. Because TT-LUT has not been created, with the process of classification, the TT-LUT table is updated in real-time. In this case, the real-time efficiency of TT-LUT is mainly affected by how many three tuple in the TT-LUT. In this experiment, the real-time efficiency is about 48.8%. The real-time efficiency of the hierarchical real-time model is the highest among all the methods, about 80.4%. It's because not only the TT-LUT part can classify part of some b8 flows whose three tuple are in TT-LUT table in real-time, but also the layered milestone part can classify flows that not in TT-LUT table in real-time.

From all experiments on the above, it's clear to see that the hierarchical real-time model based on ECOC perform best both in overall accuracy and real-time efficiency.



(a) (b)

Figure 5. (a)The single accuracy of all types of flows in four different experiments. (b)The overall accuracy and real-time efficiency of four different experiments.

4. Conclusion

Multi-class network traffic classification becomes more and more important in recent years, especially the real-time classification is of a great challenge. In this paper, a hierarchical real-time model based on ECOC is proposed to classify the network traffic. Our method performs well compared with other systems not only in overall accuracy, but also in real-time efficiency.

In the future work, two topics should be considered to improve the model's efficiency: (1) It is not determined which feature is more important for the network traffic. (2) Although a number of classification algorithms have been applied to the network traffic classification, we still need to find the new classification algorithms to improve the classification accuracy and generalization ability.

Acknowledgement

Project was supported by the Natural Science Foundation of China (61070130), National Natural Science Foundation of Shandong Province of China (No. ZR2010FM040), Special Funding Project for Independent Innovation Achievements transform of Shandong Province (No. 2012CX30202) and Foundation of Shandong University for Independent Innovation (No. 2012DZ038).

References

[1] Ma J, Levchenko K, Kreibich C, et al. Unexpected means of protocol inference. *Proceedings of the 6th ACM SIGCOMM conference on Internet measurement*. 2006; 313-326.

[2] Xie X, Yang B, Chen Y, et al. Network traffic classification based on error-correcting output codes and nn ensemble. *Fuzzy Systems and Knowledge Discovery 2009*. 2009; 3: 475-479.

[3] Dreger H, Feldmann A, Mai M, et al. Dynamic application-layer protocol analysis for network intrusion detection. *USENIX Security Symposium*. 2006; 257-272.

[4] Moore A W, Papagiannaki K. Toward the accurate identification of network applications. *Passive and Active Network Measurement*. 2005; 41-54.

[5] Moore A W, Zuev D. Internet traffic classification using bayesian analysis techniques. *ACM SIGMETRICS Performance Evaluation Review*. 2005; 33(1): 50-60.

[6] Erman J, Mahanti A, Arlitt M, et al. Identifying and discriminating between web and peer-to-peer traffic in the network core. *Proceedings of the 16th international conference on World Wide Web*. 2007; 883-892.

[7] Bernaille L, Teixeira R, Akodkenou I, et al. Traffic classification on the fly. *ACM SIGCOMM Computer Communication Review*. 2006; 36(2): 23-26.

[8] McGregor A, Hall M, Lorier P, et al. Flow clustering using machine learning techniques. *Passive and Active Network Measurement*. 2004; 205-214.

[9] Erman J, Mahanti A, Arlitt M, et al. Offline/realtime traffic classification using semi-supervised learning. *Performance Evaluation*. 2007; 64(9): 1194-1213.

[10] Park J, Tyan H R, Kuo C C J. Internet traffic classification for scalable qos provision. *IEEE International Conference on Multimedia and Expo 2006*. 2006; 1221-1224.

[11] Auld T, Moore A W, Gull S F. Bayesian neural networks for internet traffic classification. *IEEE Transactions on Neural Networks*. 2007; 18(1): 223-239.

[12] Hou Y, Mai Y. Chaotic Prediction for Traffic Flow of Improved BP Neural Network. *TELKOMNIKA Indonesian Journal of Electrical Engineering*. 2013; 11(3): 1682-1690.

[13] Zhao J, Huang X, Sun Q, et al. Real-time feature selection in traffic classification. *The Journal of China Universities of Posts and Telecommunications*. 2008; 15: 68-72.

[14] Zander S, Nguyen T, Armitage G. Automated traffic classification and application identification using machine learning. *The IEEE Conference on Local Computer Networks*. 2005; 250-257.

[15] Kuncheva L I. Using diversity measures for generating error-correcting output codes in classifier ensembles. *Pattern Recognition Letters*. 2005; 26(1): 83-90.

[16] Escalera S, Tax D M J, Pujol O, et al. Subclass problem-dependent design for error-correcting output codes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 2008; 30(6): 1041-1054.

[17] Escalera S, Pujol O, Radeva P. On the decoding process in ternary error-correcting output codes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 2010; 32(1): 120-134.