Traffic Classification Using En-semble Learning and Co-training

HAITAO HE, CHUNHUI CHE, FEITENG MA, JUN ZHANG, XIAONAN LUO
School of Information Science and Technology
Sun Yat-Sen University
Guangzhou, 510275
CHINA

Abstract: - Classification of network traffic is the essential step for many network researches. However, with the rapid evolution of Internet applications the effectiveness of the port-based or payload-based identification approaches has been greatly diminished in recent years. And many researchers begin to turn their attentions to an alternative machine learning based method. This paper presents a novel machine learning-based classification model, which combines ensemble learning paradigm with co-training techniques. Compared to previous approaches, most of which only employed single classifier, multiple classifiers and semi-supervised learning are applied in our method and it mainly helps to overcome three shortcomings: limited flow accuracy rate, weak adaptability and huge demand of labeled training set. In this paper, statistical characteristics of IP flows are extracted from the packet level traces to establish the feature set, then the classification model is created and tested and the empirical results prove its feasibility and effectiveness.

Key-Words: - traffic classification, ensemble learning, co-training, network measurement

1 Introduction

In recent years, Internet has obtained rapid growth in network scale, number of users and applications. At the mean while, it has been undergone a great evolution, which brings great challenges to network measurement. Traffic classification has attracted a lot of research interests in the past few years, as accurate classification of network traffic is an essential step for many other areas, such as network administration, traffic engineering, security, and QoS control.

In early literatures, port-based identification approach was widely used in network traffic classification, as traditional applications use standard ports assigned by IANA (for instance, WEB traffic uses port 80 and SMTP uses port 25). However, since the year 2002, an increasing number of network applications would no longer use the standard ports for communicating. Moreover, dynamic port allocation technology and camouflage technology have been widely used in order to breakthrough firewalls and other network security equipment. The effectiveness of port-based identification approach was greatly diminished [1, 2], and researches showed that it could not obtain more than 70% accuracy rate nowadays.

An alternative approach is payload-based identification. It identifies network traffic by searching the packet payload for signatures of known applications [3, 4]. This approach is very useful and employed by many commercial bandwidth management products. However, pay-load-based identification has several limitations. First, this technology can only identify network traffic whose signatures are known as prior. Second, it brings great challenge to the processing and storage capacity of network equipment, especially in high-bandwidth environment. Third, the payload encryption technology, the tunnel technology and the evolution of Internet applications have further reduced the effectiveness of payload-based approach.

In the past few years, many researchers turn their attentions to machine learning [5, 6] based approaches, in which the statistical characteristics of IP flows are concerned. Flow characteristics, such as volume, duration and packet size, are extracted from the network data to establish the feature set. Then supervised learning or unsupervised clustering methods are employed to label each flow as a certain application. According to Erman.J [7], there are several reasons why these approaches are recommended. First, different applications have different behaviors and thus exhibit different flow statistics. For example, P2P applications would have larger average packet size while IM client would have a smaller one. Second, although obfuscation of flow statistics is possible, it is generally difficult to implement. Third, classification based on flow statistics can benefit from a lot of work on flow sampling/estimation techniques.

Recent machine learning approaches generally employ only one classifier. There are several limitations for single classifier. First, it's difficult to

improve the classifier accuracy when it exceeds a certain level. Second, it may achieve a fairly high classification accuracy rate in one net-work environment, while it is usually not high in another. Third, a large amount of labeled training data is needed when using supervised learning approaches. It's difficult to obtain a great deal of labeled samples, which is usually hand classified, in the real network environment with high bandwidth and diverse applications. Contrarily, it's very easy to collect unlabeled samples, and it's well worth while to study how to improve the classifier with them.

In this paper, a novel machine learning based traffic classification model is proposed, which combines ensemble learning with semi-supervised co-training techniques. Ensemble learning [8] is a learning paradigm that constructs a set of classifiers and then classifies new examples by taking votes. By combining the predictions of a set of classifiers, ensemble learning can achieve a much better performance than single classifier. Co-training [9] is a semi-supervised learning paradigm that utilizes both labeled and unlabeled samples. Unlabeled samples are used to refine the classifiers. Thus, a high accuracy can be obtained by training with a small number of labeled samples mixed with a large number of unlabeled samples. We evaluate our classification model with one-week traces captured at the edge of south campus of Sun Yat-Sen University in China, and the results show that ensemble learning and co-training techniques do help to improve the performance of network traffic classification.

The remainder of this paper is structured as follows. Section 2 presents our classification model. Section 3 describes the dataset used in this work and flow properties. Then experimental results and analysis will be presented in section 4. Finally, concluding remarks and ideas for future work end this paper.

2 Classification Model

Given a labeled example set $L = \{(x_1,y_1), (x_2,y_2),\ldots(x_{|L|},y_{|L|})\}$ and an unlabeled example set $U = \{x_1 ,x_2 ,\ldots x_{|U|}\}$ mapping function $f:X\rightarrow Y$, which can assign each x_i a correct prediction of predefined class y_i. The x_i values are typically vectors of the form $< x_{i1} , x_{i2} , \cdots , x_{in} >$ with n statistical features of a IP flow and the y values are typically drawn from a discrete set of application classes $y_1 , y_2 , \cdots, y_m$. In addition, we would further consider how to distinguish unknown application types from known ones.

Generally speaking, the accuracy rate achieved by single classifiers is limited. In theory, a learning algorithm can be viewed as searching in a space η of hypotheses to identify the best hypothesis in the space. When the amount of training data available is too small compared to the size of the hypothesis space, the learning algorithm can not find the best hypotheses in η that closed to the real one. In this paper, a paradigm called ensemble learning is employed, which constructs multiple classifiers. By considering all the predictions of those classifiers, this algorithm can "average" their votes and reduce the risk of choosing wrong answer. Moreover, several researches [8] showed that ensemble learning technique can further improve the applicability of algorithm when classifying different data sets.

In traditional supervised learning approaches, a large amount of labelled training samples are needed in learning process, which is difficult to satisfy in the network measurement area. However, unlabeled samples, which may give help to the learning process, are so easy to be collected. An effective way to utilize these unlabeled samples is Semi-supervised learning, which combines labelled and unlabeled samples in learning process. Co-training is a semi-supervised learning paradigm that employs several classifiers. A key benefit of this method is that accurate classifiers can be obtained by training with a small number of labelled data mixed with a large number of unlabeled data.

In this paper, a novel traffic classification approach based on Co-Forest [10] algorithm is proposed, which combines ensemble learning paradigm with co-training techniques. In standard co-training paradigm, two classifiers are firstly trained from L. Then, each of them selects the most confident examples in U to label from its point of view, and the other classifier updates itself with these newly labelled examples. One of the most important aspects in co-training is how to estimate the confidence of a given unlabeled example. In standard co-training, the confidence estimation directly benefits from the two sufficient and redundant attribute subsets, where labelling confidence of a classifier could be regarded as its confidence for an unlabeled example. It is difficult to achieve sufficient and redundant view in traffic classification areas, which is needed in the standard co-training mode. The requirement of sufficient and redundant views greatly reduces the applicability of extending co-training algorithm in the real world.

However, if an ensemble of N classifiers, which is denoted by $H^{\square}$, is used in co-training instead of two classifiers, the confidence could be estimated in an efficient way. When determining the most confidently labelled examples for a component

classifier of the ensemble h_i *(i = 1, 2, · · · , N)*, all other component classifiers in $H^{\square}$ except for h_i are used. These component classifiers form a new ensemble, which is called the corresponding ensemble of h_i, denoted by H_i. Note that H_i differs from $H^{\square}$ only by the absence of h_i. Now, the confidence for an unlabeled example can be simply estimated by the degree of agreements on the labelling, i.e., the number of classifiers that agree on the label assigned by H_i. By using this method, Co-Forest firstly trains an ensemble of classifiers on L and then refines each component classifier with unlabeled examples selected by its corresponding ensemble.

Specifically, in each learning iteration of Co-Forest, the corresponding ensemble H_i examines each example in U. If the number of classifiers voting for a particular label exceeds a preset threshold C, the unlabeled example along with the newly assigned label is then copied into the newly labelled set L'_i. The set $L \square L'_i$. is used for the refinement of h_i in this iteration. Note that the unlabeled examples that are selected by H_i are not removed from U; therefore, they might be selected again by other H_j *(j≠i)* or the corresponding ensembles in the following iterations.

Considering that not all types of applications generating flows are known as priori and new ones may appear over time, a simple strategy is employed to distinguish unknown application types from known ones. When N classifiers examine one example x and the number of classifiers voting for a particular class does not meet a preset threshold θ, we believe that this example belongs to *an unknown application.*

$$H(x)=\begin{cases}\arg\max\sum_{i=0}^{N} f(x) & \geq \theta \\ unknown\ \ application & < \theta\end{cases} \quad (1)$$

In the ensemble learning process, an important a spect is maintaining the diversity among the classifi ers in order to keep the voting system effective. If e very classifier is similar to each other, the classifiers would make a same prediction to an example, just a s the single classifier mode. In this paper, an ensemb le strategy similar to Bagging [11] is employed to m aintain the diversity among classifiers. For each com ponent classifier h_i (i = 1, 2, · · · , T), a training set w ith size N is resampled from the training data with b ootstrap sampling strategy. Different from Co-Fores t, a further technique to maintain the diversity of cla ssifiers is employed. This technique, which is simila r to cross-validated committees [12], is to manipulat e the set of input features available to the learning al gorithm.Firstly, the input features are divided into N subsets s_i (i = 1, 2, … , N), then the classifier h_i sele cts the feature set S_i for training. It is noted that S_i se lects all the subsets only by the absence of s_i. Pseud o-code for classification procedure is shown in Fig.1 .

Input: *the labeled set L, the unlabeled set U, the confidence threshold C, the number of classifier N*
Process:
*/** Construct a ensemble classifier consisting N classifier */*
for $i \square \{1,...,N\}$ *do*
$S_i \leftarrow BootstrapSample(L)$
$h_i \leftarrow BuildClassifier(S_i, Attributes(i))$
end for
*/** Initialize some variables before Co-Training */*
for $i \square \{1,...,N\}$ **do**
$e_{0,i} \leftarrow 0.5$
$t \leftarrow 0$
end for
*/** Execute Co-Training until none of classifiers can learning anything from unlabeled data */*
Repeat until *none of classifiers changes*
$t \leftarrow t+1$
for $i \square \{1,...,N\}$ **do**
$e_{t,i} \leftarrow MesuraOutOfBagError(H_i, L)$
$L_{t,i} \leftarrow \theta$
if $(e_{t,i} < e_{t-1,i})$
$U'_{t,i} \leftarrow SubSampled(U, e_{t-1,i}|L_{t-1,i}|/e_{t,i})$
for *each* $x_u \square U'_{t,i}$ **do**
if $(Confidence(H_i, x_u) > C)$
$L'_{t,i} \leftarrow L'_{t,i} \square \{(x_u, H_i(x_u))\}$
end for
end for
for $i \square \{1,...,N\}$ **do**
if $(e_{t,i}|L_{t,i}| < e_{t-1,i}|L_{t-1,i}|)$
$h_i \leftarrow BuildClassifier(L \square L'_{t,i}, Attributes(i))$
end for
end of repeat
Output: *a ensemble classifiers consisting N classifiers*

Fig. 1- Pseudo-code of the algorithm

3 Data Set

Several network traffic traces are used for training and evaluating our classification model. The traces are captured at the edge of the campus network of Sun Yat-Sen University, which present a snapshot of the traffic going bidirectionally. Considering the day-pattern and week-pattern of network traffic, we captured 21 5-minute traces during January 7-13, 2008, separately on 04:00 AM, 10:00 AM and 22:00 PM in each day. For the high-bandwidth of the network (>200Mbps) and the limited disk capacity of our network measurement system, only the first

150 bytes of each packet were captured. Table 1 shows the summary of the 5-minute traces on 10:00 AM.

	Aggregate			Two-Way		
	Flows	Bytes	%TCP flows	Flows	Bytes	%TCP flows
Jan.2008,7	633k	14804GB	85.09	197k	4717GB	85.42
Jan.2008,8	671k	15192GB	78.71	216k	5081GB	79.12
Jan.2008,9	613k	14974GB	76.63	201k	4878GB	76.98
Jan.2008,1(	656k	14940GB	72.2	221k	4822GB	72.62
Jan.2008,11	606k	15978GB	74.2	208k	5858GB	74.61
Jan.2008,12	707k	16771GB	71.74	233k	6757GB	72.16
Jan.2008,13	612k	17425GB	74.18	195k	7302GB	74.56

Table 1 - Summary of the traces

In this paper, we only focus on several TCP applications as shown in Table 2. It should be noted that despite of Web applications, all the others are typical and popular in China. Specifically, Xunlei, which is based on P2SP techniques, has replaced BitTorrent and Emule and been the most popular file-sharing application. QQ is an Instant Messaging tool widely used by Chinese, with more than 270 million active users. And PPLive is the largest P2P network television in China. By the way, the applications are hand classified, while other applications which cannot be identified are labelled as *unknown.*

	Flows		Bytes(MB)	
	Amounts	Proportion	Amounts	Proportion
Web	920430	50.42%	21102.4	18.59%
PPLive	32481	1.78%	2267.3	2.00%
Xunlei	215892	11.83%	40406.9	35.59%
FTP	20698	1.13%	12.94	0.01%
MSN	3356	0.18%	10.19	0.01%
QQ	14887	0.82%	75.11	0.07%
QQGame	1002	0.05%	35.91	0.03%
Unknown	616681	33.78%	49621.6	43.71%
Total	1825427	100.00%	113532	100.00%

Table 2 - Application statistics of dataset

In this paper, IP flow is defined as a bidirectional exchange of packets between two nodes according to the 5 tuples (*srcIP, desIP, srcPort, desPort, protocol*). Note that, only the two-way flows are concerned, for they take up the most volume of the trace. Considering flow characteristics which are independent to packet payload and easy to compute, the following statistical characteristics of IP flow are taken into account.

- Flow volume in packets and bytes
- Flow duration and rate (bytes/duration)
- Upload/download bytes
- Mean and std. of packet length in the flow
- Mean and std. of packet interval in the flow
- Distribution of packets in the flow (density [13], burstiness [14])

As each flow consists of two directions (upload and download), characteristics are calculated in both directions (except upload/download bytes).

4 Experiment results and analysis

In many papers, the distribution of training samples in each application is dependent on their actual proportion in the network. It sounds reasonable, but the numbers of samples of some dominate applications, such as Web and Xunlei, may be much larger than others, even tens of thousands of times, so the classification result will be heavily partial to the classes with more training samples. In this case, the class with few examples may not be correctly identified. However, it won't influence the totally high accuracy due to their few bases. An extreme condition is that high classification accuracy can be obtained with all the weak application samples mis-classified. The biased training sample strategy would not do harm to entire accuracy of classifier, while it is harmful for us to evaluate the effectiveness of a classification method. Based on this consideration, this paper establishes the training set with an un-biased strategy, in which 300 labelled examples of each traffic class would be selected (If an application have less than 300 examples, all of its examples would be selected).

To evaluate the validity our classifier model in different network circumstance, the data sets of the three periods (04:00 AM, 10:00 AM, and 10:00 PM) are tested respectively. According to the unlabeled rate, the training set would be randomly separated into labelled set L and unlabeled set U. Different ratio of (20%, 40%, 60%, 80%) are tested to see its influence to the classification model. Note that, the algorithm would be tested for 20 times, and the result averages their error rate.

4.1 Performance comparison

According to Zhou et al. [15], a large size of ensemble does not necessarily lead to better performance of an ensemble. Thus, the ensemble size N in Co-Forest is not suggested to be too big. In the experiments, the value of N is set to 6, and the threshold C would take the value of 0.75. When 4 or more classifiers predict an example as application a, it will be labelled as a.

In order to prove that our method performs better than the previous machine learning approaches, some traditional algorithms (J4.8, Random Tree, Bagging, and Random Forest) in WEKA are used for comparison. Random Tree and Bagging are ensemble learning methods, which are used to compare with our ensemble learning method combined with co-training technique. J4.8 and Random Tree are famous supervised learning methods, which would be compared with the ensemble learning methods. In our experiments, Bagging uses J4.8 as its basic classifier, while Random Forest uses Random Tree. Both of the two ensemble learning approaches employ 6 classifiers as our method. Note that, these 4 algorithms for comparison only use the labelled set *L* for training.

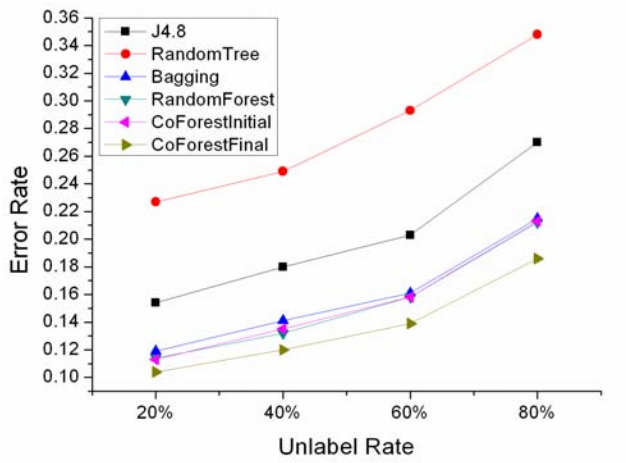

Fig. 2 - Performance comparison of classifiers

All the experiments got the similar results. Because of space limitation, only the results about datasets on 10:00 AM are listed in Table 3. In the Co-Forest columns, initial and final denote the error rates before and after co-training respectively, while improve shows the improvement brought by the co-training process. As the randomness of the training examples, sometimes the error rate would increase after co-training. However, these are minority phenomena, and the average result of 20 runs proves that co-training technique do improve the performance of the classification model with different unlabeled rate.

Learned from Table 3, we find that ensemble learning methods (Bagging, Random Forest, and Co-Forest) perform much better than supervised learning methods (J4.8 and Random Tree). Bagging gets 10 percent error rate lower than J4.8 averagely, while Random Tree performs better than Random Tree over 14 percent averagely. It's clear that ensemble learning can improve the performance of traffic classification, when the accuracy of single classifier is limited. We can also see that, co-training technique further improves the performance of ensemble learning. The error rate of Random Forest and the initial of Co-Forest are almost the same. Nevertheless, after co-training, the error rate of Co-Forest decreases obviously. With different unlabeled rate, Co-Forest can get approximately 11 percent error rate decline after co-training process, on average. It indicates that the unlabeled data can help to improve the performance of classification model. An intuitive comparison of the algorithms is showed in Fig. 2.

Unlabel Rate	Supervised Learning		Ensemble Learning				
	J4.8	Random Tree	Bagging	Random Forest	Co-Forest		
					initial	final	improve
20%	###	0.23	0.12	0.11	0.11	0.10	8.28%
40%	###	0.25	0.14	0.13	0.14	0.12	11.01%
60%	###	0.29	0.16	0.16	0.16	0.14	11.95%
80%	###	0.35	0.22	0.21	0.21	0.19	12.62%
Avg.	###	0.28	0.16	0.15	0.16	0.14	11.30%

Table 2 - Error rate comparison of classifiers

4.2 Adaptability comparison

In order to evaluate the adaptability of classification model, the training and test data would be selected from different networks. In this subsection, the training data is derived from the traces of female dormitories and the test data from male dormitories. The result of 10:00 AM is showed in Table 4. The accuracy of J4.8 and Random Tree get a remarkable decline over 7 percent, while the other 3 algorithms using ensemble learning get a much smaller decline. Further more, by employing co-training technique, Co-Forest gets the smallest decline among them. From the empirical results, we can suppose that ensemble learning do improve the performance of classifiers in different network environment, and co-training can further help to improve it.

Unlabel Rate	Supervised Learning		Ensemble Learning		
	J4.8	Random Tree	Bagging	Random Forest	Co-Forest
20%	7.36%	10.56%	2.17%	1.71%	1.49%
40%	6.93%	8.12%	2.68%	2.09%	1.09%
60%	8.95%	5.62%	2.32%	2.38%	1.31%
80%	7.74%	5.71%	1.86%	1.99%	1.57%
Avg.	7.74%	7.50%	2.26%	2.04%	1.37%

Table 4 - Accuracy decline comparison of classifiers in different network environment

4.3 Classify more traffic

Finally, our algorithm was used to classify all the captured data. Table 5 shows the classified result of female dormitories on 10:00 AM. Compared with the hand classified data set, our classify model can identify much more flows (about 20% of the total flows) than the hand classify method that based on payload signature. The flow amounts of all applications increase, except QQGame. We suppose that it is caused by the deficiency of labelled examples (only 86) of QQGame in the training process. Besides, our algorithm can label the low-confidence examples as unknown, which are considered as other applications different from the predefined ones.

	Actual Traffic		Labeled Set		Increase	
	Flow	Ratio	Flow	Ratio	Flow	Improve
Web	188399	62.24%	174972	57.80%	13427	7.67%
PPLive	3749	1.24%	3351	1.11%	398	11.88%
Xunlei	63282	20.90%	30428	10.05%	32854	107.97%
FTP	13503	4.46%	3306	1.09%	10197	308.44%
MSN	1268	0.42%	406	0.13%	862	212.32%
QQ	3297	1.09%	1785	0.59%	1512	84.71%
QQGame	57	0.02%	86	0.03%	-29	-33.72%
All Labled	273555	90.37%	214334	70.80%	59221	27.63%
Unlabled	29164	9.63%	88385	29.20%		
Total	302719	100.00%	302719	100.00%		

Table 5 - Results of actual traffic classification in female dormitories

5 Conclusion and future work

This paper presents a novel traffic classification model that combines ensemble learning with co-training techniques. The experiment results show that this method can achieve a higher accuracy than single classifier approaches, with better adaptability in the different network environment and the ability to identify unknown applications, which is very important to network measurement in the Internet with great evolution. Although this classification model performs much better than traditional methods, many problems are left for further works, such as 1) in the ensemble learning mode, how to maintain the diversity of classifiers to the great extent, 2) and in the co-training mode, how to utilize the unlabeled data more effectively. We believe it is worthy of being studied in the future.

References:

[1] Karagiannis T, Konstantina, and Papagiannaki, BLINC: Multilevel traffic classification in the dark, *SIGCOMM'05*, Philadelphia, USA, 2005.

[2] Karagiannis T, Broido A, and Faloutsos M, Transport layer identification of P2P traffic, *IMC'04*, Taormina, Sicily, Italy, 2004. pp. 121-134.

[3] Sen A, Spatscheck O, and Wang D, Accurate, scalable in-network identification of P2P traffic using application signatures, *WWW'04*, New York, USA, 2004, pp. 512-521.

[4] Haffner P, Sen S, and Spatscheck O, ACAS: Automated construction of application signatures. *SIGCOMM'05*, Philadelphia, USA, 2005, pp.197-202.

[5] Bernaille L, Teixeira R, and Akodkenou I, Traffic classification on the fly. *Computer Communication Review,* Vol. 36, No. 2, 2004, pp. 23-26.

[6] Crotti M, Dusi M, and Gringoli F, Traffic classification through simple statistical fingerprinting, *Computer Communication Review*, Vol. 37, No.1, 2007, pp. 7-16.

[7] Erman J, Mahanti A, and Arlitt M, Identifying and discriminating between web and peer to peer traffic in the network core, *WWW'07*, Banff, Alberta, Canada, 2007.

[8] Dietterich T G. Ensemble learning, *In The handbook of brain theory and neural networks*, 2nd ed. MIT Press, 2002.

[9] Blum A and Mitchell T, Combining labeled and unlabeled data with co-training, *Proc. of the eleventh annual conference on computational learning theory*, Madison, Wisconsin, USA, 1998, pp. 92-100.

[10] Li M and Zhou Z, Improve computer-aided diagnosis with machine learning techniques using undiagnosed samples, *IEEE Transactions on Systems, Man and Cybernetics - Part A*, Vol. 37, No. 6, 2007, pp. 1088-1098.

[11] Breiman L, Bagging Predictors, M*achine Learning*, Vol. 24, No. 2, 1996, pp. 123-140.

[12] Parmanto B, Munro PW and Doyle HR, Improving committee diagnosis with resampling techniques, *Advances in Neural Information Processing Systems*, Vol. 8, pp. 882-888, 1996.

[13] Ribeiro V J, Zhang Z-L, and Moon S, Small-time scaling behavior of Internet backbone traffic, *Computer Networks*, Vol. 48, No. 3, 2005, pp. 315-334.

[14] Lan K C and Heidemann J, A measurement study of correlations of Internet flow characteristics, *Computer Networks*, Vol. 50, No. 1, 2006, pp. 46-62.

[15] Zhou Z-H, Wu J, and Tang W, Ensembling neural networks: many could be better than all, *Artif. Intell.*, Vol. 137, No. 1-2, 2002, pp. 239-263.