IP Traffic Classification Based on Machine Learning

Donghong Qin and Jiahai Yang
Department of Computer Science and Technology
Tsinghua University
Beijing, P. R. China
donghong.qin@gmail.com

Jiamian Wang and Bin Zhang
Network Research Center
Tsinghua University
Beijing, P. R. China
yang@cernet.edu.cn

Abstract—**With the rapid development of Internet, many network applications (e.g., P2P) use dynamic ports and encryption technology, which makes the traditional port and payload-based classification methods ineffective. Hence, it is important and necessary to find the more effective ones. Currently the machine learning (ML) techniques provide a promising alternative one for IP traffic classification. In this work, we use the ML-based classification method to identify the classes of the unknown flows using the payload-independent statistical features such as packet-length and arrival-interval. In order to improve the efficiency of the classification methods, the feature reduction techniques are further adopted to refine the selected features for attaining a best group of features. Finally we compare and evaluate the ML classification algorithms based on the BRASIL data source in terms of the three metrics such as overall accuracy, average precision and average recall. Our experiments show that the decision-tree algorithm is the best ML one for IP traffic classification and is able to construct the real-time classification system.**

Keywords-IP traffic flow classification; ML algorithm; performance evaluation; features optimization

I. INTRODUCTION

In recent years, Internet has undergone dramatic increases in terms of the number and type variety of applications, whose scopes [1] include interactive (e.g., telnet, games, etc.), bulk data transfer (e.g., ftp and p2p file downloads, etc.), collaborative (e.g., mailing lists), real-time applications (e.g., VoIP, video streaming, etc.), and so on. These applications mostly require IP network to provide better Quality of Service (QoS) services. Nowadays, network operators are also actively seeking for most important applications to provide different QoS service capabilities, then obtaining the new add-valued business benefits. Although current academia and industry have provided several QoS mechanisms (e.g., diffserv[2, 3]) which have not been widely deployed mainly because it is still hard to implement the QoS-guaranteed applications. A basic barrier behind this situation is that there lacks an effective classification method to identity or classify the special one from the aggregated traffic.

The traditional traffic classifications based on the well-known TCP or UDP port are becoming increasingly less effective because the growing networked applications are using random port numbers [1, 4]. To address this shortage of the port-based classification, many researchers present a reliable method named the deep packet inspection (DPI)[3] which is a packet-payload-based match classification. However the legal privacy and widespread encryption are taken into account, the DPI technology will be greatly restricted. Therefore future classification methods must be transparent and free of port and payload. Currently traffic classification methods using ML techniques are widely and deeply investigated [4-6]. These methods generally include four important steps: (1) Define some important features such as packet lengths, inter-packet arrival interval; (2) Construct an ML-based model; (3) Training the model to attain the ML classifier associating a group of features with the known traffic classes; (4) Using the above classifier to identify or classify the unknown traffic flows in the intellectual classification system.

In this work, our main goal selects an effective ML-based classification method and studies various ML algorithms in terms of performance metrics such as overall accuracy, average precision and average recall. Further we want to attain an effective and efficiency ML algorithm for the Intellectual Traffic Flow Classification System (ITFCS). Further we apply the feature reduction techniques [12] into designing the ML algorithm, which not only improve the efficiency but also at the same time maintain a fair good performance of the algorithm. The remainder of the paper is organized as follows. Section Ⅱ discusses the related work. Section Ⅲ elaborate ML algorithms and its related features. Section Ⅳ presents and analyses our experimental results. Finally, Section Ⅴ concludes this paper and gives future directions.

II. RELATED WORK

Currently, the ML-based classification methods have attracted many researchers and there are lots of the related works [4-8]. Williams et al. [4] investigate and compare the performance of the four ML algorithms. Their results demonstrate it is useful to differentiate algorithms based on both computational performance and classification accuracy. They concern about the four special applications. Differently we focus on an application category instead of special applications in this work. We believe that there are two reasons as follows (a) a special protocol probably describes the implementation of one application which is not bounded by that protocol; and (b) there are many change and adaptation in the specific details of an application implementation. Therefore there exist the considerable varieties of both the different implementations and protocol behaviors of the application. So we focus on the traffic flow category rather than special application. Lim et al. [7] conduct an extensive survey of 33 algorithms across 32 diverse datasets. They find that the algorithms show similar classification accuracy but quite different training performance. Li and Canini [8] study the effective and efficient classification of network-based applications. And then a four-way comparison of application-identification methods such as Port-based, DPI, Naive Bayes and C4.5 decision tree, are studied in detail. In this paper, we assume that the network applications are classified into several special categories such as inter, service, multimedia and bulk by their QOS requirements. Based on the same data source and selected features, we try to find the best method from the existing ML algorithms and construct the key classification module for our future intelligent classification systems.

III. TRAFFIC FLOW CLASSIFICATION MODEL

A. *Machine Learning Algorithms*

Machine learning (ML) [9] techniques provide a promising alternative method used to classify flows based on independent statistical features such as packet length and arrival intervals. ML algorithms for IP traffic classification generally fall into two categories: supervised and unsupervised, based on whether the manual intervention is needed or not. Since our experimental datasets already include the label class of network flow data, we only study the supervised ML algorithms. Below briefly describe the basic principles of the used ML algorithms.

Naïve-Bayes(NBD, NBK) is a classification method based on the Bayesian theorem[9]. It calculates and analyses the relationship between each attribute and the class of the sample. From the computing results, it can derive a conditional probability of an attribute and the class, which is the features' prior knowledge of the Naïve-Bayes classifier. In the classification process, the classifier must estimate the probabilities of the the unknown sample instance as a class, by combining the prior knowledge with the actual value of the unknown sample instance. Moreover the classifier must estimate the probabilities of the feature having a certain value. The continuous feature can have a large (or possibly infinite) number of values, thus the probability cannot be estimated from the frequency distribution. Nowadays there are two solutions for this problem: by fitting the continuous probability distribution, or by using the discretization techniques. Because the latter method transforms the continuous features into the discrete ones and does not require the distribution model, our work uses the the discretization method.

Decision Tree (DT) is an important and effective ML method. It constructs a classification model based on a tree structure [9, 10]. In the DT model, a node represents a certain feature, and a branch is the relevant condition threshold that partitions the instance sample in this level. A leaf denotes one class and it terminates by traversing a series of nodes and branches. If the sample traverses from the root node to the special leaf node, the class of the leaf node is the sample's class (also see Fig.3).

Nearest Neighbor (NN). In 1968, Cover and Hart propose the NN algorithm. It is a basic and simple ML classification algorithm in the pattern recognition field. Assume that there is a classification problem including c_1, c_2, ..., c_m class, and each class has the samples of N_i, for i = 1, 2, ..., m. We can design and require the discriminator function of the c_i class: $d_i(x) = \min_i abs(x - x_i^j), j = 1,2,...,N_i$, where the subscript i denote the c_i class, j is the j[th] of the N_i samples of the c_i class. Farther, this classification function can be written as: $d_i(x) = \min_i d_i(x), i = 1,2,...,m$, that is , for the unknown sample x, if x has the minimal Euclidean distance between the sample and the class center, its class is the one belonging to the c_i center. So this decision method is named as the Nearest Neighbor. Actually the generalization of Nearest Neighbor algorithms, namely the k-NN algorithm, is often used, because the k-NN algorithm can enhance the robustness of the models. Especially, on the low dimensional classification, k-NN is a highly good and extensively used method.

Support Vector Machine (SVM). The SVM is a pattern recognition method based on the statistical learning theory (STL). Although the classification of the low dimensional space is difficult, if the low dimensional space is transferred into the high dimensional one, the classification of the high dimensional space becomes easy relatively. As such, the result brings about the computation overhead; the best solution is designing or selecting the appropriate kernel functions. Intuitively, an SVM model is a classification algorithm for the classification of the samples space, and it requires that the samples of the different categories are divided as widely as possible by the support vector. Briefly speaking, SVM construct the optimal hyper-plane in the sample space and make it have maximal distance with the other different classes' ones, thus achieving the maximal generalization capability.

Linear Discriminator is a relative simple discriminator function. Assume that $g(x) = \omega^T x + \omega_0$ is a linear function of the x vector, where w^T and w_0 is called as the coefficient vector. For the m-class classification problem, one can define m discriminator function. In the training phase, the discriminator function is obtained by using the sample dataset to estimate the parameter of w_i and w_{i0}. And then, for the unknown sample x, it belongs to the class of the largest discriminator function.

Artificial Neural Networks (ANN) consists of an interconnected group of artificial neurons, and it processes and computes the information using a connection approach. In the most cases an ANN is an adaptive system that changes its structure based on external or internal information that flows through the network during the training phase. ANN provides a common and practical learning method from the samples. The simplest ANN is a single-layer perception network only including the input and output layer. Let the input mode be the n-dim vector $x = (x_1, x_2, ..., x_n)'$, that is, the input layer contains *n* nodes. If the output mode has *m* class, there are generally *m* output layer neurons and the output of these neurons is determined by the linear threshold function [9].

B. Data Source

Our dataset is taken from the publicly available BRASIL dataset [11], which was captured in different days and at the different sites, including the original and processed traffic data. In the original dataset, each traffic flow has 249 features while each traffic flow has 12 features and a hand-verified class-label (ground-truth) in the processed one. According the above-mentioned focus of this work, our dataset chooses four common and important categories including Bulk, Interactive (ab., Inter.), Service and Multimedia. Table 1 shows the four categories' QoS requirement. Although in the current Internet they have important representativeness because of their different QoS requirements, and if network management system could identifies and classifies these application categories, Internet would provide the available QoS service for these afforded users whose applications require the better performance.

TABLE I. VARIOUS APPLICATIONS CATEGORIES' QOS

Class	Delay	Interactive	Bandwidth	Reliability
Inter.	High	Many	Low	Mid
Bulk	-	Less	High	High
Multimedia	High	Mid	Mid	Mid
Service	Mid	Mid	Low	High

Moreover, in order to characterize each application class, we need each class's reference data and extract a set of representative features from them. Selecting the network traffic based on port numbers may not yield the reliable statistics that represent the forenamed class; in turn the classification process needs effective reference data. To break this circular dependency, we select some per-class applications based on their typical usage and popularity, thus they have a low likelihood of being contaminated by other class application traffic. The reference applications will be used to estimate a number of statistics features such as flow-size and packet-arrival-interval. Based on the above criterion, the reference applications of each class are selected as follows: interactive (e. g., telnet), bulk (e.g., ftp), multimedia (e.g., real-media), service (e.g., dns). These applications data are easily separated and extracted from the BRASIL dataset, finally obtaining the dataset named the TCM dataset.

C. Feature Set Selection

The features selection must consider whether they fully reflect the essential characteristic of network traffic flows and how they influence mutually. To avoid starting from scratch, we try to begin with the existing works [4] and call their selected feature as the BFS feature set, see the table 2. Although the subsequent experimental results (see IV.B) show the good classification performance, the deficiency of the various ML algorithms is worth to further study because it associates with the real-time use of the ML classification system. The classification efficiency of the ML algorithms mainly depends on the type and number of the selected features.

TABLE II. THE BFS FEATURE SET

Abbreviation	Description
fpackets	Number of packets in forward direction
maxfpktl	Maximum forward packet length
minfpktl	Minimum forward packet length
meanfpktl	Mean forward packet length
stdbpktl	Standard deviation of backward packet length
minbpktl	Minimum backward packet length
protocol	Protocol

Some research results [12] show that many traffic flow features have some correlation to a certain extent, so how to eliminate the correlation of the features and select the best feature combination are an important and key research in the feature extracting technology. It is important how to select a few representative and independent features for the classification method. To improve the algorithms' efficiency and reduce executing time, we try to use some feature analysis techniques to select best classification features. In pattern recognition, there are two famous and important feature reduction methods [12]: Correlation-based Feature Selection (CFS) and Consistency-based Feature selection (CON). They both evaluate different combinations of traffic flow features to identify an optimal feature subset, and different subset search techniques such as forward and backward are used in the evaluating processes. Many researches show that CFS is a better one than CON [4], so we further refine the above set using the CFS technique and obtain the new feature set which include: serv_port, max_data_wire, max_data_ip, max_segm_size_ab, req_sack_ba, q3_data_wire and ar_data_wire. We find that it has higher classification accuracy, but the serv_port feature is depend to the port of server; therefore after removing the server and client ports the new feature subset is again obtained. After this adjustment the selected features are mss_requested_ab, min_segm_size_ab, min_segm_size_ba and duration. According to our multiple experience results, we finally choose the new feature sets shown in the table 3, called as FRS.

TABLE III. THE FRS FEATURE SET

Abbreviation	Label	Description
max_data_ip_ab165	C	Maximum of total bytes in IP packet
var_data_ip_ba187	E	Variance of total bytes in IP packet
min_segm_size_ab83	D	The minimum segment size observed during the lifetime of the connection. (client->server)
min_segm_size_ba84	B	The minimum segment size observed during the lifetime of the connection. (server->client)
prtcl	A	Protocol number

IV. EXPERIMENT AND EVALUATION

A. Evaluation Method and Perforance Metric

The key criterion differentiating the performance of classification model (or classifier) is predictive accuracy (i.e., how accurately a classification model makes decisions when presented with previously unseen data). A number of metrics can express the predictive accuracy. Assume there is a traffic class X in which we are interested, mixed in with a broader set of IP traffic. A traffic classifier is being used to identify (classify) packets (or flows of packets) belonging to class X when presented with a mixture of previously unseen traffic. The classifier is presumed to give one of two outputs - a flow (or packet) is believed to be a member of class X, or it is not.

A common way to characterize a classifier's accuracy is through metrics known as Accuracy, Precision and Recall. These metrics are defined as follows:

- Accuracy: Percentage of correctly classified instances among the total number of instances.
- Recall: Percentage of members of class X correctly classified as belonging to class X.
- Precision: Percentage of those instances that truly have class X, among all those classified as class X.

In order to study and compare various algorithms' performances, we use k-fold cross validation method [12]. In the validation process, the whole dataset is divided into k subsets. Each time one of the k subsets is used as the test data and other k-1 subsets form the training data. Performance statistics are averagely calculated across all k trials. We believe, these metrics provide a good indication of how well the classifier will perform on the unseen data.

B. Experiment Evaluation

Usually the evaluation experiment needs a number of the samples, due to the limited number of the samples, we adopt the k-fold (k=10) cross-validation to assess the performance of the ML algorithms. The preliminary evaluation experiments of various algorithms use the TCM dataset and the BFS feature-set and we acquire the classification performance such as accuracy, precision and recall. TABLE 4 shows various algorithms' performance.

TABLE IV. ALGORITHM PERFORMANCE

Name	Accuracy(%)	ClassPrecision(%)	ClassRecall(%)
DT	96.50±3.91	100, 91,100,96	90, 100, 96, 100
NBK	93.75±6.32	100, 100,80,97	100, 90, 100, 85
NN	92.50±4.03	96, 91, 90,93	96, 98, 90, 86
LSVM	80.00±5.00	98, 98, 100, 66	92, 88, 70, 98
ANN	77.50±5.12	78, 80, 76, 77	86, 64, 94, 66
LDA	75.00±9.80	79, 70, 81, 82	74, 74, 88, 70
NBD	71.00±9.27	69,71, 79, 67	78, 86,67,70

Seen from the table above, DT and NBK have better performance than other ones, whose accuracy are 96.5% and 93.7% respectively. But ANN, LDA and NBD are merely less than 80%. Generally speaking, DT and NBK are used as good traffic classification algorithms because of their higher accuracy more than 90%. Especially the DT performance metrics such as precision and recall list in the table 5. For precision metric, we know that the precision of Inter and Multimedia have arrived at 100%, that is, for these class, the DT algorithm can almost completely differentiate and classify them. On the other hand, for the recall metric, Service and Bulk are also 100%, indicating that the DT is able to identify these two classes completely. For the other case, the precision and recall metrics are also more than 90%. Therefore, DT has fairly good performance for this classification experiment.

TABLE V. THE DT PERFORMANCE RESULTS

	Inter.	Service	Multimedia	Bulk	Precision (%)
Inter.	45	0	0	0	100.0
Service	5	50	0	0	90.9
Multimedia	0	0	48	0	100.0
Bulk	0	0	2	50	96.1
Recall (%)	90.0	100.0	96.0	100.0	

C. BFS and FRS Comparation

In order to evaluate the overall accuracy of various ML algorithms, we have done many experiments based on the BFS and FRS feature sets. Figure 1 compares the accuracy for each ML algorithm when using the BFS and FRS feature set. The DT, NBK and NN algorithms achieve greater than 90 % accuracy using the BFS set, and there is little change when using the FRS set. The LSVM, ANN, NBD and LDA do not perform better possibly due to the use of different traffic classes, features and equally weighted classes. Although the FRS feature set has fewer features than the BFS, its overall accuracy is as nearly good as the BFS.

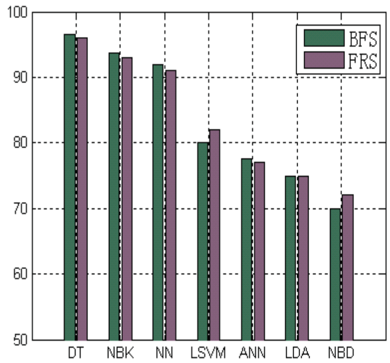

Figure 1. Accuracy comparison of the two feature sets

The Fig.2 shows the performance comparison of the DT model based on the two feature sets. Seeing from the figure 2, although the FRS-based performance using are not as same as the BFS one, their difference is very subtle in terms of metrics such as accuracy, precision and recall, respectively, 1.00%, 1.11% and 1.00%. In the practical classification system, the FRS-based DT model (FRS-DT) is better choice for the real-time classification because the computational efficiency is top-priority than others.

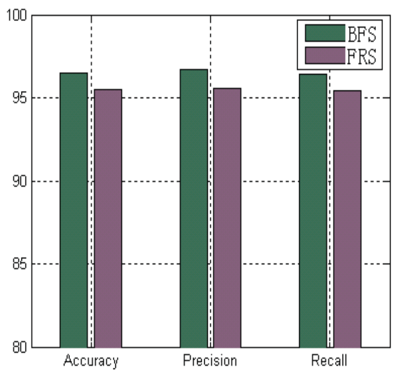

Figure 2. The classification performance comparison

D. FRS-DT-based Classification Model

According to the above results, the Decision Tree classification model has the best performance whatever accuracy, recall, precision and efficiency. Therefore we consider using DT model to construct the future classification system. Through our experiment, we have gotten the FRS-DT model of the Figure 3, where the nodes meaning of A, B, C, D and E are shown in the table 3 above. FRS-DT will be used to predict the class of the test dataset, and the classification results see in the table 6. Seen from the table 6, FRS-DT have good performance, for example, its precision and recall metrics are more than 95%(except for the Inter's precision) while its recall is also higher than 97%. From these experiment results, we believe that FRS-DT is an effective and efficiency classification method, so it will possibly be used to construct the real-world traffic flow classification in the future.

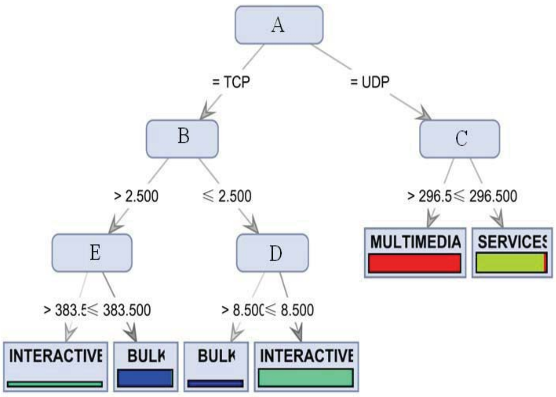

Figure 3. The FRS-DT model diagram

TABLE VI. THE PERFORMANCE OF THE FRS-DT MODEL

	Bulk	Inter.	Service	Multimedia	Precision (%)
Bulk	3131	3	0	0	99.9
Inter.	29	170	0	0	85.4
Services	0	0	51	2	96.2
Multimedia	0	0	0	71	100.0
Recall (%)	99.0	98.2	100.0	97.2	

V. CONCLUSIONS

Nowadays IP network adopts the best-effort service to forward data packets, but it can hardly ensure the throughput, delay, jitter and loss-rate of network applications. So it leaves end-system to deal with the various performance problems. If the networks can identify and classify the application class and adopt the available mechanism (e.g. diffserv) to ensure the application's QoS requirement, this will start the new add-value services for special applications. Therefore the traffic classification capability is one of the primary key technologies for the QoS-aware applications. In this paper, we widely and deeply investigate the ML-based classification technologies, based on the payload independent statistical features. We study and evaluate some ML algorithms' performances. Experimental results show that the DT model has very high accuracy and are very promising in the traffic flow classification fields. And we observe that the DT algorithm, whatever performance or efficiency of traffic classification, is the highly promising ML classification method. In the future work, we will further to base on the current work, and develop the DT-based classification prototype system.

ACKNOWLEDGMENT

This work is supported by the National Basic Research Program of China under Grant No. 2009CB320505, the National Science and Technology Supporting Plan of China under Grant No. 2008BAH37B05, and the National High-Tech Research and Development Plan of China under Grant No. 2008AA01A303 and 2009AA01Z251.

REFERENCES

[1] M. Roughan, S. Sen, O. Spatscheck, and N. Duffield, "Class-of-service mapping for qos: a statistical signaturebased approach to ip traffic classification," in Proceedings of the 4th ACM SIGCOMM conference on Internet measurement, pp. 135–148, ACM, 2004.

[2] Y. Bernet, J. Binder, S. Blake, M. Carlson, B. Carpenter, S. Keshav, E. Davies, B. Ohlman, Z. Wang, and W. Weiss. A framework for differentiated services. Internet Draft, February 1999. http://search.ietf.org/internet-drafts/draft-ietf-diffserv-framework-02.txt.

[3] S. Blake, D. Black, D. Black, H. Schulzrinne, D. Black, M. Carlson, E. Davies, Z. Wang, and W. Weiss. Rfc 2475 - an architecture for differentiated service, December 1998. Available at http://www.faqs.org/rfcs/rfc2475.html.

[4] 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, vol. 36, no. 5, p. 16, 2006.

[5] T. T. T. Nguyen and G. Armitage, "A survey of techniques for internet traffic classification using machine learning, "IEEE Communications Surveys & Tutorials, vol. 10, no. 4, pp. 56–76, 2008.

[6] A. Moore and D. Zuev, "Internet traffic classification using bayesian analysis techniques," in Proceedings of the 2005 ACM SIGMETRICS international conference on Measurement and modeling of computer systems, p. 60, ACM, 2005.

[7] T. Lim, W. Loh, Y. Shih, "A Comparison of Prediction Accuracy, Complexity, and Training Time of Thirty-three Old and New Classification Algorithms", Machine Learning, volume 40, pp. 203-229, Kluwer Academic Publishers, Boston, 2000.

[8] W. Li, M. Canini, A. W. Moore and R. Bolla, "Efficient application identificationu and the temporal and spatial stability of classification schema", Computer Networks, vol. 53, no. 6,pp. 790-809,20008.

[9] T. M. Mitchell, "Machine learning", McGraw-Hill Education (ISE Editions). December 1997.

[10] R. Kohavi and J. R. Quinlan, Will Klosgen and Jan M. Zytkow, editors, "Decision-tree discovery", in Handbook of Data Mining and Knowledge Discovery, pp. 267-276, Oxford University Press, 2002.

[11] "http://www.cl.cam.ac.uk/research/srg/netos/brasil/."

[12] I. Witten and E. Frank, Data Mining: Practical machine learning tools and techniques. Morgan Kaufmann Pub, 2005.

[13] T. Karagiannis, K. Papagiannaki, and M. Faloutsos,"Blinc: multilevel traffic classification in the dark," ACM SIGCOMM Computer Communication Review, vol. 35, no. 4, p. 240, 2005.

[14] S. Zander, T. Nguyen, and G. Armitage, "Automated traffic classification and application identification using machine learning," 2005.

[15] L. Bernaille, R. Teixeira, I. Akodkenou, A. Soule, and K. Salamatian, "Traffic classification on the fly," ACM SIGCOMM Computer Communication Review, vol. 36, no. 2, p. 26, 2006.