Computer Networks 119 (2017) 102–111

Contents lists available at ScienceDirect

Computer Networks

journal homepage: www.elsevier.com/locate/comnet

Novel feature selection and classification of Internet video traffic based on a hierarchical scheme

Yu-ning Dong[a,*], Jia-jie Zhao[a], Jiong Jin[b]

[a] College of Telecommunications & Information Engineering, Nanjing University of Posts and Telecommunications, Nanjing 210003, China
[b] School of Software and Electrical Engineering, Swinburne University of Technology, Melbourne, VIC 3122, Australia

ARTICLE INFO

Article history:
Received 4 October 2016
Revised 15 March 2017
Accepted 24 March 2017
Available online 29 March 2017

Keywords:
Statistical features
QoS
Video traffic classification
k-Nearest Neighbor classification

ABSTRACT

Accurate traffic classification is critical for efficient network management and resources utilization. Different video traffics have different QoS (Quality of Service) requirements. To provide Internet video services with better QoS support, a fine grained classification scheme for network video traffic is proposed in this paper. Through extensive statistical analysis of typical video traffic flows with a consistency-based method, several new flow statistical features are extracted. They are found to be more effective in discriminating different video traffics, especially from the QoS perspective, than commonly used features available in the literature. A hierarchical *k*-Nearest Neighbor (*k*NN) classification algorithm is then developed based on the combinations of these statistical features. Experiments are performed to evaluate the effectiveness of the proposed method on a large scale real network video traffic data. The experimental results show that the proposed method outperforms existing methods applying commonly used flow statistical features.

1. Introduction

In the recent years, Internet video services are widely used for the rapid development of network and multimedia technologies. According to Cisco's forecast report [1], the proportion of Internet video traffic in 2020 will reach 82% of consumer Internet traffic where consumer includes fixed IP traffic generated by households, university populations, and Internet cafes. Meanwhile, a variety of new applications and diverse protocols make the network environment extremely complex, which undoubtedly causes a series of problems such as efficient network management and QoS (Quality of Service) guarantees for multimedia services. In order to cope with these challenges from the perspective of Internet Service Providers (ISP) and network regulators, accurate classification of network traffic is believed to be the key [2]. For different traffic types, operators can allocate network resources according to their QoS requirements.

This work is motivated by the fact that different video applications may require different QoS support and network resources, while video flows belonging to the same category usually have similar QoS requirements. The correct identification of video flows can thus help ISPs to understand what level of QoS they need and make appropriate resource allocation for them in order to improve the end user's quality of experience. The paper presents an Internet video traffic classification scheme based on hierarchical *k*-Nearest Neighbor (*k*NN) method. This scheme first identifies the most effective QoS related discriminative features and feature combinations by applying statistical analysis and data mining techniques to the acquired video data; then classifies the video flows with a hierarchical classification algorithm using the above features or features combination. To the best of our knowledge, this is the first work that attempts to tackle the video service classification issue with finer granularity from a QoS perspective.

Part of our work has been reported in [3]. However, this paper extends the work of [3] substantially in several aspects. Our main contributions include: 1) a consistency-based feature analysis and selection method is presented to systematically find some new and effective features for video traffic classification; 2) a new hierarchical *k*NN classification scheme is proposed which uses the feature combinations. We also give out more detailed descriptions of flow features analysis in this work. Experimental results show that this scheme can achieve better classification accuracy than existing methods using commonly statistical features.

The rest of the paper is structured as follows. Related works are reviewed in Section 2. Section 3 presents the consistency-based method of feature analysis and selection. Section 4 gives the description of hierarchical *k*NN classification algorithm. Section 5 reports the experiment results. Finally, the paper concludes in Section 6.

[*] Corresponding author.
E-mail addresses: dongyn@njupt.edu.cn (Y.-n. Dong), fzzjj2008@126.com (J.-j. Zhao), jiongjin@swin.edu.au (J. Jin).

http://dx.doi.org/10.1016/j.comnet.2017.03.019
1389-1286/© 2017 Elsevier B.V. All rights reserved.

2. Related work

In the last decade, there has been extensive research on exploring traffic classification method [4]. Here we first review relevant works on ML (machine learning) methods for traffic classification, then focus on related researches on Internet video traffic.

2.1. Internet traffic classification

The relevant research on network traffic classification mainly focuses on ML methods based on flow statistical characteristics [5], including supervised and unsupervised methods.

2.1.1. Supervised methods

The supervised classification methods are provided with a collection of traffic datasets and their pre-identified classes. Zhang et al. [6] proposed a Naïve Bayes based classification scheme using feature discretization and flow correlation. A large scale real-world network dataset with different protocols was carried out in their work. Neural networks have a built-in ability to modify their synaptic connections and weights to adapt to the surrounding environment. Their attributes of non- linearity and adaptability are especially desirable for P2P traffic identification [7]. Jaiswal et al. [8] developed a reduced statistical feature dataset and compared six ML algorithms for traffic classification. Various internet applications (VoIP, Multimedia Streaming, bulk data transfer, Interactive traffic, Email services, WWW traffic and Database traffic) are used in their work. Du et al. [9] put forward a method of P2P traffic identification based on support vector machines (SVM). The method could effectively detect the P2P traffic network flows with three statistical characteristics.

*k*NN classifier is a simple yet often effective supervised method, which includes many advantages (as given in 4.1). A large amount of literature has been reported on researches with *k*NN. Zhang et al. [10] proposed three improved Nearest Neighbor algorithms by incorporating correlated information into the classification process. Their experiments are carried out on two real-world traffic data sets. Patwary et al. [11] presented and implemented PANDA, a parallel and distributed kd-tree based KNN algorithms. The results show that their methods are more suitable for state-of-the-art Big Data analytics problems. Silas et al. [12] designed and implemented a real time flow-based network traffic classification system (NTCS). Their identified application traces include: Www, Https, Ftp, Xvttp and Isakmp Protocol. The modules of the system are built as concurrent processes, which are more effective on Internet traffic monitoring. They use some machine learning methods, such as *k*NN, C4.5 Decision Tree and AdaBoost, to validate the reliability of the approach.

2.1.2. Unsupervised methods

The Unsupervised classification method is essentially a synonym for clustering. Clustering techniques have been applied in the context of Internet traffic analysis for a long time. Zhang et al. [13] adopted a semi-supervised ML technique to effectively discriminate zero-day application. Liu et al. [7] applied FCM clustering to classify P2P application. Their work aimed at reducing the computational complexity of FCM (Fuzzy C-Means) while keeping the clustering accurate. Zhang et al. [14] proposed an encrypted traffic classification scheme based on improved *K*-Means that helps reduce the impact of random initial clustering centers. Their dataset sampled from 5 classes: Skype, QQ, SSH, SSL, MSN.

2.2. Internet video traffic classification

Previous works mostly classified video traffic into one or two classes with coarse granularity. For example, Mu et al. [15] proposed a parallelized network traffic classification scheme using hidden Markov model, where they divided video services into conversational and streaming videos; Gonçalves et al. [16] merely distinguished peer-to-peer (P2P) video traffics. Nguyen et al. [17] make an identification of first-person-shooter online game and VoIP traffic by Naive Bayes and C4.5 Decision Tree ML algorithms. They achieved IP traffic classification by using statistics derived from sub-flows–a small number of packets taken at any point in a flow's lifetime. Claypool et al. [18] proposed a scenario to analyze the network performance of the OnLive thin client game system. Their scenario verifies the differences among traditional network game, thin client game, pre-record video and on live video in terms of bit rate, packet size and inter-packet time. They found that different video streams have different features. Datta et al. [19] presented a case study of Google Hangouts (a semi peer-to-peer application). They used application semantics to identify a set of features and three conventional classification to assess the performance. Wang et al. [20] find two features (downstream/upstream bandwidth) appropriate to classify Internet video traffics, and they propose a modified K-SVD classification algorithm to get QoS classes. Zink et al. [21] proved that trace statistics is relatively stable over short time period while long term trends can be observed. They also showed that P2P paradigm can reduce network video traffic significantly and allow for faster access to video clips.

Nevertheless, our work is substantially different from all the previous ones. Because most of them are either for a particular type of traffic, or just to emphasize on the improvement of algorithms without considering the key problem, that is, how to mine meaningful features from the original flow to enhance the performance [2]. This paper will thus be dedicated to find useful statistical characteristics that can better distinguish different types of video traffic flows.

Table 1
Dataset of video applications.

Application	Traffic class	Volume
ASD	Asymmetric standard definition videos	0.56 GB
AHD	Asymmetric high definition videos	1.23 GB
HTTP-download	HTTP-download videos	2.95 GB
QQ	Interactive video communication class	1.19 GB
Xunlei	P2P video data sharing	4.48 GB
Sopcast	Network live TV	2.62 GB

3. Dataset and feature selection

We have adopted *k*NN algorithm to classify six types (as given in 3.1) of typical video traffics. The key and novel idea we leverage is to select some novel features in a systematic way and build a hierarchical scheme to classify flows from our data acquisition. In the following we further describe our data acquisition and features we selected.

3.1. Data acquisition

We captured real-world video flow data using WireShark[1] in NUPT campus network environment in different time (morning, afternoon and evening) from October 2013 to July 2014. In our study, a flow sample refers to sequences of packets captured in 30 min during the flow's life time of a video application. We grabbed 60 video flows for each application, namely 360 flows in total. Total size of captured data is 13.03GB (see Table 1).

The captured flow trace data contains five columns: the packet arrival time, source and destination IP addresses, protocol and

[1] http://wiki.wireshark.org/;.

Table 2
Some captured statistical features of video flows.

Feature ID	Feature description	Short name	Gain ratio
1	Ratio of downstream bytes to upstream bytes	RDBUB	0.731
2	Average packet inter-arrival time downstream	APITD	0.721
3	Standard deviations of packet size downstream	SDPSD	0.715
4	Information entropy of packet size downstream	IEPSD	0.706
5	Ratio of downstream packages to upstream packages	RDPUP	0.664
6	Average packet size downstream	APSD	0.655
7	Packet Rate	PR	0.628
8	The number of downstream sub-flows	NDSF	0.567
9	The number of downstream valid IP addresses	NDVIP	0.56
10	Byte Rate	BR	0.55
11	Standard deviations of packet inter-arrival time downstream	SDPITD	0.542
12	Information entropy of package arrival interval	IEPAI	0.541

packet size (bytes). Then we process the raw data stream and calculate statistical features of different network video flows. After extensive analysis on video flow data with the consistency-based method, we select a number of QoS related features.

In this paper we focus on six different kinds of Internet video traffics: asymmetric standard definition (SD)/high definition (HD) videos (e.g., Youku/YouTube[2] SD/HD online video play), HTTP-download video data, interactive video communication class (e.g., QQ/Skype[3] video chat), P2P video data sharing (e.g., Xunlei[4] video download), network live TV (e.g., Sopcast[5] live TV broadcasting). Table 1 gives detailed information of the six typical applications. In the following sections and experiments, we will use these typical applications to analyze the features of each traffic class. Besides, the six applications can be further divided into two parts: ASD, AHD and HTTP-download applications are represented as the asymmetric traffic type, while QQ, Xunlei and Sopcast applications are regarded as the symmetric traffic type.

3.2. Features extraction

In the experiment we computed more than 40 statistical features from the original data. The data is divided into downstream and upstream parts. Downstream data refers to the data downloaded to local IP, while upstream data refers to the data uploaded from local IP. Previous works [2] have shown that downstream data carry more information than upstream data. Hence, our analysis mainly focuses on the downstream data. In addition, we adopt information gain ratio [22] to measure the importance of each statistical feature. Table 2 shows some relevant features, which are ranked in descending order according to information gain ratio. To save space, we only list the features whose information gain ratio is greater than 0.5.

3.3. Consistency-based feature selection

Consistency-based feature selection [23] algorithm, as one of the filter approaches, can evaluate the pros and cons of a subset. This method focuses on an inconsistency metric according to which a feature subset is said inconsistent if there exist at least two instances such that they have same values of features but different class labels [23]. The level of consistency of a subset can be measured by an inconsistency rate, which denotes as the ratio of the number of inconsistent samples to the number of all samples. If inconsistency rate of subset S is low, the subset is said to be consistent. In other words, it is believed that the subset S is a good subset.

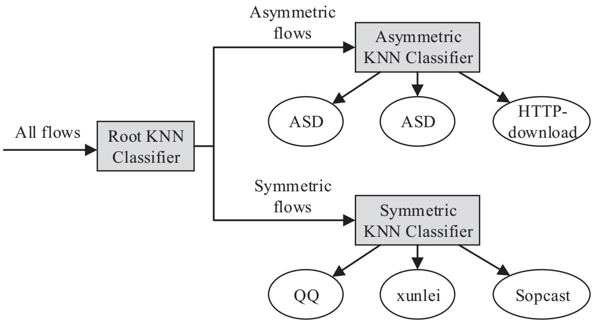


Fig. 1. Hierarchical *k*NN classification scheme for Internet video traffic classification.

In this paper, we use a modified version of this method in combination with a layered *k*NN classifier to evaluate the goodness of a feature subset (refer to Fig. 1). We only consider the subset which contains two features in order to minimize the number of features and to reduce time cost in exhaustive search. Some two-element subsets used for the three classifiers are shown in Table 3, ordered according to their inconsistency rates.

We only list the 5 least inconsistency rate among all feature combinations. If two feature subsets have the same value of inconsistency rate, we give priority to the subset whose features contains higher information gain ratio. In other words, more important feature subset is preferentially considered. For example, the tuple {1, 4} means the feature combination of RDBUB and IEPSD, which is better than {4, 11} representing the combination of IEPSD and SDPITD. As a result, in the root classifier, we select the feature combination of RDBUB and IEPSD; the combination of RDBUB and APITD is selected as the asymmetric classifier; the feature combination of RDBUB and IEPSD are adopted in the symmetric classifier.

3.4. Feature description

After the procedure of consistency-based feature selection, we finally select four features: ratio of downstream bytes to upstream bytes (RDBUB, 1), information entropy of packet size downstream (IEPSD, 4), the number of downstream sub-flows (NDSF, 8), average packet inter-arrival time downstream (APITD, 2) and their combinations to classify the typical network video applications. The detailed descriptions of the four features mentioned above are as follows:

(1) *RDBUB*
RDBUB refers to the ratio of total received downstream data bytes to the total upstream data bytes after removal of overhead packets (e.g., control packet). This feature is generally not sensitive to the changes of network condition and relatively stable. It can help distinguish between symmetric (e.g.,

[2] http://www.youtube.com/;.
[3] http://skype.gmw.cn/;.
[4] http://dl.xunlei.com/;.
[5] http://www.easetuner.com/.

Table 3
Inconsistency rates of each classifier (refer to Fig. 1).

Classifier	Description	Inconsistency rates of subset in ascending order		
Root	Distinguish symmetric and asymmetric flows	{1, 4}/0.0	{4, 11}/0.0	{2, 4}/0.0028
		{3, 6}/0.0028	{4, 6}/0.0028	…
Asymmetric	Distinguish ASD, AHD and HTTP-download	{1, 2}/0.0056	{2, 3}/0.0111	{2, 4}/0.0111
		{2, 6}/0.0111	{2, 8}/0.0111	…
Symmetric	Distinguish QQ, Xunlei and Sopcast	{4, 8}/0.0	{4, 9}/0.0	{4, 10}/0.0
		{4, 12}/0.0	{1, 2}/0.0056	…

P2P streaming) and asymmetric video traffic combined with IEPSD or NDSF as described later.

(2) *IEPSD*
IEPSD indicates information entropy of packet size downstream, which is defined in Eq. (1). This feature can measure the uniformity of packet size distribution. The larger the value is, the more uniform the packet size distribution is. Our experiments show that this feature is effective to distinguish among QQ, Xunlei and Sopcast.

$$E = -\sum p(x_i)\log_2 p(x_i) \tag{1}$$

where E denotes the IEPSD, x_i is the ith packet size and $p(x_i)$ the probability density function (PDF) of the ith packet size x_i .

(3) *NDSF*
NDSF is the number of downstream sub-flows fragments. A sub-flow fragment refers to continuous packets from a fixed source IP. Our study has shown that there are many sub-flow fragments in symmetric traffics, which implies that this feature is available to distinguish these traffics. It is found in our experiment that one can get a better classification result using logarithmic form of NDSF (or log-NDSF for short) than using its original form.

(4) *APITD*
APITD is for average packet inter-arrival time of packets. Clearly, this feature is distinct in different video traffics. Three asymmetric traffics can be better distinguished in our experiments. However, for symmetric traffic, this feature may bring some poor results.

4. Hierarchical *k*NN classifier

4.1. kNN classifier

K-nearest neighbor (*k*NN) [11] is a supervised learning algorithm for classifying objects that has been used in many fields, such as data mining, pattern recognition and many others. It classifies a new object based on the *k* nearest neighbors from the training space. The new object is assigned to the class most common among its nearest neighbors.

*k*NN has some advantages: 1) It is the simplest of all machine learning algorithms; 2) It can be easily implemented, because each data point only depends on the *k* nearest neighbors; 3) The classification results of *k*NN are analytically tractable. However, *k*NN may obtain low accuracy results with irrelevant features.

4.2. Hierarchical classification scheme

With the increased number of applications involved, the performance of classification algorithms will naturally deteriorate [24]. In other words, it is often hard to distinguish all kinds of traffics at a time by using a large number of features. For example, it is rather challenging to split Sopcast from all other flows directly with the selected features. However, it will be relatively easier to split Sopcast with specific statistical features that make such application differentiable from symmetric video traffic. Therefore, the paper employs hierarchical classification scheme which allows each sub-classifier to deal with a limited subset of video flows that are easier to distinguish them.

Fig. 1 gives the hierarchical *k*NN classification scheme developed in this paper. Gray square nodes are sub-classifiers and white elliptical nodes represent the individual video services. Our scheme contains two layers. In the first layer, we use root *k*NN classifier to divide all video flows into symmetric and asymmetric flows. The feature combination of RDBUB and IEPSD is used in this stage. Then in the second layer, asymmetric video flows are further split into three classes: ASD, AHD and HTTP-download, and symmetric video flows are split into finer grained classes: QQ, Xunlei and Sopcast.

5. Experiments

In this section, we adopted the hierarchical *k*NN scheme to classify the six kinds of typical video applications. The steps of the classification scheme include: 1) process the original data flow to obtain QoS related statistical features; 2) use different statistical features or feature combinations at different levels to identify traffic data in a coarser grain; 3) obtain final classification results after multi-level classification scheme.

5.1. Experiments setup

We exploit the feature combinations together with the proposed hierarchical *k*NN algorithm, and verify the performance of our method in contrast to the method of [10]. We grabbed 360 Internet video flows, extracted corresponding features, and used layered *k*NN algorithm, where $k = 3$, since a better overall accuracy can be obtained. 50% of the flows are randomly chosen as the training set and the rest for the test set. The final results are the average of 20 runs.

Precision and recall are two commonly used performance metrics of classification results [25], which reflect the completeness and correctness of the classification result. They are defined as below:

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

where, True Positives (TP) is the number of class j's samples identified as class j; False Negatives (FN) is the number of class j's samples identified as other class; False Positives (FP) is the number of other class's samples identified as class j; True Negatives (TN) is the number of a given class's sample identified as another class.

Precision and recall rates reflect the classification performance from two different aspects; while F-measure is the weighted harmonic mean of precision and recall, defined as below:

$$F - measure = \frac{2 * Precision * Recall}{Precision + Recall} \tag{4}$$

Table 4
Considered features in some related work.

Approach	Selected features	Number of used features
Proposed method	RDBUB, IEPSD, log-NDSF, APITD	4
Zhang et al. [6]	Packets, Bytes, Packet Size, Inter-Packet Time	6
Liu et al. [7]	Traffic flow and packet statistics	15
Jaiswal et al. [8]	Packets/sec, Mean IP packet length, Mean IP payload length, Bytes/Flow, Flow duration, Mean IAT, Bytes/sec, SD of IP packet length, SD of IAT, SD of IP payload length	10
Du et al. [9]	Packet, data flow and connection characteristic	Not mentioned
Zhang et al. [10]	Packets, Bytes, Packet Size, Inter-Packet Time	7 or 6 features
Pereira et al. [12]	Elapsed time between the first and last packets, number of packets, number of bytes, the number of all packets with at least a byte of TCP data payload, the number of all packets seen with the PUSH bit set in the TCP header, and the median and the variance of the number of bytes in IP packet	14
Zhang et al. [13]	Packets, Bytes, Packet Size, Inter-Packet Time	9
Zhang et al. [14]	Packet length, inter arrival time, duration of the flow, total packets, total volume, protocol	More than 20
Mu et al. [15]	Inter Packet Time(IPT) and Payload Size(PS)	2
Wang et al. [19]	Downstream/upstream rates	2
Luigi et al. [24]	Layer-4 features provided by Tstat	At most 35 for each classifier

Overall accuracy reflects the overall classification accuracy of the system, defined as follows,

$$Overall\,accuracy = \frac{\sum_{i=1}^{c} TP_i}{\sum_{i=1}^{c} TP_i + FN_i} \tag{5}$$

This work runs on the computer with 2.0 GHz Intel Pentium CPU and 4.0 GB of RAM. All of features are extracted using shell script in Linux environment. The time cost in each experiment represents time consumption for each classification algorithm, which is calculated using "Java" language. Other metrics like precision, recall, F-measure and overall accuracy are obtained from WEKA platform.

5.2. Experiment results

In this section, we use the above metrics to validate the effectiveness of selected feature combinations with the hierarchical scheme. The model of hierarchical scheme are shown in Fig. 1. In order to do the comparison reasonably, we studied selected features in other related work, which is shown in Table 4.

Most of approaches adopt simple features to finish their classification work, such as average/standard deviations of packet size/packet inter-arrival time. Some approaches [12,23] further have a consideration of TCP related features. However, these statistical features are not applicable to video flows because UDP or HTTP protocols are also contained in these videos. In addition, payload related features are not considered in our paper because these features may inspect contents of packets and face challenges of privacy policies and regulations [5]. After removing TCP/payload related features, we found that most of literatures use four commonly used statistical features: average packet size downstream (APSD), average packet inter-arrival time downstream (APITD), standard deviations of packet size downstream (SDPSD) and packet inter-arrival time downstream (SDPITD).

Method [10] proposed three improved Nearest Neighbor algorithms, where AVG-NN method obtains the best performance. Because AVG-NN algorithm is not for classifying Internet video traffic, we conduct the experiments of AVG-NN by using our dataset and their commonly used features (APSD, APITD, SDPSD and SDPITD) for a fair comparison. We refer to this case as “AVG-NN [10]” classifier. “Proposed” denotes our proposed hierarchical *k*NN algorithm in combination with the feature combinations mentioned in Section 3.3. Fig. 2 shows the precision, recall and F-measure of “AVG-NN” and the proposed method.

When $k = 3$, the overall accuracy of AVG-NN [10] is 0.8711, while that of the proposed method is 0.9897. It can be seen from Fig. 2 that our method has obvious advantages for QQ, Xunlei and Sopcast. Moreover, we enhanced the F-measure of ASD and AHD by 8.3% and 8.6% respectively in comparison with [10]. The performance of our method is slightly better for HTTP-download and the F-measure achieves more than 98%. From an overall perspective, the classification performance of our method is better in comparison with [10].

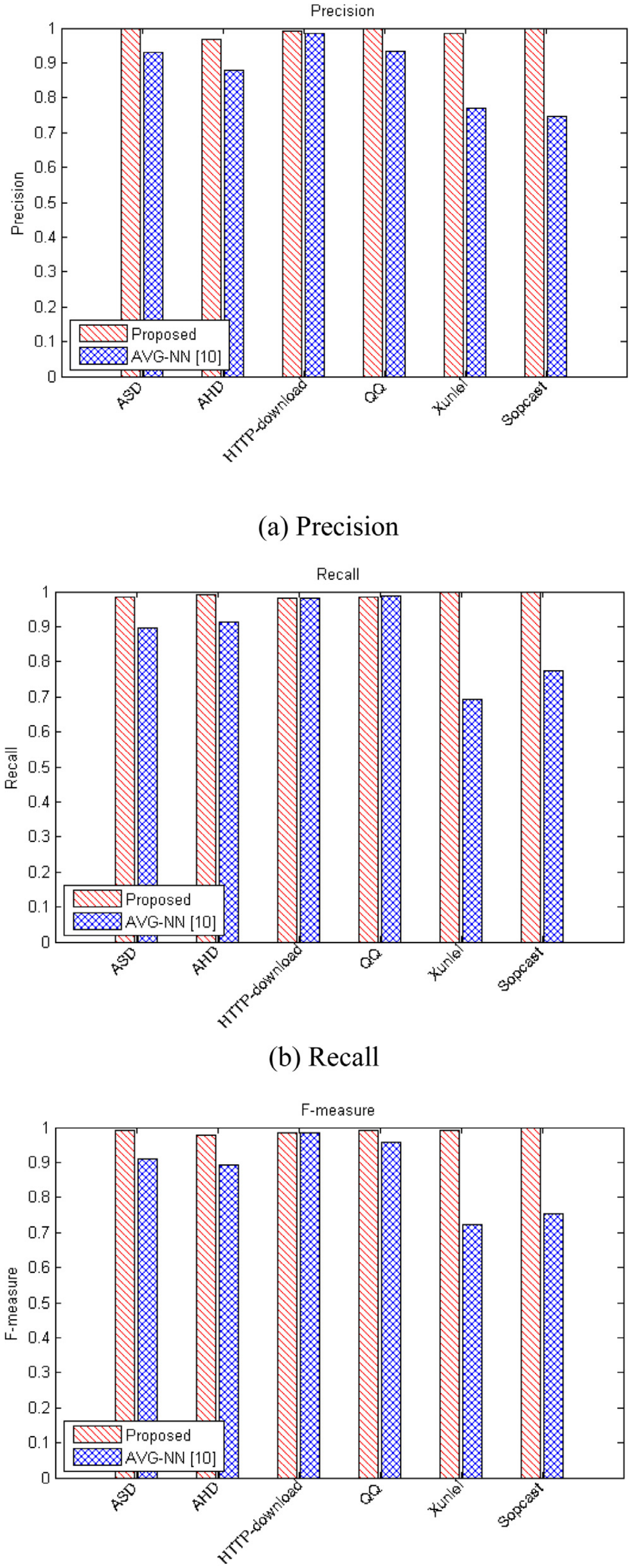
Reference [8] lists 6 well-known algorithms in machine learning: Decision Tree (C4.5), Naïve Bayes, Random Forest, Neural Network (MLP, RBF) and SVM. The experiments are repeated in using these algorithms with our selected features and the features used in [10]. For the hierarchical algorithm (Hier-*k*NN and Hier-SVM), features used in each layer are mentioned in Section 3.3; whereas other algorithms classify video traffic directly with two different sets of features.

Fig. 3 lists the overall accuracy of each algorithm, and Table 5 gives time cost of each algorithm. For the algorithm of C4.5 decision tree, it takes 187 ms to finish the classify process, yet the overall accuracy is only close to 0.95. The proposed feature selection method can raise overall accuracy rate of 1.6% comparing to the features used in [10]. Naive Bayes is suitable to distinguish the small dataset. Note that our features lift its accuracy rate to 99.39% from 96.97%. For Random Forest, neural network (RBF) and SVM, although all of overall accuracy of these algorithms are over 0.97, it takes longer time to perform the training phase. Note that the proposed features can improve the accuracy rate by 1.1%, 1.5% and 4.2% respectively. As for neural network (MLP), all of the overall accuracy is over 0.97, no matter what feature combinations are adopted. In general, we can see that most algorithms reach to a higher overall accuracy when proposed features are used rather than those of [10], which further illustrates that proposed feature combinations are more effective.

In order to study the effect on hierarchical algorithm, we do another experiment on SVM. We find that although overall accuracy of hierarchical SVM is slightly lower than that of non-hierarchical SVM, the running time (especially in the training phase) is lower than that of non-hierarchical SVM. This is because increased feature dimension results in increased complexity of modeling work in training phase. In other words, hierarchical algorithm decreases execution time of SVM. Note that our method can increase overall accuracy rate by 18.8% for hierarchical SVM.

5.3. Experiment analysis

According to Section 5.2, higher performance can be achieved when using features combination of proposed method rather than



(a) Precision

(b) Recall

(c) F-measure

Fig. 2. Classification results of different methods.

those of [10]. We firstly analyze four proposed features (RDBUB, IEPSD, log-NDSF and APITD) mentioned in Section 3.4.

As shown in Fig. 4, each sample point represents a traffic flow, the x-axis and y-axis denote two different features RDBUB and IEPSD respectively.

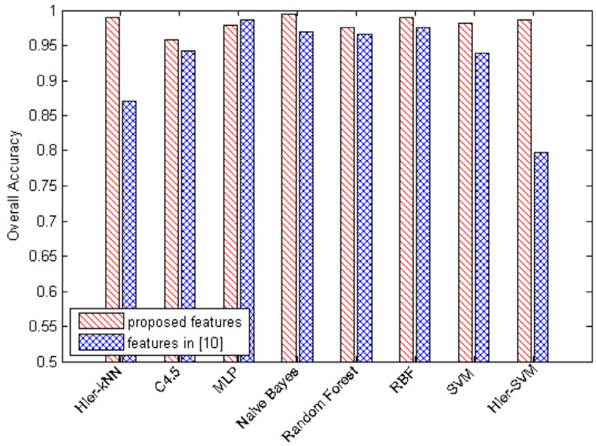


Fig. 3. Overall accuracy of different algorithms using different features.

Table 5
Time cost of each algorithm.

Algorithm	Training time [ms]	Test time [ms]	Total time [ms]
Proposed	123	253	376
AVG-NN [10]	51	155	206
C4.5	175	12	187
Naïve Bayes	59	108	167
Random Forest	1952	159	2111
MLP	6619	17	6636
RBF	1609	93	1702
SVM	1726	44	1770
Hier-SVM	1459	90	1549

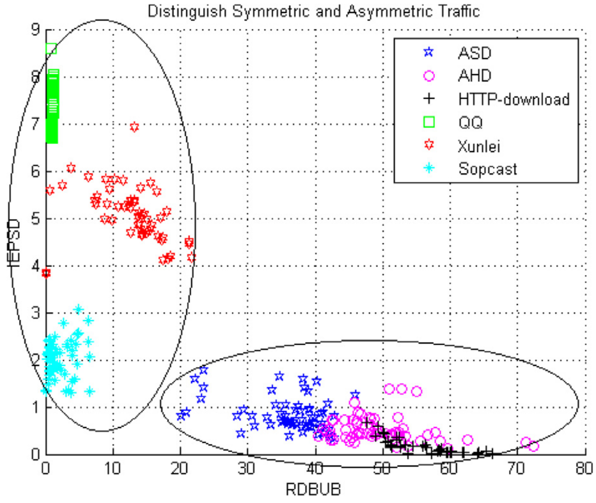


Fig. 4. Distinguish between symmetric and asymmetric video applications.

For the feature RDBUB, symmetric traffic is significantly less than that of asymmetric traffic. For the features IEPSD, QQ and Xunlei are larger than those of symmetric traffic, while the difference between Sopcast and symmetric traffic is not obvious. It is easy to divide these samples into two parts with two circles (see Fig. 4). Note that the feature combination of RDBUB and IEPSD can distinguish between symmetric and asymmetric applications in the two-dimensional space, but cannot separate these two categories using any single feature. This implies that the selected features combination is often more effective.

To further differentiate the following three applications (QQ, Xunlei and Sopcast), we select the combination of IEPSD and RDBUB, as shown in Fig. 5. It can be seen that there is overlap be-

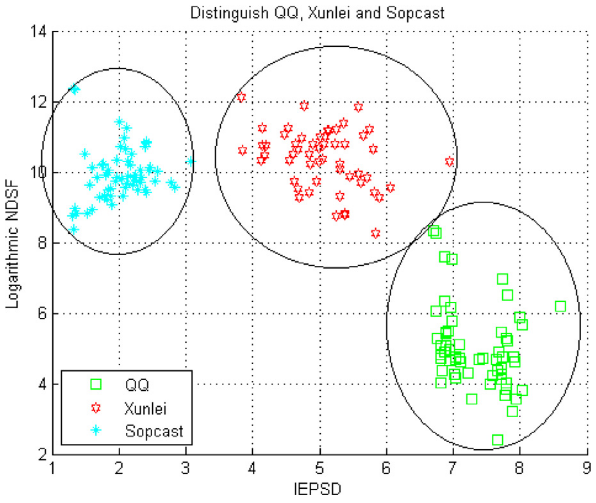


Fig. 5. Distinguish among QQ, Xunlei and Sopcast.

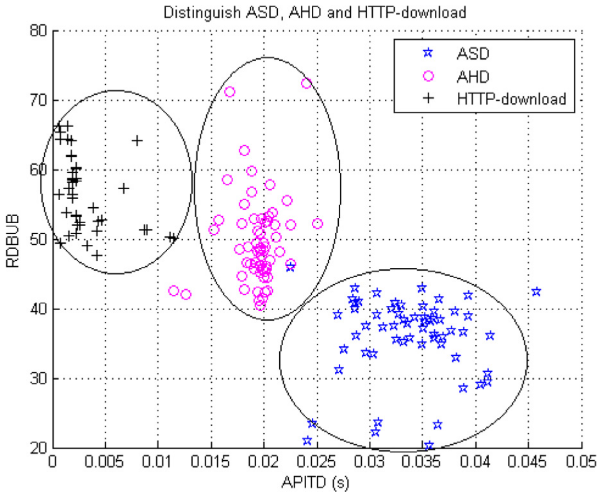


Fig. 6. Distinguish among ASD, AHD and HTTP-download.

tween QQ and Xunlei in the information entropy of packet size. However, as a whole, the two features are effective in distinguishing these three traffics (QQ, Xunlei and Sopcast) in Fig. 5.

For asymmetric applications, we select RDBUB and APITD as the feature combination, as shown in Fig. 6. We found that the three applications show a great difference in the feature APITD, namely, the feature of average inter-arrival time of packets. To investigate the causes, we further measure the feature for ASD, AHD and HTTP-download respectively. For the asymmetric SD (ASD) from the feature space available, the inter-arrival time of packets is mainly distributed between 0.025 s and 0.04 s, while the result of the asymmetric HD (AHD) is about 0.02 s. HTTP-download is a traditional application type, similar to FTP. So the inter-arrival time can be very small, for around 0.005 s. Unfortunately, although there exists an obvious difference between HTTP-download and asymmetric video flows, a few samples may be confused with this feature. In Fig. 6, we can clearly see that two ASD samples will be misclassified as HTTP-download flow. Even so, it seems to be acceptable, because we can generally divide most of the flows correctly by using this feature, especially for ASD and AHD, where often exists a heavy overlap between them.

In order to carry out a deep discussion of classified result, it is necessary to analyze the confusion matrix. Fig. 7 gives the confusion matrix of our proposed method using four features: RDBUB, IEPSD, log-NDSF and APITD. Note that true positive rate can reach more than 98% for each application, no matter on asymmetric video applications or symmetric videos. Therefore, the four features in this paper are suitable for the classification of 6 kinds of video traffics.

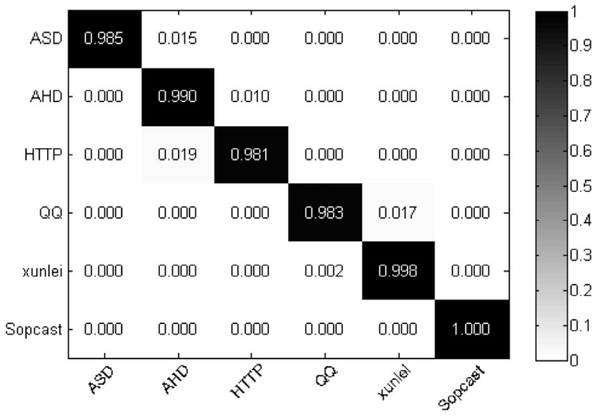


Fig. 7. Confusion matrix of the proposed method using four features: RDBUB, IEPSD, log-NDSF and APITD.

Table 6
Comparison of different pairwise scheme.

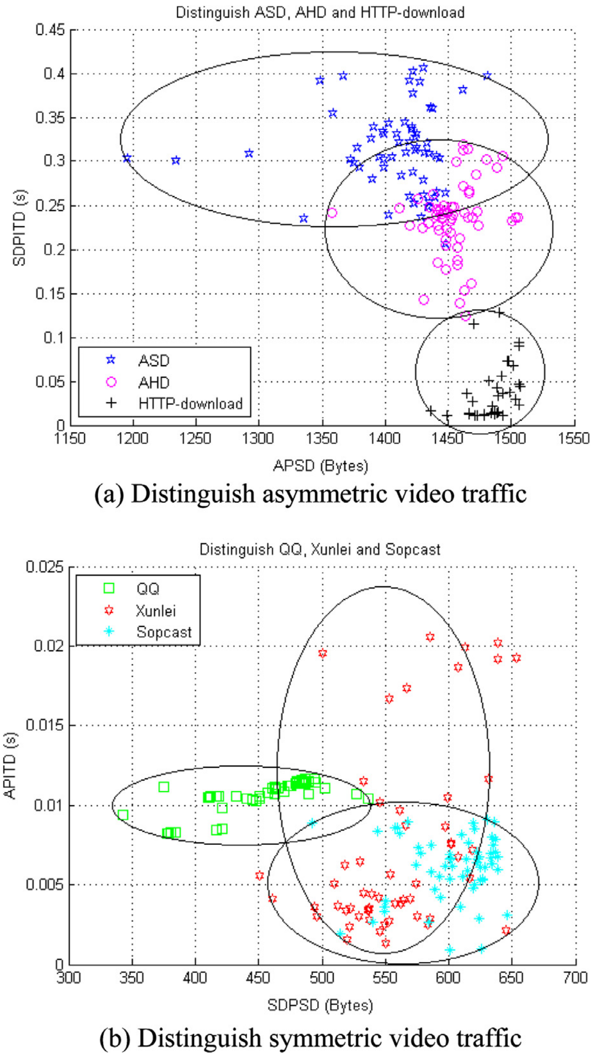
Number of features in each classifier	Root	Asymmetric	Symmetric	Number of used features	Search time [ms]
One-element	{1}	{2}	{4}	3	25
Two-element	{1, 4}	{1, 2}	{4, 8}	4	79
Three-element	{1, 2, 4}	{1, 3, 11}	{2, 4, 6}	6	205
Four-element	{1, 2, 3, 4}	{1, 2, 3, 11}	{1, 4, 6, 10}	7	353
Unlimited	{1, 4}	{1, 3, 11}	{4, 6}	5	2154
Non-hierarchical	{1,3,4,11}	–	–	4	722

The sample distributions when using four commonly used features (APSD, APITD, SDPSD and SDPITD) are displayed in Fig. 8. Note that the distribution of Xunlei and Sopcast are quite decentralized, in other words, there is no obvious boundary between Xunlei and Sopcast in Fig. 8(b). Therefore, it is difficult to distinguish among QQ, Xunlei and Sopcast with SDPSD-APITD feature space. As for asymmetric services, it is obvious that our method in Fig. 6 can easily distinguish these three types of video streams compared with those in Fig. 8(a).

We also conduct a further discussion on method [10] using the confusion matrix. We can see that, for asymmetric video applications, corrected classified instances in ASD/AHD are close to 0.9, which is lower than correct rate in Fig. 9. For symmetric videos, true positive rate in xunlei and Sopcast traffics is less than 0.8. It is because a heavy overlap between xunlei and Sopcast may lead to negative outcomes that more instances are misclassified. It is clear that the proposed method achieves higher overall accuracy, which further validates the effectiveness of the proposed method.

5.4. Pairwise scheme

In Section 3.3, we only consider the case of the feature subsets with two elements for each classifier. A detailed analysis will be performed in this section, and the experiment model is shown in Fig. 1. In Table 6, we take into account of six pairwise schemes of feature subsets for each classifier: the hierarchical *k*NN algorithm



(a) Distinguish asymmetric video traffic

(b) Distinguish symmetric video traffic

Fig. 8. Sample distributions of method [10] using four features: APSD, APITD, SDPSD and SDPITD.

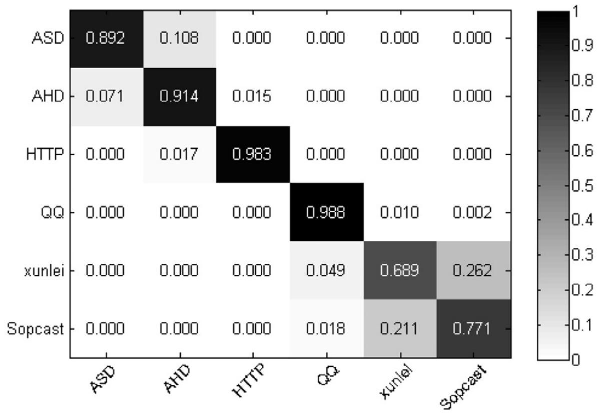


Fig. 9. Confusion matrix of method [10] using four features: APSD, APITD, SDPSD and SDPITD.

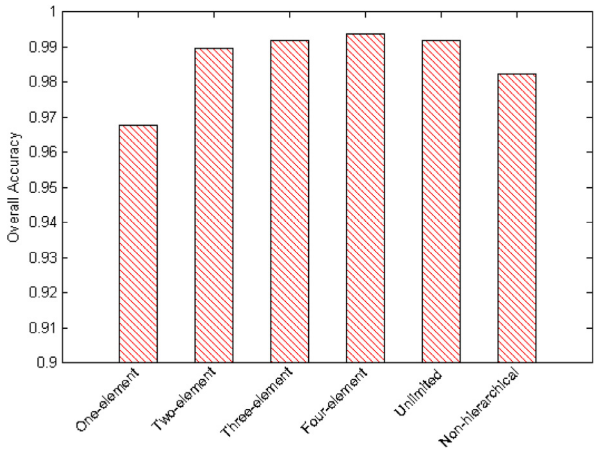


Fig. 10. Overall accuracy of different pairwise scheme.

Table 7
Performances with different numbers of features.

	Precision		**Recall**		**F-measure**	
	Our method	All four features	Our method	All four features	Our method	All four features
ASD	1.000	1.000	0.986	0.988	0.993	0.994
AHD	0.969	0.986	0.990	0.998	0.979	0.992
HTTP	0.990	0.964	0.981	1.000	0.985	0.982
QQ	0.998	0.979	0.983	0.979	0.991	0.979
Xunlei	0.983	0.967	0.998	0.935	0.990	0.951
Sopcast	1.000	1.000	1.000	0.968	1.000	0.984

with one-element feature subsets, with two-element feature subsets, with three-element feature subsets, with four-element feature subsets, without limitation and non-hierarchical *k*NN algorithm. "Root" represents that feature subsets are selected by root *k*NN classifier. "Asymmetric" and "Symmetric" stand for asymmetric *k*NN classifier and symmetric *k*NN classifier respectively. For example, {1, 2} is short for the selected feature combination of {RDBUB, IEPSD} (refer to Table 2). Total selected features and search time of each scheme are also listed out in Table 6. We can draw the following conclusions: 1) When the number of features in each classifier is raised, total features and the search time will also increase; 2) If we do not consider the limit of pairwise (scheme 5 and 6), the search time will increase significantly.

Fig. 10 gives the overall accuracy of different pairwise schemes. For the case of only one feature in each classifier, low accuracy rate of 0.9677 will be obtained. For the case of feature subsets contains two or more elements, overall accuracy is close to 0.99. Note that when each classifier contains two features, overall accuracy has reached to 0.9897. The expansion of feature in each classifier will only increase the complexity of scheme, while have a little effect on overall accuracy. After the above analysis, we find that the two-element scheme can strike a good balance between high accuracy and low search time.

5.5. Effect on amounts of features

One interesting question is whether and how the amount of features will have an effect on experiment results. Therefore, we designed an experiment to test it. The experiment model is the same hierarchical *k*NN scheme (refer to Fig. 1), but the amount of features are different in each classifier node. Originally, each classifier use two selected features as mentioned before, then we compare the case of using all four features for each classifier. Tables 7 and 8 report the performance differences between them. It is shown that a small gap of classification accuracy exists between the two cases. However, in the latter case, each classifier will consume more time.

Table 8
Time consumption of the classifiers using different numbers of features.

	Proposed method [ms]	All four features [ms]
Root	172	216
Asymmetric	99	134
Symmetric	105	127
Total	376	477

6. Conclusion

Network video traffic classification is essential for network management and QoS support. In this paper, we proposed a novel network video traffic classification scheme based on a hierarchical *k*NN classifier. In this scheme, we find some new and effective statistical features by consistency-based analysis and data mining, and validate the effectiveness of these feature combinations in video traffic classification. Through this hierarchical scheme, finer grained classification for Internet video traffic can be achieved. Under a real network environment, the effectiveness of proposed classification scheme with our feature combinations is validated in experiments compared with existing method. Although the hierarchical scheme incurs a certain degree of complexity, it seems to be acceptable if considering the enhancement of overall accuracy metrics on classification results.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (No.61271233), and the HIRP program of Huawei Technology Co. Ltd.

References

[1] Cisco Visual Networking, Index: forecast and methodology, 2015–2020, 2016. June 6 June 6 http://www.cisco.com/c/en/us/solutions/collateral/service-provider/ip-ngn-ip-next-generation-network/white_paper_c11-520862.html .

[2] A Anand, G Veciana, Invited paper: context-aware schedulers: realizing quality of service/experience trade-offs for heterogeneous traffic mixes, in: 2016 IEEE 14th International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt), 2016, pp. 1–8.

[3] Y N Dong, L T Yao, H X Shi, Fine grained classification of internet video traffics, in: 2015 21st Asia-Pacific Conference on Communications (APCC), Kyoto, 2015, pp. 580–584.

[4] A Dainotti, A Pescape, K Claffy, Issues and future directions in traffic classification, IEEE. Network. 26 (1) (2012) 35–40.

[5] N Al Khater, E Overill R, Network traffic classification techniques and challenges, in: Tenth International Conference on Digital Information Management, IEEE, 2015, pp. 43–48.

[6] J Zhang, C Chen, Y Xiang, et al., Classification of correlated internet traffic flows, in: Trust, Security and Privacy in Computing and Communications (TrustCom), 2012 IEEE 11th International Conference on 25-27 June 2012, Liverpool, IEEE, 2012, pp. 490–496.

[7] D Liu, C H Lung, P2P traffic identification and optimization using fuzzy c-means clustering, in: 2011 IEEE International Conference on Fuzzy Systems (FUZZ), 27-30 June 2011,Taipei, IEEE, 2011, pp. 2245–2252.

[8] R.C. Jaiswal, S.D. Lokhande, Machine learning based internet traffic recognition with statistical approach, in: Annual IEEE India Conference (INDICON) 13-15 Dec. 2013, Mumbai, 2013, pp. 1–6.

[9] D Jiang, L Tao, P2P traffic identification research based on the SVM, in: Wireless and Optical Communication Conference (WOCC) 16-18 May 2013, Chongqing, 2013 22nd., IEEE, 2013, pp. 683–686.

[10] J Zhang, Y Xiang, Y Wang, et al., Network traffic classification using correlation information, IEEE. Trans. Parallel. Distrib. Syst. 24 (1) (2013) 104–117.

[11] MMA Patwary, et al., PANDA: extreme ecale parallel K-Nearest neighbor on distributed architectures, in: 2016 IEEE International Parallel and Distributed Processing Symposium (IPDPS), 2016, pp. 494–503.

[12] S.S.L. Pereira, J.L.d.C.e. Silva, J.E.B. Maia, NTCS: A real time flow-based network traffic classification system, in: 10th International Conference on Network and Service Management (CNSM) and Workshop, Rio de Janeiro, 2014, pp. 368–371.

[13] J Zhang, X Chen, Y Xiang, L Zhou W, J Wu, Robust network traffic classification, in: ACM Transactions on Networking, 99, IEEE, 2014, p. 1.

[14] M Zhang, H Zhang, B Zhang, et al., Encrypted traffic classification based on an improved clustering algorithm, in: Trustworthy Computing and Services, Springer, Berlin Heidelberg, 2013, pp. 124–131.

[15] X Mu, W Wu, A parallelized network traffic classification based on hidden markov model, in: 2011 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC) 10-12 Oct. 2011, Beijing, IEEE, 2011, pp. 107–112.

[16] GD Gonçalves, Í Cunha, AB Vieira, Predicting the level of cooperation in a peer-to-peer live streaming application, Multimedia. Syst. (2014) 1–20.

[17] T.T.T. Nguyen, G. Armitage, P Branch, S. Zander, Timely and continuous machine-learning-based classification for interactive IP traffic, IEEE/ACM. Trans. Networking. 20 (6) (Dec. 2012) 1880–1894.

[18] M Claypool, D Finkel, A Grant, M Solano, Thin to win? network performance analysis of the onlive thin client game system, in: 2012 11th Annual Workshop on Network and Systems Support for Games (NetGames) 22-23 Nov.2012, Venice, IEEE, 2012, pp. 1–6.

[19] J Datta, N Kataria, N Hubballi, Network traffic classification in encrypted environment: a case study of Google hangout, in: 2015 Twenty First National Conference on Communications (NCC), Mumbai, 2015, pp. 1–6.

[20] ZJ Wang, YN Dong, HX Shi, LY Yang, PP Tang, Internet video traffic classification using QoS features, in: 2016 International Conference on Computing, Networking and Communications (ICNC), Kauai, HI,, 2016, pp. 1–5.

[21] M Zink, K Suh, Y Gu, J Kurose, Characteristics of YouTube network traffic at a campus network–measurements, models, and implications, Comput. Networks. 53 (4) (2009) 501–514.

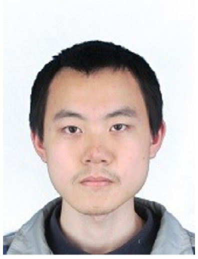
[22] Q Liu, Z Liu, R Wang, et al., Large traffic flows classification method, in: IEEE International Conference on Communications Workshops, IEEE, 2014, pp. 569–574.

[23] ZT Fernando, IS Thaseen, C. Aswani Kumar, Network attacks identification using consistency based feature selection and self organizing maps, in: 2014 First International Conference on Networks & Soft Computing (ICNSC), IEEE, 2014, pp. 162–166.

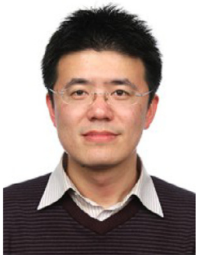
[24] L. Grimaudo, M. Mellia, E. Baralis, Hierarchical learning for fine grained internet traffic classification, in: 2012 8th International Wireless Communications and Mobile Computing Conference (IWCMC),, IEEE, 2012, pp. 463–468.

[25] Y Zhibin, GB Kil, S Kim, Traffic classification based on visualization, in: 2011 IEEE 2nd International Conference on Networked Embedded Systems for Enterprise Applications (NESEA), 8-9 Dec. 2011,Fremantle, WA, IEEE, 2011, pp. 1–6.

Yu-ning Dong received his B.E, M.E degrees from Nanjing University of Posts & Telecommunications (NUPT), Ph.D degree in Electrical Engineering from Southeast University, and M.Phil degree in Computer Science from The Queen's University of Belfast (QUB). He is currently Professor with College of Communications and Information Engineering, NUPT. His research interests include multimedia communications, wireless networking, and network traffic identification.



Jia-jie Zhao received his bachelor in Nanjing University of Posts & Telecommunications (NUPT), China, in 2014. He is currently a Master degree candidate of NUPT. His research interests include network traffic identification, pattern recognition, and multimedia communication.



Jiong Jin (IEEE M'11) received the B.E. degree with First Class Honours in Computer Engineering from Nanyang Technological University, Singapore, in 2006, and Ph.D. degree in Electrical and Electronic Engineering from The University of Melbourne, Australia, in 2011. He is currently a Senior Lecturer in School of Software and Electrical Engineering, Swinburne University of Technology, Australia. His research interests include network design and optimization, nonlinear systems and sliding mode control, networked robotics, Internet of things, cyber-physical systems and applications in smart grids and smart cities.