Video Traffic Attributes for End Host Identification

Kazumasa Oida and Kenichi Yamashita

Abstract—**Internet TV sites often use source IP addresses in received packets to identify users or to retrieve user-related information such as organizations, geographic locations, etc. Users, on the other hand, consult a blacklist to check whether packets are from malicious websites or not. Source IP addresses in received packets, however, do not always agree with addresses of true source hosts due to VPN, NAT, proxy technologies, or malicious attacks. To improve end host identification, this paper proposes using video traffic features, which are the decay rates of the aggregated variance, as a signature of a source or destination host. Experimental results show that if sample decay rates of 100 Internet TV sites are given as a training dataset, 94.5% of the TV sites are correctly identified by the naive Bayes classifier.**

Index Terms—**End host identification, video traffic, decay rate, classifier, information gain**

I. Introduction

Nowadays, there are a large number of Internet TV sites all over the world. Many of the sites provide free of charge TV channels and make profits through advertising. These sites often utilize source IP addresses in received packets to identify users or to retrieve user-related information such as organizations, geographic locations, etc. to improve personalized service and advertisement. Users, on the other hand, may feel the need to verify that each received packet does not come from malicious servers. This is especially true when they are receiving stock quote streams. For this requirement, security software companies update the list of IP addresses with negative reputation to prevent customers from accessing malicious websites.

In the above-mentioned services, each source IP address is related to a host operated by a specific customer, an Internet TV provider, or a cyber-criminal. Therefore, by using the relations, source IP addresses in arriving packets are used to infer persons who sent the packets. However, the relations do not always hold since some hosts may use temporarily assigned addresses. VPN technologies [1], for example, may allocate a different IP address to a remote computer. Furthermore, the source IP address in the packet header may be replaced by proxy servers [2], NAT devices [3], or malicious hosts who are making IP address spoofing attacks [4].

In addition to source IP addresses, this paper proposes using statistical features of video traffic for improving end host identification. Previous works in [5]-[7] indicate that the communication environment, which includes application software, communication protocols, propagation delays, etc., affects variability of Internet traffic over multiple time scales. This result suggests that variability of video traffic should vary when a source or destination host changes. Inspired by this finding, this paper derives a signature of a client or a server from sample variances calculated at many time scales. The biggest advantage of our approach is that there are no message exchanges and no necessary negotiations between two end hosts. As far as we know, our approach is new and currently there are no ongoing similar works.

This paper is organized as follows: Section II introduces the decay rate used for end host identification. Section III calculates sample decay rates of four characteristic stream types (variable-bitrate, congested, sequential and parallel types) to investigate their impacts on the decay rate. Section IV obtains the percentage of correctly classified Internet TV sites under the condition that a training dataset for all TV sites is a priori given. This section also discusses the effect of the number of clients on the percentage. Section V evaluates the effectiveness of decay rates based on information gain and correlation. Finally, Section VI presents the conclusions.

II. Multi-Timescale Variability

A. Variance Plot

We first introduce the variance plot [8] (also called the variance-time plot [9] or the aggregated variance [10]) for a time series $\{X_k\}$, where X_k denotes the number of arriving packets during the k-th time interval of length δ. The m aggregated series of $\{X_k\}_{1\le k\le N}$, $\{X_k^{(m)}\}$, are obtained by dividing $\{X_k\}$ into blocks of length m and averaging the series over each block as

$$X_\ell^{(m)} = \frac{1}{m}\sum_{i=\ell m-m+1}^{\ell m} X_i,\ \ell = 1,2,\ldots,\lfloor N/m \rfloor, \quad (1)$$

where m is a positive integer, N is the size of series $\{X_k\}$, and $\lfloor x \rfloor$ is the largest integer that does not exceed x. The aggregated variance $V^{(m)}$ is the sample variance of $\{X_k^{(m)}\}$; i.e.,

$$V^{(m)} = \frac{1}{\lfloor N/m \rfloor - 1}\sum_{k=1}^{\lfloor N/m \rfloor}(X_k^{(m)} - \bar{X})^2, \quad (2)$$

where $\bar{X} = \frac{1}{N}\sum_{i=1}^{N} X_i$. The variance plot is obtained by plotting $\log_{10} V^{(m)}$ versus $\log_{10} m$ for various aggregation levels m.

Manuscript received July 12, 2012; revised August 12, 2012.

The authors are with the Department of Computer Science and Engineering, Fukuoka Institute of Technology, 3-30-1 Wajiro-Higashi, Higashi-ku, Fukuoka, 811-0295 Japan (e-mail: oida@fit.ac.jp)

Fig. 1(a) shows the variance plot for two different video streams. Throughout the paper, the sampling interval δ is 10^{-5} (s) and the number of samples N is 6×10^6 (therefore, $N\delta$ is one minute). The sampling interval δ is set to 10 μs because the packet transmission time is 12 μs when the packet size is 1.5 Kbytes (a typical video packet size) and the transmission rate is 1 Gbps (the rate of our LAN). The exactly second-order self-similar process $\{X_k\}$ satisfies

$$Var X_k^{(m)} = m^{2H-2} Var X_k, \quad (3)$$

where $Var X_k$ is the variance of X_k and H is the Hurst exponent [5]. From (3), we have

$$2H-2 = \frac{\log_{10}\left(Var X_k^{(m)}\right) - \log_{10}\left(Var X_k\right)}{\log_{10}(m)}. \quad (4)$$

Since $V^{(m)}$ is an estimate of $Var X_k^{(m)}$, if $\{X_k\}$ is exactly second-order self-similar, slope β of the line through points of the variance plot satisfies $\beta = 2H-2$.

B. Decay Rates

A sampled traffic series shows self-similarity only in a limited range of m. In other words, slope β varies with m. Points in Fig. 1(a), for example, indicate two concave curves rather than straight lines. Hereafter, we assume that $m(\geq 1)$ is a real number. Let us consider the decay rate $\beta(m)$, the amount of change in $\log V^{(\lfloor m \rfloor)}$, defined by

$$\beta(m) = \log_{10} V^{(\lfloor 10^{\Delta} m \rfloor)} - \log_{10} V^{(\lfloor m \rfloor)}, \quad (5)$$

where Δ is a positive constant. Decay rate $\beta(m)$ varies with m since it is affected by many factors such as protocols, computer software and hardware, routers, propagation delays and resource (processors, bandwidth, etc.) competition among video streams, etc. In general, decay rate $\beta(m)$ at small levels m (i.e., small timescale behavior of $\beta(m)$) is determined by computer hardwares, communication devices, etc. that act quickly. Human users and protocols implemented by software affect even larger timescale behavior of $\beta(m)$. From the perspective of host identification, resource competition is a factor that adds noise to attribute $\beta(m)$. This paper assumes that human users do not affect video streams. That is, they do not change/interrupt streams while collecting traffic data.

Fig. 1(b) shows decay rate $\beta(m)$ of the two streams in Fig. 1(a). In the figure, the interval between two points (Δ) is constant; that is $\Delta = \log(m_{i+1}) - \log(m_i)$ for all i, where m_i is the i-th smallest level used to obtain the points. For simplicity, β_i is used to denote $\beta(m_i)$. We obtain Δ and m_i as $\Delta = \log(N/50)/(M+1)$ and $m_i = 10^{i\Delta}$, where M is the size of $\{\beta_i\}_{1\leq i\leq M}$. We use $M = 20$, unless otherwise mentioned. In this case, $\Delta \approx 0.24$. Uncorrelated streams (i.e., $H = 0.5$) satisfy $\beta(m) = -\Delta$. In general, β_i fluctuates around $-\Delta$. For effective identification, attributes used for identification should be mutually independent. Let $\{V_k^{(\lfloor m \rfloor)}\}$ be the aggregated variances of stream k. Fig. 1(a) shows that if there exists j that satisfies $V_1^{(\lfloor m_j \rfloor)} < V_2^{(\lfloor m_j \rfloor)}$, then $V_1^{(\lfloor m_i \rfloor)} < V_2^{(\lfloor m_i \rfloor)}$ for all i. In this case, we do not need more than one variance. Whereas, Fig. 1(b) shows that $\{\beta_i\}$ do not have such a feature.

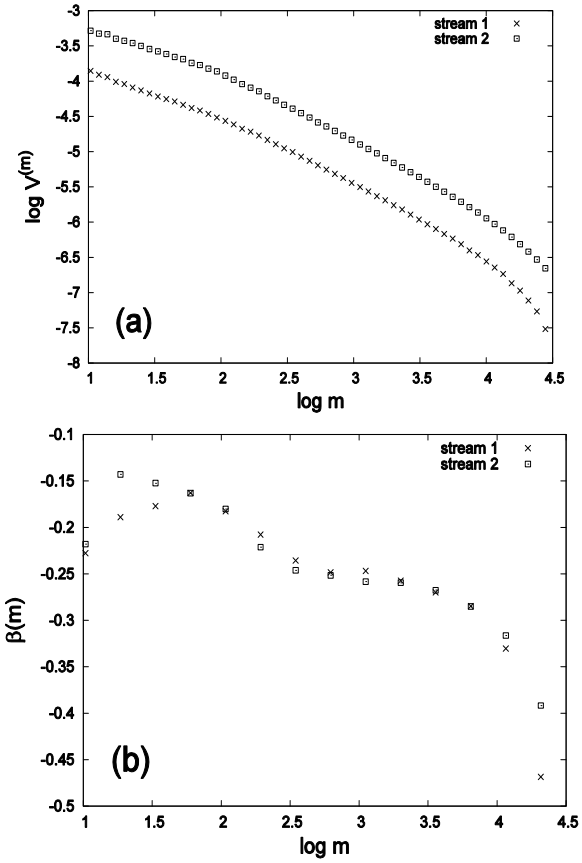
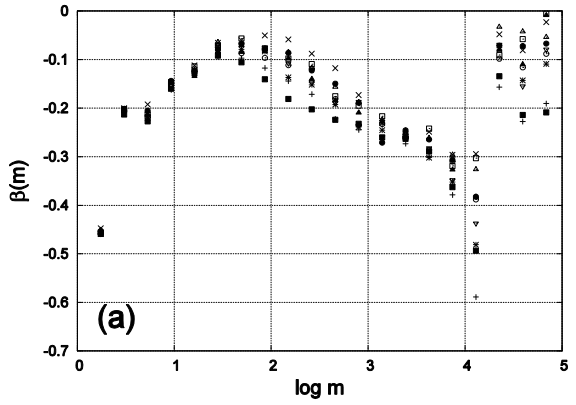


Fig. 1. (a) Variance plot and (b) decay rate for two streams.

III. Captured Packets

This paper focuses on the client-server model. One hundred TV channels are accessed with Flash Player ver. 11.2, Windows Media Player ver. 12, or Silverlight ver. 4.1. They are served by content delivery networks, Web hosting companies, live streaming video platforms, etc. All video streams use the TCP protocol and most of them flow at constant rates. Video bitrates are from 30 Kbps to 2097 Kbps. All packets from a TV site are captured by a client with WinDump [11]. They are used in experiments in Section IV.



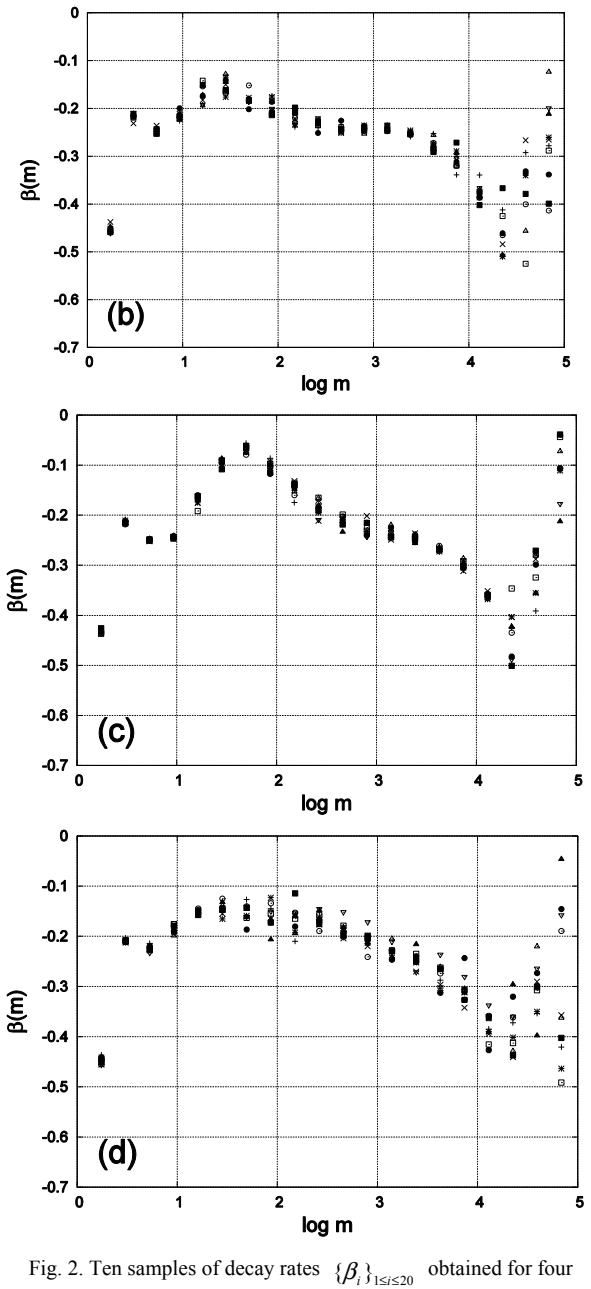


Fig. 2. Ten samples of decay rates $\{\beta_i\}_{1\le i\le 20}$ obtained for four characteristic streams in Table I.

TABLE I: TCP STATUSES OF FOUR STREAMS MEASURED IN 12 MINUTES.

Stream	Number of Connections	Number of retransmissions	RTT (ms) avg, stdev
(a)	1	0	6, 35
(b)	1	1032	46, 82
(c)	58 (sequential)	25 *	-
(d)	2 (parallel)	0 *	6, 34 **

* Totals of all connections.
** Statistics of the connection carrying a larger number of packets.

Table I shows TCP statuses of four characteristic streams: the variable-bitrate, congested, sequential, and parallel types. They are calculated by tcptrace [12]. In most cases, one TCP connection is established for a TV channel. Streams (a) and (b) in Table I correspond to this one connection type. Stream (c) is the sequential type, in which a connection is frequently replaced with a new one. Stream (d) is the parallel type, in which multiple connections are established to deliver video packets. There are two sequential-type and three parallel-type streams in the 100 TV channels.

This paper uses decay rates $\{\beta_i\}$ for host identification. Fig. 2 depicts ten samples of decay rates $\{\beta_i\}_{1\le i\le 20}$ for four streams (a)-(d) in Table I. The figure demonstrates that each stream has unique decay rates. The uniqueness and time-invariance of decay rates are key to successful identification, where the time-invariance requires that ten samples are not largely different. As shown in the figure, points at m_{19} and m_{20} are particularly dispersing. This is because the number of samples is small. From (2), $V^{(m)}$ is calculated with $\lfloor N/m \rfloor$ samples, which decreases with m. Therefore, $\beta(m)$ is unstable at large m. Due to wide dispersion, β_{19} and β_{20} are not useful for identification.

Fig. 2(a) shows that most of decay rates are variable. Stream (a) is the variable-bitrate type. The largest arrival rate $\overline{X}$ in ten samples is at least three times greater than the smallest. Not all TV channles provide good video quality. We experienced video quality degradation when watching at least three channels. From Table I, stream (b) must have experienced congestion (the congested type) since both the number of retransmissions and the standard deviation of RTTs in Table I are high. Compared with decay rates of streams (b) and (c), those of streams (a) and (d) vary more greatly over a wide range of m, even though their TCP statuses are normal. This result suggests that the variable-bitrate and parallel types have a greater tendency to cause identification errors.

IV. SUPERVISED LEARNING

A. Classifiers

Classifiers identify which of a set of classes a new sample belongs, on the basis of a training set of data that contain classes and their attributes, where a class is a client and server pair and attributes are decay rates $\{\beta_i\}$. Note that a class agrees with a server if there is only one client. This section uses three classifiers implemented in Weka ver. 3.6.6 [13]: rotation forest (RF) [14], naive Bayes (NB) [15], and k-nearest neighbor (KNN) [16], where $k=1$ throughout the paper. They are the best performed classifiers evaluated based on the true positive rate. The true positive rate (TPR) of class x is the percentage of samples which were classified as class x, among all samples which truly have class x. Since there is more than one class, the TPR is averaged over all classes.

Evaluations are made according to the cross-validation test, which is described as follows: The original sample is randomly partitioned into ten subsamples. A single subsample of the ten subsamples is retained as the validation data for testing the algorithms, and the remaining nine subsamples are used as training data. The cross-validation process is repeated ten times, with each of the ten subsamples used exactly once

as the validation data. The ten results are averaged to produce a single estimation.

B. One Client

A client accesses 100 TV channels. For each channel, ten

samples of attribute $\{\beta_i\}_{1\le i\le M}$ are calculated. Therefore, there are 1000 samples in total. They are used as input data for the classifiers. There are nine TV channels whose servers coincide with those of other channels. Thus, the number of classes results in 91. Fig. 3(a) shows TPRs when M varies. From the figure, the TPR is not very sensitive to M. TPRs of NB and RF are roughly 90%. Classifier RF achieves good performance, but its computation time is longer than those of NB and KNN. Meanwhile, Fig. 3(b) shows TPRs when the number of classes changes, where classes are selected randomly. The TPR decreases with the number. Classifiers RF and KNN correctly classify all samples when the number is twenty.

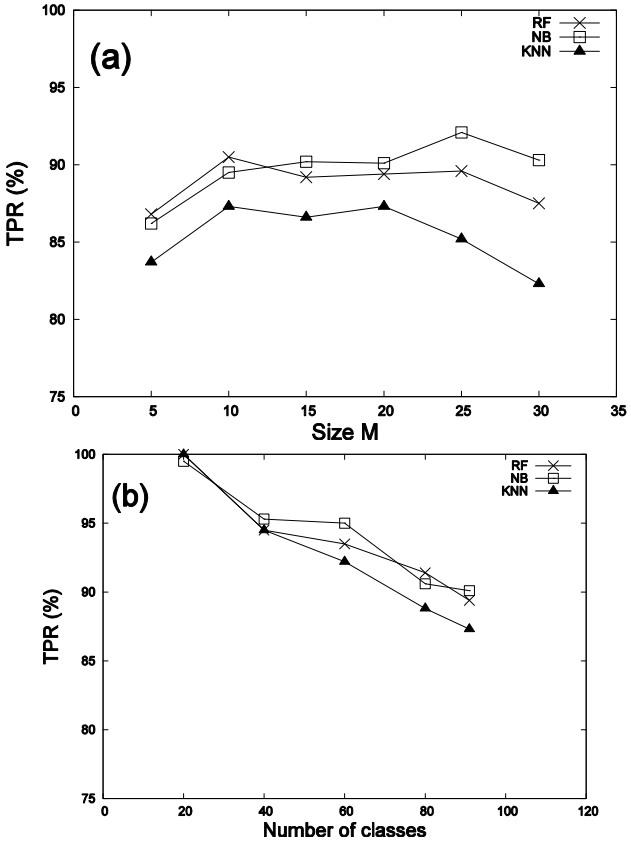


Fig. 3. TPRs (%) as functions of (a) the size of $\{\beta_i\}_{1\le i\le M}$ and (b) the number of classes.

Let us consider the case where a client receives two streams from a cluster of servers, all of which have exactly the same performance and functionality and are connected to the same LAN. The attributes in this paper cannot distinguish two servers in such a cluster. (We think the two servers should be regarded as identical.) We refer to this issue as the cluster problem. Fig. 4 may correspond to this case (but only the company knows it is true). As shown in the figure, except for variable decay rates at large levels, two servers present extraordinarily similar decay rates. Many errors made by three classifiers in Fig. 3(a) are due to this cluster problem if every cluster consists of exactly the same specification servers. Let us consider the case where the cluster problem does not occur. Table II shows that TPRs considerably increase by replacing 21 TV channels such that any two servers do not belong to the same cluster. The number of classes is 100 after the replacement.

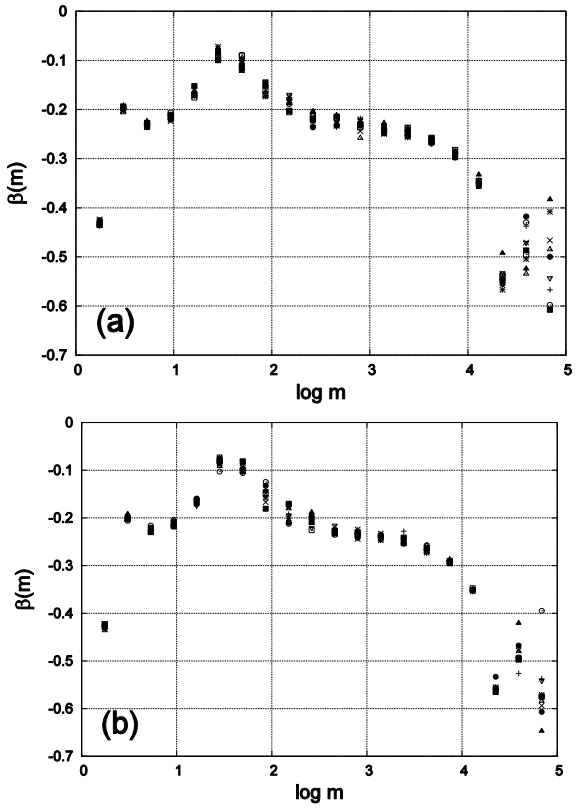


Fig. 4. Ten samples of $\{\beta_i\}_{1\le i\le 20}$ for two Akamai servers in Japan. The last decimal numbers of their IP addresses are (a) 126 and (b) 125.

TABLE II: TPRS (%) BEFORE AND AFTER REPLACING 21 TV CHANNELS.

	before	after
RF	89.4	93.4
NB	90.1	94.5
KNN	87.3	92.6

TABLE III: TPRS (%) IN THE ONE AND TWO CLIENT CASES.

	One client	Two clients
RF	89.4	96.1
NB	90.1	96.5
KNN	87.3	93.7

C. Two Clients

To investigate the sensitivity of attributes $\{\beta_i\}$ to changes in the communication environment, randomly selected 50 channels are newly accessed by another PC on the same LAN. We use original 100 TV channels in Section III. In this case, the number of classes (the combination of client and server pairs) increases from 91 to 96. Since the number of clients is two, classifiers can make use of behavioral differences between two clients if their hardware or software configurations are different. Furthermore, since there are cases in which two TV channels provided by a server cluster are accessed by different clients, the number of errors due to the cluster problem decreases. Table III demonstrates that all TPRs significantly increase. However, some errors made by three classifiers in the two client case are still caused by the cluster problem.

D. Sampling Period

To reduce the sampling period from 60 s to 20 s, N is set to 2×10^6. From (2), aggregated variance $V^{(m)}$ is calculated with $\lfloor N/m \rfloor$ samples. Therefore, if N decreases, we cannot obtain decay rates $\beta(m)$ at large m. Table IV shows TPRs in the one client case. Compared with TPRs in the one client case in Table III, all TPRs at 10 samples are considerably low. However, TPRs steadily rise with the number of samples per channel.

TABLE IV: TPRs (%) WHEN THE NUMBER OF SAMPLES PER TV CHANNEL VARIES. $M = 17$ AND $N\delta = 20$ S.

	10 samples	20 samples	30 samples
RF	77.8	81.2	83.3
NB	77.4	82.2	84.4
KNN	70.0	71.5	74.7

V. ATTRIBUTE EVALUATION

Let C be a random variable whose value is a class and let A be a random variable whose value is an attribute. The information gain is the mutual information $I(C;A)$ of C and A given by

$$I(C;A) = H(C) - H(C \mid A), \tag{6}$$

where $H(\cdot)$ stands for the entropy and $H(\cdot \mid A)$ is the entropy given that the value of random variable A is known. Thus, $I(C;A)$ is the reduction in the entropy of C achieved by learning the value of A. We use X, V, and B_i to indicate random variables of packet arrival rate $\overline{X}$, variance $V^{(1)}$, and decay rate β_i, respectively.

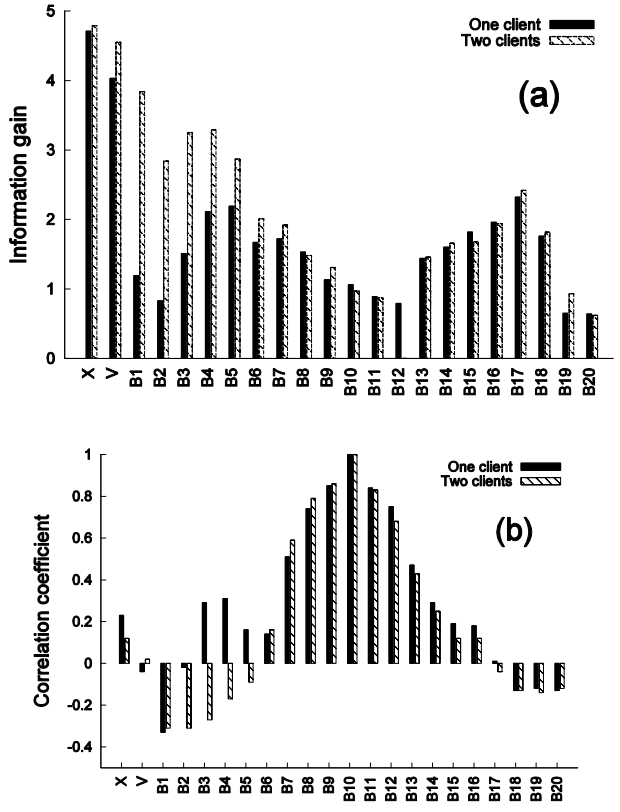


Fig. 5. (a) Information gains $I(C;\cdot)$. (b) Correlation coefficients $\rho(B_{10},\cdot)$.

Fig. 5(a) shows information gains in two cases: one client and two clients. The figure demonstrates that X and V provide the two highest information gains. Table V shows that by using $\overline{X}$, $V^{(1)}$, and $\{\beta_i\}_{1\le i\le 20}$ for classification, TPRs of all classifiers are further improved. If $\overline{X}$ is used, however, we perform stream identification rather than host identification since two streams flowing at different rates are differentiated even if their source and destination hosts are identical. The reason variance $V^{(1)}$ is not used is that $\overline{X}$ and $V^{(1)}$ are strongly correlated. The correlation coefficient between $\overline{X}$ and $V^{(1)}$ for the 100 streams in Section III is 0.82.

Fig. 5(a) shows that gains of B_{19} and B_{20} are small (i.e., useless), since their values largely fluctuate as shown in Fig. 2. It also shows that as the number of clients increases, information gains at decay rates $B_1, B_2, .., B_5$ significantly grow. Time scales of these decay rates range from $\lfloor 10^{\Delta} \rfloor\delta$ s ($=10$ μs) to $\lfloor 10^{6\Delta} \rfloor\delta$ s ($=280$ μs). This range is governed by client hardware and operating systems. In fact, the versions of the Windows operating system are different (7 and Vista), but the same version of Internet Explorer and Flash/Silverlight/WMP plugins are used on the two clients.

TABLE V: TPRs (%) FOR VARIOUS ATTRIBUTES. THE NUMBER OF CLIENTS IS ONE.

	$\overline{X}$, $V^{(1)}$, $\{\beta_i\}$	$\{\beta_i\}$	10 highest β_i	$\{\beta_{2i+1}\}$
RF	94.3	89.4	88.2	81.5
NB	94.7	90.1	88.2	85.1
KNN	90.7	87.3	85.7	71.4

Let us consider a sample of decay rates $\{\beta_i\}_{1\le i\le 20}$ as a point $(\beta_1, \beta_2, .., \beta_{20})$ in the 20-dimensional space. Classification divides the space into J disjoint regions such that points in region R_k belong to class C_k, where J is the number of classes. From Fig. 5(a), information gains of B_{19} and B_{20} are smaller than those of B_4 and B_5. This can be verified by looking at two planes $\beta_4 - \beta_5$ and $\beta_{19} - \beta_{20}$ in Fig. 6. In the $\beta_4 - \beta_5$ plane, points belonging to the same class tend to form a cluster. Whereas, in the $\beta_{19} - \beta_{20}$ plane, various class points are highly mixed, so that it is difficult to partition the plane.

Meanwhile, Fig. 5(b) shows correlation coefficients between B_{10} and a random variable A ($\rho(B_{10}, A)$). From the figure, B_{10} is strongly correlated with B_9 and B_{11}. In most cases, B_i and B_j are highly correlated if $|i-j| \le 2$. This result suggests that β_{2i+1}, $i = 0,1,\ldots,9$, could be used to reduce the number of attributes. However, Table V shows that the attributes do not provide good performance. The table recommends that ten highest attributes with respect to the information gain should be used if ten attributes are selected from twenty attributes $\{\beta_i\}_{1\le i\le 20}$. From Fig. 5(b), the correlaton coefficient between two decay rates is affected by the number of clients only at small levels. This result implies that the decay rate changes mostly at small levels.

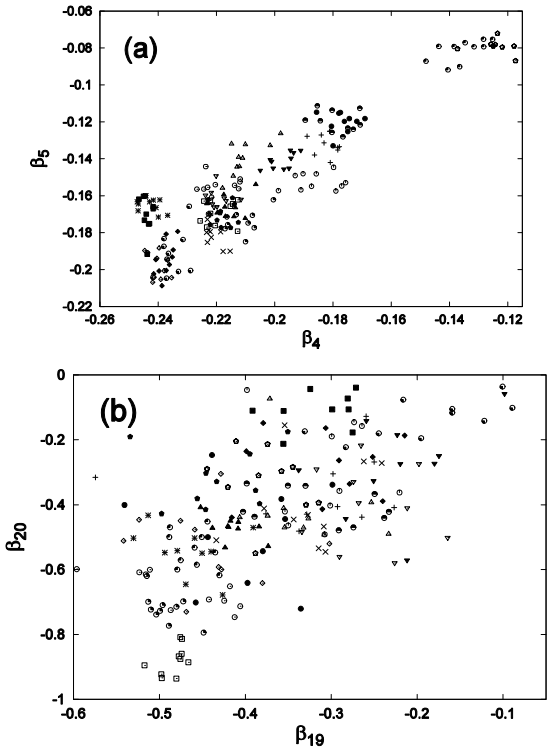


Fig. 6. Two planes $\beta_4 - \beta_5$ and $\beta_{19} - \beta_{20}$ for the one client case. A point style corresponds to one of thirty classes, which are randomly selected.

VI. Conclusions

To identify end hosts precisely, this paper took a new approach that makes use of statistical features of video traffic. We proposed using decay rates $\{\beta_i\}$ and evaluated their effectiveness with 100 Internet TV channels and three classifiers. Evaluations were made according to the cross-validation test. The experimental results are summarized as follows:

The naive Bayes classifier achieved the best performance in most cases, and the percentage of the correctly identified Internet TV sites was 94.5% if any two servers do not belong to the same cluster of servers.

The percentage steadily rose with the number of sample attributes $\{\beta_i\}$ per TV channel.

Decay rates $\{\beta_i\}$ at small levels sensitively responded to an increase in the number of clients and increased the percentage from 90.1% to 96.5%.

References

[1] J. C. Snader, "VPNs Illustrated: Tunnels, VPNs, and IPsec," *Addison-Wesley*, 2005.
[2] R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, and T. Berners-Lee, "Hypertext transfer protocol – HTTP/1.1," *RFC 2616*, June 1999.
[3] P. Srisuresh, J. Networks, and K. Egevang, "Traditional IP Network Address Translator (Traditional NAT)," *RFC 3022*, 2001.
[4] P. Ferguson and D. Senie, "Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing," *RFC 2827 (Best Current Practice)*, updated by RFC 3704, May 2000.
[5] K. Park and W. Willinger, "Self-Similar Network Traffic and Performance Evaluation," *Wiley-Interscience Published*, 2000.
[6] M. E. Crovella and A. Bestavros, "Self-Similarity in World Wide Web Traffic: Evidence and Possible Causes," *IEEE/ACM Transactions on Networking*, vol.5, no. 6, pp. 835-846, 1997.
[7] A. Feldmann, A. C. Gilbert, W. Willinger, and T. G. Kurtz, "The Changing Nature of Network Traffic: Scaling Phenomena," A*CM Computer Communication Review*, vol. 28, pp. 5-29, Apr. 1998.
[8] J. Beran, "Statistics for Long-Memory Processes," *Chapman and Hall*, New York, 1994.
[9] W. E. Leland, M. S. Taqqu, W. Willinger, and D. V. Wilson, "On the Self-Similar Nature of Ethernet Traffic (extended version)," *IEEE/ACM Transactions on Networking*, vol. 2, pp. 1-15, 1994.
[10] M. Taqqu, V. Teverovsky, and W. Willinger, "Estimators for Long-Range Dependence: an Empirical Study," *Fractals*, vol.3, pp.785-798, 1995.
[11] Software WinDump. [Online]. Available: http://www.winpcap.org/windump/
[12] Software tcptrace. [Online]. Available: http://www.tcptrace.org/
[13] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The WEKA Data Mining Software: An Update," SIGKDD Explorations, vol. 11, no. 1, 2009. [Online]. Available: Software Weka http://www.cs.waikato.ac.nz/ml/weka/
[14] J. J. Rodriguez, L. I. Kuncheva, and C. J. Alonso, "Rotation Forest: A new classifier ensemble method," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 10, pp.1619-1630, 2006.
[15] G. H. John and P. Langley, "Estimating Continuous Distributions in Bayesian Classifiers," *Eleventh Conference on Uncertainty in Artificial Intelligence*, San Mateo, pp.338-345, 1995.
[16] D. Aha and D. Kibler, "Instance-based learning algorithms," *Machine Learning*, vol. 6, pp. 37-66, 1991.