Classification of Network Traffic via Packet-Level Hidden Markov Models

Alberto Dainotti, Walter de Donato, Antonio Pescapè
Department of Computer Science and Systems
University of Naples “Federico II”
{alberto, walter.dedonato, pescape}@unina.it

Pierluigi Salvo Rossi
Department of Electronics and Telecommunications
Norwegian University of Science and Technology
salvoros@iet.ntnu.no

***Abstract*— Traffic classification and identification is a fertile research area. Beyond Quality of Service, service differentiation, and billing, one of the most important applications of traffic classification is in the field of network security. This paper proposes a packet-level traffic classification approach based on Hidden Markov Model (HMM). Classification is performed by using real network traffic and estimating - in a combined fashion - Packet Size (PS) and Inter Packet Time (IPT) characteristics, thus remaining applicable to encrypted traffic too. The effectiveness of the proposed approach is evaluated by considering several traffic typologies: we applied our model to real traffic traces of *Age of Mythology* and *Counter Strike* (two Multi Player Network Games), *HTTP, SMTP, Edonkey, PPlive* (a peer-to-peer IPTV application), and *MSN Messenger*. An analytical basis and the mathematical details regarding the model are given. Results show how the proposed approach is able to classify network traffic by using packet-level statistical properties and therefore it is a good candidate as a component for a multi-classification framework.**

I. Introduction

Network traffic classification is the process of analyzing traffic flows and associating them to different categories of network applications and it represents an essential task in the whole chain of network security. Studies in the field of traffic classification started in the last years, when the traditional use of transport protocol ports for classification purposes became unreliable while different kinds of new network applications were emerging (multiplayer network games, p2p IPTV, file sharing). Beyond the need to understand which kind of traffic is carried on the Internet links, other main motivations for looking for new and reliable traffic classification techniques today are to offer proper Quality of Service (QoS) depending on the category of traffic carried by flows, and to perform a billing not only based on bandwidth usage but also on the traffic category. However, in addition to these issues, some of the most important and widely spread applications of traffic classification pertain to network security: (i) the enforcement of security policies on the use of different applications; (ii) the ability to classify encrypted traffic; (iii) the identification of malicious traffic flows. For these reasons, several new approaches to traffic classification are being proposed and studied. As of today, though, no definitive answer is present. The debate in the scientific community is still open, and, as it happened in the recent past for intrusion detection systems [1], approaches based on the joint work of different traffic classification techniques (multi-classification) seem to be among the more promising solutions. New trends in network applications and protocol design, indeed, make traffic classification particularly difficult. Protocol encapsulation, encrypted transmission, use of non-standard ports, concerns related to users privacy, and need to keep up with huge traffic loads on network links are posing tremendous limits to some of the developed techniques. Payload inspection techniques, for example, make application identification difficult or even impossible under some of the above-cited conditions (mainly for both privacy and performance issues). On the other side, approaches based on statistical properties of the network traffic are looking more promising and robust to encryption, protocol obfuscation, privacy, etc.

In this paper we propose a novel classification technique based on packet-level statistical properties of network traffic exhibited by different applications. Specifically, we propose the use of Packet-Level Hidden Markov Models (PL-HMMs), that we have proposed and validated in the past for modeling purposes [2]. In this work we present the algorithms, the statistical properties taken in consideration, and we test the proposed classification approach on a set of application traffic that ranges from traditional network applications (e.g. HTTP, Email) to more recent ones as network games and peer-to-peer video streaming. The presented results are encouraging and show that the proposed PL-HMM approach may be a good candidate as a technique to be used in a multi-classification scenario (that is, when different classification engines are used and their output is combined by a decision system).

The rest of the paper is organized as follows. In Section II a brief description of the motivations is given. Section III provides details on the analytical model at the base of our classifier. Section IV discusses the applications considered and the measurement approach. Finally, in Section V we show results of traffic classification. Section VI ends the paper.

II. Motivation and Related Work

Several classification techniques have recently been presented in literature. Approaches based on deep payload inspection are usually considered very reliable for traffic that is not encapsulated into other application-level protocols and for un-encrypted traffic. However, the current trends show that

[0] This work has been partially supported by PRIN 2007 RECIPE Project, by CONTENT NoE, and NETQOS EU projects, by WILATI+ project.

the portion of encrypted traffic on the Internet is constantly increasing [3], and several applications are using protocol encapsulation or obfuscation to evade network policy enforced through filtering [4]. Moreover, access to full payload is often not possible (e.g. due to privacy issues). For these reasons, researchers are proposing approaches that look more robust because based on the intrinsic properties of the network traffic as it is generated by different applications. Flow-level parameters (e.g. flow duration, transmitted bytes, transmitted packets) are a popular choice, a valid alternative or combination is to exploit measurements coming from packet level (e.g. packet size, inter-packet times). Several notable works [5] [6] [7] [8] [9] presented in literature consider some of these properties to build classification features, and then use statistical or machine learning approaches to classification. Results show that a perfect classification approach does not exist. The use of different features and classifiers can bring more accuracy under some conditions or in identifying some applications while may not be satisfying in other cases. It is therefore probable that in the future we will see multi-classifier approaches, able to collect the advantages of different techniques and compensate for each weakness, being proposed.

In this paper we propose a technique for traffic classification based on a statistical approach that takes into account some new packet-level properties of network traffic, trying to offer a contribution in terms of techniques to exploit intrinsic properties of traffic generated by different network applications. Indeed, as explained in the following sections, the use of PL-HMMs allows us to take into account joint characteristics of inter-packet times (IPT) and payload size (PS), as well as their temporal correlation. We use studies from our modeling work based on HMMs [2]: the traffic generated by a specific application is modeled as a flow of packets, seen as a sequence of (IPT,PS) pairs generated according to different distributions depending on the hidden state of the source.

In [8], HMMs have been used, and compared with other techniques, for traffic classification of flows at an early stage. Sequences made of only the first 4 to 10 packets were used to train HMMs and to attempt flow classification. However, differently from our work, only packet sizes were considered in this paper. An approach based on profile HMMs has been proposed in [10]. This work is very different from ours, in that the authors present two separate classifiers working separately on IPTs or on PSs, and a left-to-right structure for the state topology of the HMM is used. However, a proposal for extending their approach was later presented in a technical report [11], where they try to account for joint IPT and PS modeling via vector quantization. Proposed profile HMMs in [11] present a very complex state structure depending on the length of the training sequence, with a pair of different states for each packet. They are designed for one-dimensional observable variables. IPT and PS joint information is taken into account via vector quantization, thus a codebook labeling IPT and PS allowed pairs is used as observable variable. Furthermore, a heuristic technique, namely model surgery, is needed to account for different trace lengths. As it will be clear in the next section, compared to [11], the model proposed in this paper works directly on a two-dimensional observable variable, thus exploits IPT and PS joint information without needing any pre-processing like vector quantization. Our approach presents a fully-connected structure for the state topology that allows an enormous reduction of the number of states, avoids post-processing like model surgery, and although being much less structured than the profile HMMs with respect to the traffic characteristics is still able to achieve good classification results.

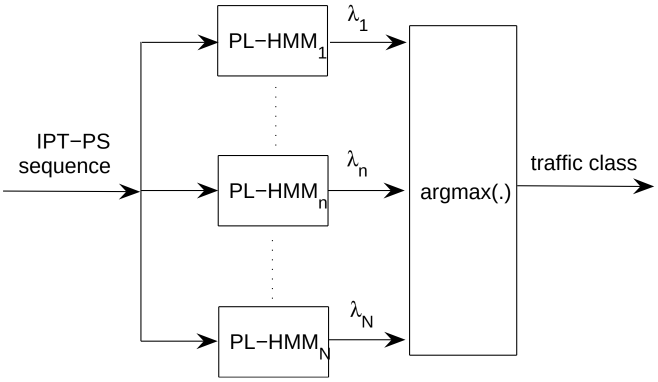


Fig. 1. Architecture of the classifier.

III. The Analytical Model

Notation - Column vectors are denoted with lower-case bold letters, with a_i denoting the ith element of vector $\boldsymbol{a}$; matrices are denoted with upper-case bold letters, with $A_{i,j}$ denoting the (i, j)th element of matrix $\boldsymbol{A}$; $(.)^{\mathrm{T}}$ and $\mathbb{E}\{.\}$ denote transpose and expectation operators; $a|b = b_0$ denotes the conditional random variable a given that $b = b_0$; the symbol $\sim$ means "distributed as".

Figure 1 shows the general system architecture that we are considering for traffic classification. It is composed by a bank of parallel PL-HMMs and a multi-input single-output block pointing at the maximum input. In order to capture the characteristics of N different typologies of network traffic, it is assumed that the N different PL-HMMs in the bank have been obtained via the Baum-Welch training proposed in [2]. The Baum-Welch algorithm [12] is an iterative procedure that looks for model parameters maximizing the probability that the model itself generates the sequences used as training set. Each PL-HMM of the bank is then used to compute the likelihood (λ_n), representing the probability that the test sequence belongs to the traffic typology associated to the PL-HMM. The maximum likelihood then selects the best estimate for the traffic typology.

A. PL-HMM

The single PL-HMM is an HMM composed by a discrete hidden state variable $x[\ell] \in \{s_1, \ldots, s_K\}$ and a continuous bi-dimensional observable variable, $\boldsymbol{y}[\ell] = (d[\ell], b[\ell])^{\mathrm{T}}$, where K denotes the number of the states for the HMM, $d[\ell]$ denotes $10 \log_{10}(\mathrm{IPT}/1\mu\mathrm{s})$ and $b[\ell]$ denotes PS of the ℓth packet. IPT and PS are jointly described with memory and correlation taken into account by the state variable, and assumed statistically independent given the state.

The single PL-HMM is characterized by the set of parameters $\mathcal{M} = \{\boldsymbol{A}, \boldsymbol{g}^{(t)}, \boldsymbol{w}^{(t)}, \boldsymbol{g}^{(p)}, \boldsymbol{w}^{(p)}\}$, denoting the state transition matrix, the conditional IPT and PS distribution vectors, respectively, i.e.

- $A_{i,j} = \Pr(x[\ell+1] = s_j | x[\ell] = s_i)$;
- $d[\ell] | x[\ell] = s_i \sim \text{Gamma}(g_i^{(t)}, w_i^{(t)})$;
- $b[\ell] | x[\ell] = s_i \sim \text{Gamma}(g_i^{(p)}, w_i^{(p)})$.

It is apparent the Markovian assumption for the hidden state. The conditional (in ith state) pdf's for IPT and PS, are

$$\begin{aligned} f_i^{(t)}(d) &= \frac{(d/w_i^{(t)})^{g_i^{(t)}-1} e^{-(d/w_i^{(t)})}}{w_i^{(t)} \Gamma(g_i^{(t)})} (d > 0) \; , \\ f_i^{(p)}(b) &= \frac{(b/w_i^{(p)})^{g_i^{(p)}-1} e^{-(b/w_i^{(p)})}}{w_i^{(p)} \Gamma(g_i^{(p)})} (b > 0) \; . \end{aligned}$$

It is worth noticing that, according to our notation, the IPT-PS sequence $\mathcal{Y} = (\boldsymbol{y}[1], \ldots, \boldsymbol{y}[L])$ corresponds to the following pair of sequences: $\mathcal{D} = (d[1], \ldots, d[L])$ for IPT values and $\mathcal{B} = (b[1], \ldots, b[L])$ for PS values.

B. Likelihood Computation

The likelihood $\lambda = \Pr(\mathcal{Y}|\mathcal{M})$ of an IPT-PS sequence $\mathcal{Y}$, given the model $\mathcal{M}$, is computed exploiting the dependencies captured by the model in both forward and backward directions. The Forward-Backward algorithm [12] is an efficient technique to compute the Forward variable α and the Backward variable β in a graphical model, i.e. the variables capturing such dependencies. More specifically, for HMM structures it is based on the following equations

$$\begin{aligned} \alpha_j[\ell] &= \sum_{i=1}^{K} \alpha_i[\ell-1] A_{i,j} f_j^{(t)}(d[\ell]) f_j^{(p)}(b[\ell]) \; , \\ \beta_i[\ell] &= \sum_{j=1}^{K} A_{i,j} f_j^{(t)}(d[\ell+1]) f_j^{(p)}(b[\ell+1]) \beta_j[\ell+1] \; . \end{aligned}$$

Basing on these formulas, the likelihood for an IPT-PS sequence $\mathcal{Y}$ is computed as

$$\lambda = \Pr(\mathcal{Y}|\mathcal{M}) = \sum_{i=1}^{K} \alpha_i[\ell] \beta_i[\ell] \; ,$$

for an arbitrary ℓ. The Forward-Backward algorithm is typically implemented in the log-domain.

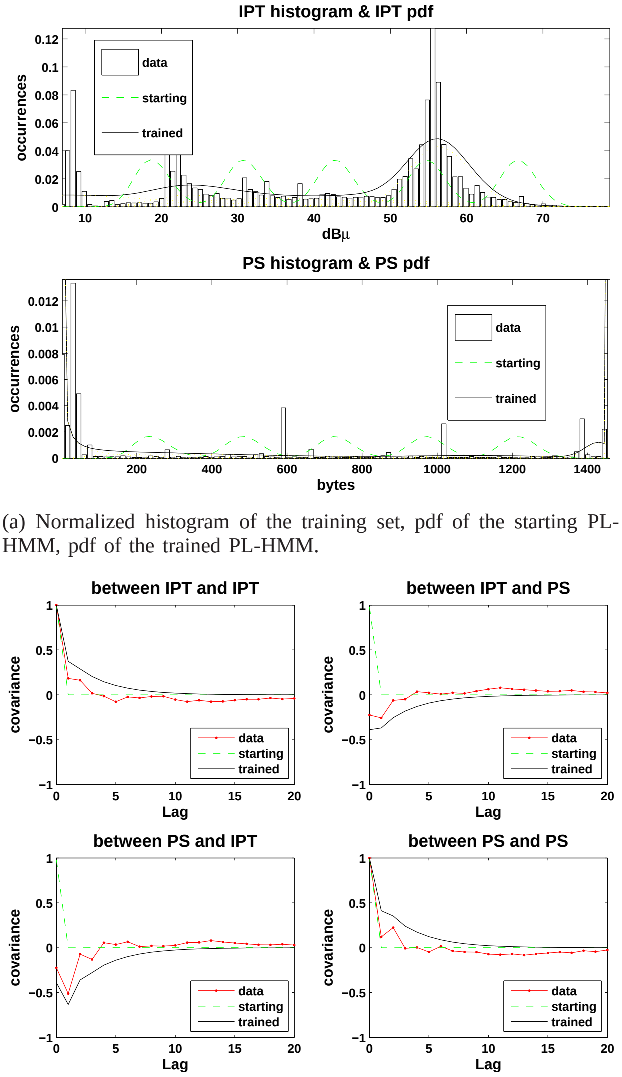
C. Trained PL-HMMs

Our trained PL-HMMs present $K = 4$ to $K = 7$ states, depending on the complexity of the protocol. We tried to keep the number of states as low as possible in order to contain computational complexity, and at the same time provide sufficient accuracy in modeling the characteristics of a specific a network-traffic typology.

The starting set of parameters for the training algorithm is chosen in order to cover almost uniformly the whole range of observed IPT and PS values. Convergence of the Baum-Welch training for all typologies was reached in a few (less than 10) iterations.

Figure 2 shows the characteristics of the PL-HMM trained to model SMTP traffic (please refer to Section IV for a description of all the applications considered in this work). From this figure, it is clear how first and second order statistics are captured by the model. This is shown also to give an intuitive idea of how packet-level properties related to marginal distributions, time dependence, and mutual dependence between IPT and PS, are captured by a trained PL-HMM made of few parameters, that can then be exploited for classification purposes. Table I shows the state parameters for the trained PL-HMM in which each state corresponds to a different short-time behavior of the application in terms of IPT and PS generation, for more details refer to [2]. Similar behavior in terms of modeling capabilities have been obtained for each of the traffic typologies described in Section IV.

Global statistics (average value and standard deviation) of



(a) Normalized histogram of the training set, pdf of the starting PL-HMM, pdf of the trained PL-HMM.

(b) IPT-PS auto- and cross-covariance for the training set, the starting PL-HMM, the trained PL-HMM.

Fig. 2. PL-HMM characteristics.

TABLE I

SMTP: STATE PARAMETERS.

	PS		IPT	
	$g_i^{(t)}$	$w_i^{(t)}$	$g_i^{(p)}$	$w_i^{(p)}$
1st state	196.91	0.28	0.15	308.12
2nd state	1.86	9.03	2.23	215.2
3rd state	22.94	1.07	1504	0.95
4th state	54	0.95	32.33	35.9
5th state	9.23	4.23	229828	0.0006

TABLE II

TRAINING SETS STATISTICS.

	IPT [dBμ]		PS [bytes]	
	mean	std dev.	mean	std dev.
AoM	47	9	13	4
CS	48	10	29	25
Edonkey	49	10	1182	377
HTTP	48	13	703	460
MSN	56	15	575	572
PPLive	66	4	177	271
SMTP	41	18	616	624

the training sets used to characterize each traffic typology are shown in Table II. It is easy to notice that IPT and PS joint characterization is needed in order to aim at successful classification. Also, analyzing differences and similarities among traffic characteristics, it is not surprising that, anticipating the results shown in Section V, AoM and PPlive will present the two best performance for correct classification, while the the worst performance for misclassification will be when confusing Edonkey with SMTP and SMTP with MSN.

IV. CONSIDERED APPLICATIONS AND MEASUREMENT APPROACH

We tested our algorithm over a heterogeneous set of network applications, shown in Table III. Each of them were verified through deep payload inspection and manual checks. The choice of the considered applications to classify was driven by the following multidimensional criteria: (i) both TCP and UDP based applications; (ii) both data and signaling traffic; (iii) both traditional and novel Internet applications. As for TCP-based and traditional applications we considered the data traffic of HTTP and SMTP (respectively related to Web and Email), still responsible for a relevant portion of the overall Internet traffic. Again, in the class of TCP-based applications and still falling in the category of traditional Internet applications, we considered Instant Messenging. It is used by about 50% of the Internet users all around the world [13], with MSN Messenger (MSN in the following) being the most popular application. In this work we consider the traffic generated by MSN clients [14]. Also, as last TCP-based application we considered the traffic associated to the Edonkey protocol [15], used by peer-to-peer file sharing applications as Emule. This category of traffic is quite novel (compared to Web and Email traffic) and it is particularly important because most of the issues related to the inability to identify applications through protocol ports started with respect to peer-to-peer file sharing applications. As regard UDP-based and innovative (and with QoS requirements) applications, we considered the traffic generated by Age of Mythology (AoM) [16], a Real Time Strategy Multiplayer game, and CounterStrike (CS) [17], one of the most played First Person Shooter games on the Internet.

TABLE III

CONSIDERED TRAFFIC

	Training			Test		
	flows	packets	bytes	flows	packets	bytes
AoM	4	109887	1.3 M	2	55671	702 K
CS	344	35108	1 M	340	27916	881 K
Edonkey	109	245290	289 M	82	190526	228 M
HTTP	7520	311661	219 M	7771	281484	188 M
MSN	18007	902375	518 M	17836	922686	557 M
PPlive	137	4520	799 K	157	6658	713 K
STMP	50070	1385238	853 M	61738	1727850	1266 M

Finally, a category of traffic that is now constantly increasing is peer-to-peer video streaming. Triple-player Operators are interested in identifying and classifying this traffic without damaging the privacy of the users. For this reason, we considered the signaling traffic generated by the PPlive application. Therefore, according to our multidimensional criteria, this last traffic typology falls in the class composed by the triple: UDP-based application, innovative Internet service, signaling traffic. To stress the importance of peer-to-peer video streaming traffic in current networks, it is worth noticing that we previously studied the traffic generated by PPlive and, while we were able to recognize that the signaling information was transmitted through UDP packets and the video data was carried by TCP packets, we were not able to reliably identify all the video streaming flows on TCP. Thus confirming that, from the Operator point of view, the ability to recognize signaling traffic instead of data traffic is of indisputable importance.

Except for network games, all the traffic was captured at University of Naples "Federico II", Italy, with the traffic from peer-to-peer applications generated by a set of controlled boxes. The AoM traces, instead, have been provided by the Worcester Polytechnic Institute, MA (USA) [18]. Whereas the CS traces have been already used for a study on network games traffic modeling [19]. According to the results shown in [20] we can state that the time invariance of IPT does not affect the classification process (based on both IPT and PS).

We considered the conventional definition of flows - given by the 4-tuple: *source IP*, *source port*, *destination IP*, *destination port* - with a timeout of 60 seconds. In this study we took into account only traffic exiting from observed hosts (e.g. packets with destination port 80 or 25 for HTTP and SMTP respectively, packets sent by observed machines in the case of peer-to-peer applications, etc.), neglecting flows in the opposite direction. We separated the available flows in two separate sets: a *training set* used for training the PL-HMM and thus building the models, and a *test set* used to verify the classifier. Flows with less than 10 packets have been excluded both from training and test sets in order to avoid numerical problems running the algorithms. From each considered flow we extracted sequences of IPT and PS. Since we wanted to characterize the traffic generated by the applications, independent as much as possible of the transport protocols, we dropped all packets with empty payload, as TCP-specific traffic, like connection establishment packets (SYN-ACK-SYNACK) and pure acknowledgment packets. For the same reason, in the estimation of the PS, we measured the byte length of the TCP/UDP payload.

TABLE IV

CLASSIFICATION RESULTS: CONFUSION MATRIX

	AoM	CS	Edonkey	HTTP	MSN	PPlive	SMTP
AoM	**100.00**%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
CS	2.94%	**93.53**%	2.94%	0.00%	0.29%	0.00%	0.29%
Edonkey	0.00%	1.22%	**90.24**%	1.22%	2.44%	1.22%	3.66%
HTTP	0.01%	0.04%	1.13%	**93.35**%	2.81%	0.49%	2.17%
MSN	0.00%	0.13%	2.34%	0.94%	**94.16**%	0.00%	2.43%
PPlive	0.00%	0.00%	0.64%	0.64%	1.91%	**96.82**%	0.00%
SMTP	0.00%	2.04%	2.23%	2.25%	3.25%	0.00%	**90.23**%

V. EXPERIMENTAL RESULTS

In Table IV we show, summarized through a confusion matrix, the results of the classification performed on the test sets. Each row represents in percentage the output of a run of the classifier over a different application test set (e.g. the cell corresponding to the HTTP row and Edonkey column tells us that 1.13% of the flows from the HTTP test set have been erroneously classified as Edonkey). All the correct classification percentages are shown on the diagonal in bold. We can see that for all the applications a correct classification percentage above 90% is achieved, with the best results obtained when trying to identify AoM and PPlive traffic. For AoM the 100% percentage value is mainly explained with the very reduced number of flows of the test set, however it is important to note that the confusion values observable on the AoM column show that it almost never happens that flows from different applications are erroneously classified as AoM (this actually happens only for CounterStrike which is a game over UDP as AoM), demonstrating that the AoM model is very strict in capturing AoM traffic properties. The worst results are obtained when trying to identify Edonkey or SMTP traffic. Here we see that there are several flows that are confused with other applications. Probably the considered statistical properties of such flows do not fit with their corresponding models. However, this is a typical situation in which a multi-classifier system may override the weaknesses of a single approach by counting also on different classification techniques based on other properties. Moreover, it is worth noticing that in this work we considered only traffic in one direction for each host, whereas by building models also for the other way and exploiting the bond between corresponding flows in the two directions (being both generated by the same application) it may be possible to achieve a better accuracy. The extension of the classifier aiming to process both traffic directions at the same time is currently under investigation.

VI. CONCLUSION

Traffic classification represents an essential task for both network management architectures [23] and network security solutions [24]. In this paper we proposed an approach for traffic classification based on HMMs applied to packet-level traffic parameters. Our approach, by jointly considering IPT and PS and taking into account also their temporal structures, is able to classify a number of traffic typologies (TCP and UDP based, data and signaling, traditional and novel Internet applications). We showed how the technique is able to achieve promising results such that it may be considered as one of the techniques to be used in a multi-classifier system. Our ongoing work is devoted to both preliminary longitudinal/portability analysis (i.e. training and testing stage using different traffic traces) and enlarge the set of considered traffic typologies. Moreover we plan to compare performance against other classifiers.

REFERENCES

[1] G. Giacinto, F. Roli, L. Didaci, “Fusion of multiple classifiers for intrusion detection in computer networks,” *Pattern Recognition Lett.*, Vol. 24, no. 12, pp. 1795–1803, Aug. 2003.

[2] A. Dainotti, A. Pescapé, P. Salvo Rossi, G. Iannello, G. Ventre, F. Palmieri, “An HMM Approach to Internet Traffic Modeling,” *IEEE Global Telecommun. Conf. (GLOBECOM)*, pp. 1–6, Dec. 2006.

[3] http://www.net-security.org/secworld.php?id=4852, Mar. 2008.

[4] T. Karagiannis, A. Broido, N. Brownlee, K.C. Claffy, M. Faloutsos, “Is P2P dying or just hiding?,” *IEEE Global Telecommun. Conf. (GLOBECOM)*, pp. 1532–1538, Dec. 2004.

[5] S. Zander, T. Nguyen, G. Armitage, “Automated traffic classification and application identification using machine learning,” *IEEE LCN*, pp. 250–257, Nov. 2005.

[6] M. Crotti, F. Gringoli, P. Pelosato, L. Salgarelli, “A Statistical Approach to IP-level classification of network traffic,” *IEEE Int. Conf. Commun. (ICC)*, pp. 170–176, Jun. 2006.

[7] J. Erman, A. Mahanti, M. Arlitt, “Internet Traffic Identification using Machine Learning,” *IEEE Global Telecommun. Conf. (GLOBECOM)*, pp. 1–6, Dec. 2006.

[8] L. Bernaille, R. Teixeira, K. Salamatian, “Early Application Identification,” *ACM Co-Next*, 2006

[9] T. Auld, A.W. Moore, S.F. Gull, “Bayesian Neural Networks for Internet Traffic Classification,” *IEEE Trans. Neural Networks*, Vol. 18, no. 1, pp. 223–239, Jan. 2007.

[10] C. Wright, F. Monrose, G. Masson, “HMM Profiles for Network Traffic Classification”, *VizSEC/DMSEC*, pp. 9–15, Oct. 2004.

[11] C. Wright, F. Monrose, G. Masson, “Towards better protocol identification using profile HMMs”, JHU Tech. Rep. JHU-SPAR051201, Jun. 2005.

[12] L.R. Rabiner, “A tutorial on Hidden Markov Models and Selected Applications in Speech Recognition”, *Procs. IEEE*, Vol. 77, no. 2, pp. 257–285, Feb. 1989.

[13] http://www.comscore.com/, Sep. 2007.

[14] http://join.msn.com/messenger/overview, Sep. 2007.

[15] http://sourceforge.net/projects/pdonkey/, Mar. 2008.

[16] http://www.microsoft.com/games/ageofmythology/, Mar. 2008.

[17] http://www.counter-strike.net/, Mar. 2008,

[18] http://nile.wpi.edu/downloads, Sep. 2007.

[19] W. Feng, F. Chang, W. Feng, J. Walpole, “A Traffic Characterization of Popular On-line Games,” *IEEE/ACM Trans. Networking*, Vol. 13, no. 3, pp. 488–500, Jun. 2005.

[20] A. Botta, A. Dainotti, A. Pescapé, G. Ventre, “Searching for Invariants in Network Games Traffic,” *Poster at Co-Next 2006 Student Workshop*.

[21] http://www.microsoft.com/technet/prodtechnol/isa/2000/maintain/isaimsec.mspx, Sep. 2007.

[22] http://www.hypothetic.org/docs/msn/general/overview.php, Sep. 2007.

[23] H. Jiang, A.W. Moore, Z. Ge, S. Jin, J. Wang, “Lightweight Application Classification for Network Management,” SIGCOMM Work. Internet Network Manag., Aug. 2007.

[24] O. Marques, P. Baillargeon, “Design of a multimedia traffic classifier for Snort,” Information Manag. & Computer Security J., Vol. 15, no. 2, Jun. 2007.