Pair-wise Similarity Criteria for Flows Identification in P2P/non-P2P Traffic Classification

José Camacho, Pablo Padilla, F. Javier Salcedo-Campos, Pedro García-Teodoro, Jesús Díaz-Verdejo
Dpt. of Signal Theory, Telematics and Communications,
CITIC - Faculty of Computer Science and Telecommunications - University of Granada,
C/ Periodista Daniel Saucedo Aranda s/n 18071 GRANADA (Spain).
josecamacho@ugr.es, pablopadilla@ugr.es, fjsalc@ugr.es, pgteodor@ugr.es, jedv@ugr.es

***Abstract*—There is a growing interest in network traffic classification without accessing the packets payload. A main concern for network management is *peer-to-peer* (*P2P*) traffic identification. This can be performed at several levels, including packet level, flow level and node level. Most current traffic identification approaches rely on flow level identification, being highly demanding and time consuming procedures. This paper introduces a similarity-based method to pair flows up, which is aimed at reducing the cost of identifying P2P/non-P2P traffic flows. For that, different similarity measures for flows pairing are proposed and analyzed.**

***Keywords*-Traffic classification; peer-to-peer; k-Nearest Neighbors**

I. INTRODUCTION

The increasing popularity and expansion of peer-to-peer (P2P) networks and applications has raised some engineering issues related to traffic and security. On the one hand, Internet service providers need to handle the large volume of traffic yielded by P2P activities to assure the minimal impact to other network services. Moreover, the exchange of any kind of information between the so-called peers, most of them anonymous, is a security risk. This risk affects users in particular, since the information exchanged might contain viruses, worms and malware. It also affects the network infrastructure, since P2P applications can be used to support other harmful activities such as coordinated DoS attacks, botnets, etc.

In this context, there is a clear interest in P2P traffic identification. This paper introduces a new method aimed at reducing the cost of identifying P2P/non-P2P traffic flows. The rest of the article is organized as follows: Section II reviews the state of the art of traffic classification and P2P traffic identification. Section III introduces the datasets used in the experimentation. Section IV motivates the use of macro-flows built upon pairs of flows. In Section V some strategies for flows pairing are presented. Section VI is devoted to compare the results obtained by these different strategies, and finally, the conclusions are drawn in Section VII.

II. STATE OF THE ART

The recognition of P2P traffic is part of a more general problem, namely the identification of network traffic [1]. Three main problems arise in the identification of the traffic on a network:

1) Characterization: There are many features that have been proposed in the literature to represent and classify network traffic. The information used includes a wide variety of parameters, from statistical data of connections from SNMP routers reports [2] (low granularity) to information obtained from TCP headers, including the signaling bits and the first bytes of payloads (high granularity) [3].
2) Identification level: Once the traffic has been parameterized, three levels are considered to perform the identification [1], [4]: node level, packet level and flow level. In the first case, the objective is to identify nodes that generate a certain type of traffic [5]. The aim of packet-based identification is to classify each packet individually. In the flow-based identification, the goal is to determine the application protocol that generates each traffic flow.
3) Identification process: A wide variety of recognition systems are used to perform the identification, ranging from heuristic or signature-based [1], [6], [7] to data mining or pattern recognition algorithms [4], [8].

P2P flow recognition has been attempted by using a number of techniques. Among them, the *k*-Nearest Neighbors (kNN) technique is remarkable because of its simplicity and high recognition rate reported. Jun et *al.* [9] performed a comparison between a number of techniques including Naïve Bayes, decision trees, kNN and other methods to classify flows from 12 different application protocols, where some of them are P2P (BitTorrent and Gnutella) and the rest non-P2P (HTTP, DNS, POP3, etc.). The results show that kNN is the best technique in terms of precision rate. Lim et *al.* [10] proposed a discretization of standard parameters of traffic flows (ports, package sizes, number of packets, duration of flow, etc..) and assessed four classification techniques: support vector machines (SVMs), kNNs, Naïve Bayes and

 ISBN: 978-1-61208-173-1

decision trees. The results indicate that the performance of kNN is similar to SVMs, which yielded the best performance. Salcedo-Campos et *al.* [11] proposed a kNN-based technique called MVC (Multiple Vector Classification) for P2P traffic identification. This method combines three kNNs applied over different sets of parameters obtained from the flows.

Most current traffic classification approaches rely on flow level techniques. Despite the good classification performance usually obtained by them, the general process is highly time consuming. In order to overcome such limitation, this paper introduces several similarity measures for flows pairing in order to identify groups of flows likely to be generated by the same protocol/service. This way, once a flow is identified with a well-known procedure (e.g., DPI tools), all the flows which are similar to it according to the flows pairing will also be (quickly) identified. The proposed pairwise approach for flows classification takes advantage of the good performance exhibited by kNN classifiers.

III. Network traffic data for the experimentation

In order to evaluate the approach and methods described in this work, an experimental setup with two steps has been considered. The first one includes the capture of a great amount of real network traffic, in this case, acquired in an academic institution network. The second one consists of the automatic classification of all the captured traffic packets and flows by means of a deep packet inspection (DPI) tool. In this scenario, the *ground truth* data-set is constituted taking into account the analysis and identification of each traffic flow and its associated traffic packets with a DPI tool, in this case openDPI [12], with a negligible percentage of classification errors.

The database used in this work contains the data captured during three days of network inspection in an academic institution. The acquisition was performed in the access router in order to control the incoming and outgoing traffic of the inner nodes of the network. The traffic flows and their packets are captured in both communication ways.

The original data-set has been divided into a calibration subset of 100,000 flows and a test subset with 100,000 flows. It should be remarked that the flows are sequentially organized so that the period corresponding to the calibration subset is previous to the one of the test subset, with no time period overlapping. Table I shows the amount of P2P traffic in both calibration and test subsets. OpenDPI tool found 35 and 41 different protocols in the calibration and test subsets, respectively.

The openDPI classification shows that HTTP is the protocol with the highest number of flows, while the portion of P2P protocols is close to 9%. Although this P2P fraction could be considered reduced, the P2P traffic volume associated is high, due to the size of each P2P flow. A more detailed analysis shows that only a reduced number of network nodes generate or receive P2P traffic, being more relevant videostreaming related protocols, which contribute to the HTTP traffic (i.e., YouTube traffic). The rest of non-P2P flows include mainly habitual protocols, such as DNS, SSL or email protocols. The majority of the P2P flows are related to BitTorrent, meanwhile Gnutella and others are found in a lower proportion. This proportion may be considered a consequence of the particular features of the protocols. The relation between the P2P traffic and the non-P2P traffic is similar in both calibration and test data-sets. Please, refer to [11] for a more detailed description of the protocols in the data set.

Table I
BASIC TRAFFIC DESCRIPTION OF THE CALIBRATION AND TEST SUBSET.

	Flows		
Subset	Total	P2P flows	non-P2P flows
Calibration	100,000	8,897	91,103
Test	100,000	8,916	91,084
Total	200,000	17,813	182,187

The feature vector representing each flow is composed of 61 variables, as Table II depicts. The feature vectors contain all the information needed for posterior analysis, including the flow identification label in the database, the protocol detected by openDPI and some traffic information concerning the flow. The IP addresses of each flow have been sorted by number. The term UP (ascending) points out that the packet is going towards the machine with the highest IP, and the term DOWN (descending) indicates that the packet is going towards the machine with the lowest IP.

In the rest of the text, the terms observation and feature vectors are used interchangeably. Two levels of classification are established: the first one considering all the protocols in the subsets and the second one considering only two classes indicating if the flows are related to P2P or non-P2P protocols.

IV. Motivation

In kNN classification, an object is assigned to the most common class amongst its k nearest neighbors. In this section, the kNN technique in its simplest form (k=1) will be applied to the calibration data-set in order to motivate the approach adopted in this paper. From here onwards, let us call this the NN technique. The NN classifies an observation (feature vector or flow) within the same class of the nearest observation in the calibration data. To establish the nearest observation to a given one, a closeness functional needs to be defined, typically based on well-known distances. An often used distance is the normalized Euclidean distance, where all variables have been normalized in variance:

$$d(\mathbf{x},\mathbf{y}) = \sqrt{(\mathbf{x}-\mathbf{y})^t \cdot \mathbf{S}^{-1} \cdot (\mathbf{x}-\mathbf{y})} \qquad (1)$$

 ISBN: 978-1-61208-173-1

Table II
VARIABLES OF THE FEATURE VECTOR FOR EACH FLOW.

Value	Description
	Flow identification
ID_FLOW	Flow ID
IP_LOW	Lower IP of the session tuple
IP_UPPER	Highest IP of the session tuple
PORT1	Port related to the lowest IP (IP_LOW)
PORT2	Port related to the highest IP (IP_UPPER)
PROT_UDP	Transport protocol UDP
PROT_TCP	Transport protocol TCP
PROT_UNK	ICMP
DIR	Direction of the first observed packet (UP or DOWN)
FIRST_TIME	Timestamp of the first packet (μs)
LAST_TIME	Timestamp of the last packet (μs)
	Related to transfer
NPACKETS	Number of packets in flow
NPACKETS_UP	Idem way UP
NPACKETS_DOWN	Idem way DOWN
PACKETS_SIZE	Complete size of all the packets in the flow
PACKETS_SIZE_UP	Idem way UP
PACKETS_SIZE_DOWN	Idem way DOWN
PAYLOAD_SIZE	Complete size of payloads
PAYLOAD_SIZE_UP	Idem way UP
PAYLOAD_SIZE_DOWN	Idem way DOWN
MEAN_PACK_SIZE	Mean packet size
MEAN_PACK_SIZE_UP	Idem way UP
MEAN_PACK_SIZE_DOWN	Idem way DOWN
SHORT_PACKETS	Number of short packets
SHORT_PACKETS_UP	Idem way UP
SHORT_PACKETS_DOWN	Idem way DOWN
LONG_PACKETS	Number of large packets
LONG_PACKETS_UP	Idem way UP
LONG_PACKETS_DOWN	Idem way DOWN
MAXLEN	Maximum packet size
MAXLEN_UP	Idem way UP
MAXLEN_DOWN	Idem way DOWN
MINLEN	Minimum packet size
MINLEN_UP	Idem way UP
MINLEN_DOWN	Idem way DOWN
	Related to time
DURATION	Flow duration (μs)
MEAN_INTERAR	Mean time between consecutive packets
MEAN_INTERAR_UP	Idem only UP
MEAN_INTERAR_DOWN	Idem only DOWN
MAX_INTERAR	Maximum time between consecutive packets
MAX_INTERAR_UP	Idem only UP
MAX_INTERAR_DOWN	Idem only DOWN
MIN_INTERAR	Minimum time between consecutive packets
MIN_INTERAR_UP	Idem only UP
MIN_INTERAR_DOWN	Idem only DOWN
	Signaling
N_SIGNALING	Number of packets containing flags
N_SIGNALING_UP	Idem way UP
N_SIGNALING_DOWN	Idem way DOWN
NACKS	Number of packets with ACK flag active
NFIN	Idem FIN
NSYN	Idem SYN
NRST	Idem RST
NPUSH	Idem PSH
NURG	Idem URG
NECE	Idem ECE
NCWD	Idem CWD
NACK_UP	Number of packets UP with ACK flag active
NACK_DOWN	Idem way DOWN
NFIN_UP	Idem FIN & UP
NFIN_DOWN	Idem FIN & DOWN
NRST_UP	Idem RST & UP
NRST_DOWN	Idem RST & DOWN

where $\mathbf{S}$ is a diagonal matrix containing the sampling variances of the variables.

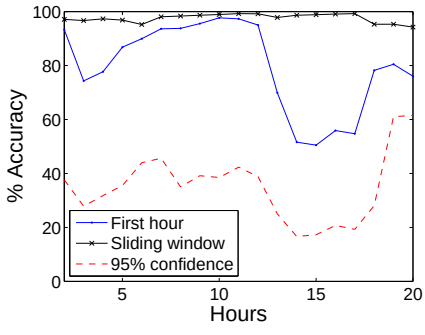
In Figure 1, the performance of the NN technique for traffic classification (Figure 1(a)) and P2P traffic identification (Figure 1(b)) is assessed with the calibration data following two different approaches. The first approach considers the traffic corresponding to the first hour as the calibration data for NN. Then, traffic classification and P2P identification are performed over the rest of the flows up to the 20th hour. Notice that the first 20 hours correspond to the calibration subset introduced in the previous section. The test subset is only employed in the experiments of Section V. The second approach considers a sliding window of one hour as the calibration data for NN. Thus, to classify an observation the nearest neighbor is obtained from the immediate preceding observations within one hour interval. Both methods are compared to a 95% confidence level for statistical significance, computed using permutation tests, a.k.a. randomization tests [13], [14]. The confidence level is useful to assess the expected performance of a random classifier in a given data-set, in order to test whether the performance of the present classifier is beyond what it is expected just by chance. Thus, in Figure 1(b), the expected accuracy of a random classifier is high (between 50% and 95%) due to the low percentage of P2P flows in the data in comparison with non-P2P flows. The random performance also changes over time due to changes in the percentage of P2P traffic. The good performance of NN is evidenced in the figures since both approaches are far above the confidence level. Also, the sliding window approach outperforms the static window in the first hour.

Another interesting question is how the similarity between flows is affected by the coincidence of IP addresses. An experiment to check this is shown in Figure 2. The 20th hour interval of traffic from a specific IP (the most common one) was classified using the NN technique from two different data-sets obtained from the previous 19 hours: traffic from the same IP and traffic from the rest. For a fair comparison, both data-sets had the same number of flows and non statistical differences on time stamp. According to the figure, most of the correct traffic classification and P2P traffic identification is obtained for traffic with the same IP.

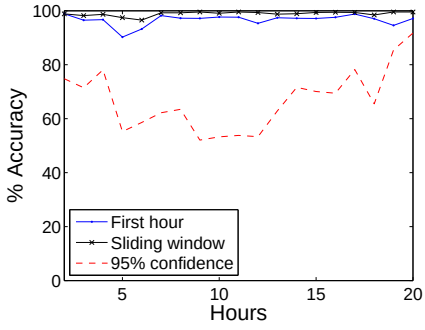
V. STRATEGIES FOR FLOWS PAIRING

The results in the previous sections show that the good performance of NN is almost restricted to traffic with the same IP. This represents a severe limitation for the general application of NN to on-line traffic classification, since it cannot be applied to traffic coming from new IPs not previously considered. Furthermore, the performance of NN is expected to degrade with the time separation between calibration flows and test flows. Finally, taking into account that calibration flows need an additional classification mechanism to perform NN, for instance payload-based classification, the direct application of NN in traffic classification is not recommended.

Nevertheless, flows identification methods based on pairing can take advantage of this good performance of NN. From the previous results, a convenient approach for



(a)



(b)

Figure 1. Percentage of accuracy for the calibration data-set in (a) traffic classification and (b) P2P traffic identification. The confidence level is computed using randomization tests.

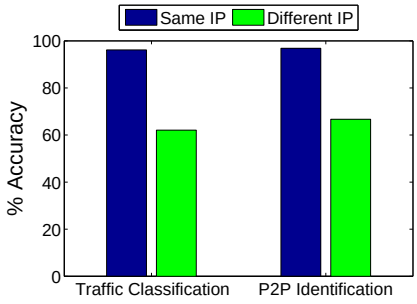


Figure 2. Percentage of accuracy in traffic classification and P2P traffic identification for the 20th hour traffic from a given IP. The performance of NN using past traffic from the same IP is compared to that of NN using past traffic from different IPs.

flows pairing is to use a time sliding window, where only those flows which share at least one IP with the current flow are considered as potential candidates for pairing. This approach has been combined with payload-based classification methods by the authors in some preliminary experiments, yielding less than 5% of payloads inspection to identify correctly close to 100% of flows. This low payload inspection level and the fact that only a time window of traffic data is stored for classification, makes this approach specially suited for on-line traffic classification in network monitoring.

A main decision within this approach is the similarity or closeness functional considered in NN for flows pairing. Here, two types of functional are compared: those based on traditional distances and a parametric functional, referred to as similarity rule. The similarity rule has been designed from first principles by the authors, taking into account the general behavior of network protocols.

A. *Distance-based approaches*

Two distances have been considered: the normalized Euclidean distance in Eq. (1) and the Mahalanobis distance [3]:

$$d_M(\mathbf{x},\mathbf{y}) = \sqrt{(\mathbf{x}-\mathbf{y})^t \cdot \mathbf{\Sigma}^{-1} \cdot (\mathbf{x}-\mathbf{y})} \qquad (2)$$

where $\mathbf{\Sigma}$ stands for the covariance matrix. The difference between normalized Euclidean and Mahalanobis distances is that in the latter the weight of the eigenvalues of the covariance matrix in the resulting distance are normalized. This may be convenient when eigenvectors of low variance (low eigenvalue associated) contain relevant information for classification.

B. *Similarity rules*

The most similar flow to a given one can be found as the one which maximizes a similarity functional. The proposed parametric definition of the similarity functional for a pair of flows is the following:

$$F = |N_{IP} - 1| + \frac{1}{d_{p1} + k_1} + \frac{1}{d_{p2} + k_1} + \frac{1}{d_t + k_2} \qquad (3)$$

where N_{IP} is the number of coincident IPs between the two flows, which is at least 1 (Recall that at least one coincident IP is assumed for flows pairing), d_{p1} and d_{p2} are the 1-norm distances between ports (ordered according to the coincident IP), measured in tens of ports, d_t is the 1-norm distance between time stamps at the beginning of the flow (first packet), measured in seconds, and k_1 and k_2 are the functional parameters.

The definition of the functional in Eq. (3) answers to the behavior of typical network protocols. Thus, servers typically use one or a reduced number of ports to accept service requests. Also, the Operating Systems in the clients typically use consecutive dynamic port numbers for the consecutive connections established. For example, this would be the behavior of a web client when connecting to a number of web pages. In particular, if these pages are hosted in the same server, the flows share the same two IPs and close ports. Finally, related flows should be close in time. Eq. (3) has been designed so that close ports and time stamps have a significant impact on the functional but it is not so much penalized by large distances. For this, 1-norm distances

 ISBN: 978-1-61208-173-1

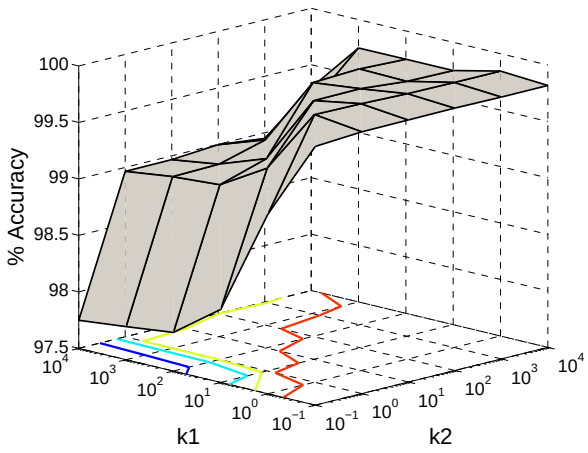


Figure 3. Parameters fitting for the similarity functional using the calibration data. Parameters k_1 and k_2 take values between 0.1 and 10.000 and are presented in a logarithmic scale

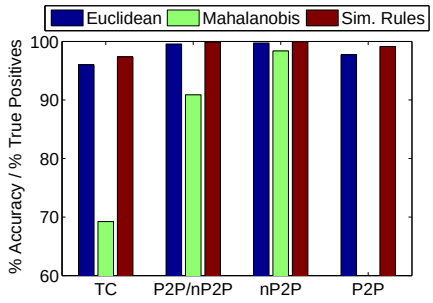
are considered instead of 2-norm distances, and they are included in inverse form in the functional.

The definition of the functional is also convenient from the practical point of view. The five variables (2 IPs, 2 ports and beginning time stamp) considered in the functional are obtained from the first packet in a flow. Therefore, one single packet is enough for flows pairing. Unlikely, distance-based approaches with the feature vector in Table II can only be applied once the flows have finished.

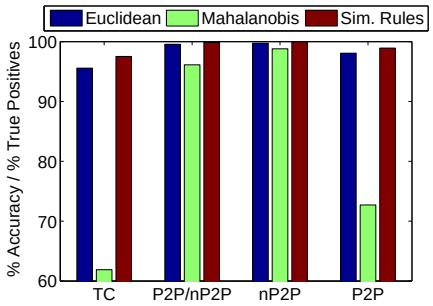
The calibration data will be used to fit the parameters of the similarity functional. Figure 3 shows the result of the calibration for k_1 and k_2 values between 0.1 and 10,000, in logarithmic scale. According to the results, the parameters are set to $k_1 = 1$ and $k_2 = 1$. It should be noted that the results are quite stable for a large interval of the parameters. In particular, the time closeness (k_2) does not seem to be relevant or even positive for certain values of k_1.

VI. Comparison

This section is devoted to compare the performance of distance-based approaches and similarity rules for flows pairing. The accuracy of each approach is defined as the percentage of flow pairs belonging to the same class. Although this work is focused on P2P classification, the pairing strategy can be used for traffic classification in general. This accuracy for traffic classification and P2P traffic identification for the calibration and test data-sets is presented in Figure 4. Notice that all the calibration decisions, such as the normalization in Euclidean distance, the covariance in the Mahalanobis distance, and the values of k_1 and k_2 parameters in the similarity rules, are set from the calibration data and then applied to the test data.



(a) Calibration data



(b) Test data

Figure 4. Comparison of strategies for flows pairing in terms of the coincidence of classes within a pair. Percentage of accuracy in traffic classification (TC) and P2P traffic identification (P2P/nP2P) and percentage of true positives in no P2P (nP2P) and P2P (P2P) traffic. The percentage of true positives of P2P traffic in the calibration data is 13%.

Figure 4 shows that the similarity rules outperform the other two approaches, being the Mahalanobis distance the worst choice. Figure 5 shows the first 30 eigenvalues of the covariance matrix in the normalized calibration data. The first four eigenvalues contain more than the 50% of the variability within the data, which evidences the collinearity of the variables considered in the feature vectors (Table II). The Mahalanobis distance normalizes the weights of the eigenvectors in the distance. This is negatively affecting the performance, showing that the eigenvectors of highest eigenvalue associated contain the relevant similarity information for classification. This is also convenient from the practical point of view, since it means that the useful similarity information is manifesting in a high number of variables. This result is coherent with those in [11]. In particular, the similarity information useful for classification is manifesting in the five variables considered in the similarity rules, which yield the best performance. This is especially convenient considering that a flow can be paired from the first packet using the similarity rules.

Finally, a comparison of the mean time between pairs has been performed for Euclidean-based pairs and similarity rules pairs. A t-test showed that this mean time is lower for

 ISBN: 978-1-61208-173-1

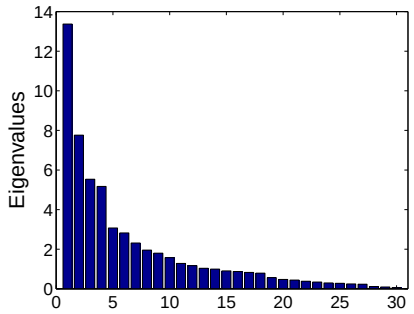


Figure 5. First 10 eigenvalues of the covariance matrix in the calibration data.

the similarity rules pairs (p-value $< 1^{-12}$) for both calibration and test data-sets. Pairs of flows with less difference in time are expected to be more reliable.

VII. Conclusion

This paper is devoted to introduce and compare different strategies for traffic flows pairing based on similarity measures. This strategy is used for fast P2P traffic classification in network monitoring, although it can be applied to traffic classification in general.

According to the results presented, flows pairing can be effectively performed using only five parameters for each flow: the IPs and port numbers and the beginning time stamp. These five parameters are combined in what has been named similarity rule. The pairing based on similarity rules outperforms the application of other traditional distances, such as the Euclidean distance, in several ways:

- The parameters in the similarity rule are available from the first packet in a flow, so that a flow can be paired only with the information in the first packet. Distance-based pairing needs the completeness of the flows.
- Similarity rules are faster to compute than distance-based pairing, since only 5 parameters are used. Also, they require less storage space.
- Classification based on similarity rules outperforms classification based on traditional distances.
- Similarity rules provide closer flow pairs in time than distance-based pairing.

Acknowledgment

Research in this paper is partially supported by the Spanish Ministry of Science and Technology through grant TEC2008-06663-C03-02.

References

[1] A. Callado, C. Kamienski, G. Szabo, B.P. Gero, and J. Kelner, "A Survey on Internet Traffic Identification," *IEEE Communications Surveys & Tutorials*, vol. 11, n. 3, pp. 37-52, 2009.

[2] S. Sen and J. Wang, "Analyzing Peer-to-Peer TrafficAcross Large Networks," *IEEE/ACM Transactions on Networking*, vol. 12, n. 2, pp. 219-232, 2004.

[3] A. Madhukar and C. Williamson, "A Longitudinal Study of P2P Traffic Classification", *Proc. of Int. Symposium on Modeling, Analysis and Simulation*, pp. 179-188, 2006.

[4] R. Keralapura, A. Nucci, and C. Chuah, "A Novel Self-Learning Architecture for P2P Traffic Classification in High Speed Networks," *Computer Networks*, vol. 54, pp. 1055-1068, 2010.

[5] L. Xuan-min, P. Jiang, and Z. Ya-jian, *"A New P2P Traffic Identification Model Based on Node Status"*, In Int. Conference on Mangement and Service Science, pp. 1-4, 2010.

[6] X. Li and Y. Liu, "A P2P Network Traffic Identification Model Based on Heuristic Rules,". *Int. Conference on Computer Application and System Modeling*, vol. 5, pp. 177-179, 2010.

[7] W. JinSong, Z. Yan, W. Qing, and W. Gong, "Connection Pattern-based P2P Application Identification Characteristic," *Proc. of Int. Conference on Network and Parallel Computing Workshops*, pp. 437-441, 2007.

[8] M. Soysal and E.G. Schmidt, "Machine Learning Algorithms for Accurate Flow-Based Network Traffic Classification: Evaluation and Comparison," *Performance Evaluation*, vol. 67, n. 6, pp. 451-467, 2010.

[9] L. Jun, Z. Shunyi, L. Yanqing, and Z. Zailong, "Internet traffic classification using machine learning," *Second International Conference on Communications and Networking in China (CHINACOM'07)*, pp 239-243, 2007.

[10] Y. Lim, H. Kim, J. Jeong, C. Kim, T.T. Kwon, and Y. Choi, "Internet traffic classification demystified: on the sources of the discriminative power," *Proceedings of the 6th International Conference On Emerging Networking Experiments And Technologies (CoNEXT'10)*, 2010.

[11] F.J. Salcedo-Campos, J.E. Díaz-Verdejo, and P. García-Teodoro, "Multiple Vector Classification for P2P Traffic Identification", In Proc. of Int. Conference on Data Communications and Networking (DCNET), 2011.

[12] OpenDPI, 2011. Available at http://www.opendpi.org

[13] F. Lindgren, B. Hansen, W. Karcher, M. S. ostr om, and L. Eriksson, "Model validation by permutation tests: Applications to variable selection," *Journal of Chemometrics*, vol. 10, pp. 521–532, 1996.

[14] S. Wiklund, D. Nilsson, L. Eriksson, M. S. ostr om, S. Wold, and K. Faber, "A randomization test for pls component selection," *Journal of Chemometrics*, vol. 21, pp. 427–439, 2007.

 ISBN: 978-1-61208-173-1