# A Hybrid Model for Network Traffic Identification Based on Association Rules and Self-Organizing Maps (SOM)

Zuleika Nascimento Djamel Sadok Stênio Fernandes
Informatics Center
Federal University of Pernambuco - UFPE
Recife, Brazil
{ztcn, jamel, sflf}@cin.ufpe.br

***Abstract*—Considerable effort has been made by researchers in the area of network traffic classification, since the Internet grows exponentially in both traffic volume and number of protocols and applications. The task of traffic identification is a complex task due to the constantly changing Internet and an increase in encrypted data. There are several methods for classifying network traffic such as known ports and Deep Packet Inspection (DPI), but they are not effective since many applications constantly randomize their ports and the payload could be encrypted. This paper proposes a hybrid model that makes use of a rule-based model along with a self-organizing map (SOM) model to tackle the problem of traffic classification without making use of the payload or ports. The proposed method also allows the generation of association rules for new unknown applications and further labeling by experts. The proposed hybrid model was superior to a rule-based model only and presented a precision of over 94% except for eMule application. The model was validated against a Measurement and Analysis on the WIDE Internet (MAWI) trace and presented true positive results above 99% and 0% false positives. It was also validated against another model based on computational intelligence, named Realtime, and the hybrid model proposed in this work presented better results when tested in real time network traffic.**

***Keywords-Association Rules; Self-Organizing Maps; Network Traffic Measurement; Genetic Algorithms.***

## I. INTRODUCTION

In recent years, the research effort toward network traffics identification has been growing [1] [2] [3] [4] [5]. As the Internet grows exponentially in both traffic volume and number of protocols and applications, it is essential to understand the composition of dynamic traffic characteristics to recognize protocols and applications which are often encrypted.

In this context, identifying traffic that passes over a network is a complex task, since access to the Internet is significantly increasing, bringing with it new users with different goals. Many peer-to-peer (P2P) applications are increasingly popular and accessible, such as eMule, Ares, and BitTorrent. The users behavior is also changing and the growth of streaming video services is notable [5], since Skype, MSN (Messenger), and other instant message services, along with sites that allow their users to upload and share videos in digital format, have become commonplace. To bring to the experts attention what passes through a network is an increasingly important activity.

There are several methods for classifying network traffic as known ports and Deep Packet Inspection (DPI) [6] [7]. The classification method based on ports performs an analysis of port numbers and is employed to identify applications or protocols. This technique proves to be quite ineffective, since most of the applications make use of random ports. The payload inspection technique or DPI, in turn, eliminates the problem of using random port number used for a specific application or protocol. The technique works starting with a classifier that extracts the payload from TCP/UDP packets and scans each packet in search of signatures that can identify the flow type. However, this technique does not work correctly in encrypted traffic data.

Recently, some methodologies have been investigated as network traffic classification tools. The work presented in [8] demonstrates the use of data mining techniques to classify flow and user behavior profiles. In order to classify the network traffic, the clustering k-means algorithm is used and compared to other model-based clustering methods along with rule-based classification models. Associations were found among flow parameters for several protocols and applications, such as Hypertext Transfer Protocol (HTTP), Mail, Simple Mail Transfer Protocol (SMTP), Domain Name System (DNS) and Internet Relay Chat (IRC). However, the variables used were source port, destination port, source IP address and destination IP address, and they may not be efficient when this technique is used for applications that enable obfuscation techniques or which are constantly changing pairs, IP addresses and random generations of ports number (e.g., eMule, BitTorrent, and Gnutella).

Bar-Yanai et al. [3] proposed a methodology based on a hybrid combination of two machine learning algorithms - K-Nearest Neighbor (KNN) [9] and K-Means [10], but this method works only with prior knowledge of the number of analyzed applications, i.e., the number of formed groups. Some works [1] [2] [4] [6] [7] [8] [11] do traffic classification based on port number, payload, or even the use of machine learning algorithms. Some of these works [1] [8] [7]

 ISBN: 978-1-61208-256-1 

exhibit signatures or association rules, resulting in extracted patterns. However, as already explained, these methods are not efficient for encrypted data and when applications make use of random ports. Furthermore, few studies validate their methods in real time.

Thus, this work presents a methodology to extract patterns automatically, and to identify and classify network traffic in real time without performing payload inspection and without the need for known ports. So, the paper proposes a real time hybrid model that combines the algorithm that automatically generates the association rules with a self-organizing maps (SOM) model to characterize the traffic. The model uses a machine learning algorithm based on Apriori [12] for the automatic extraction of association rules, choosing the most representative rules for each type of traffic. One characteristic of association rules is that these are easy to understand, unlike complex mathematical models. With the generated rules, the expert identifies the patterns more easily, assisting in decision making, e.g., blocking rules in firewalls. The model also uses an algorithm based on SOM [13] to perform grouping of protocols and applications traffic, dividing it by similarity to help with rules generation and as a second classifier. Moreover, the proposed model is capable of grouping unknown traffic for future tagging by experts (e.g., new applications traffic).

This paper is organized as follows. In Section II, we briefly review the techniques used in this paper. Section III shows the proposed model methodology. Section IV presents the experiments and the analysis of the results. Finally, Section V concludes with final considerations.

## II. FUNDAMENTALS

The machine learning is a very promising approach for traffic classification, since classification using artificial intelligence techniques can be used to identify traffic data without relying on packet payload. To deal with the analysis of huge network traffic data, machine learning techniques have been used as important tools for extracting association rules and performing traffic classification.

### A. Self-Organizing Maps (SOM)

Self-Organizing Maps (SOM) or Kohonens Self-Organizing Map is a clustering technique and data visualization technique that uses neural networks. These were based on observations of brain behavior where the unsupervised training is predominant.

The main reason for using SOM networks is to group similar input data into classes or groups called clusters [13]. What distinguishes SOM networks from others is a structure of two layers: one input and one output as shown in Fig. 1. The output layer is formed by a grid of neurons connected only to its immediate neighbors where, in this example, there are j input neurons and 15 output neurons.

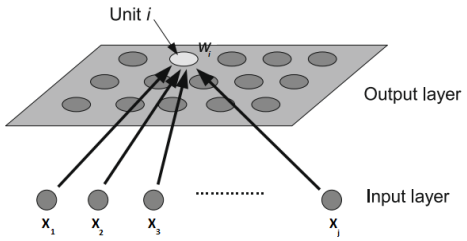


Figure 1. Self-Organizing Maps (SOM). (figure adapted from [14])

SOM networks produce a topological mapping and are based on competitive learning. In competitive learning, an objective function is required to determine the winner neuron, that is, a neuron which weight vector are nearest to the input vector. One of the key metrics used to determine the winner neuron is the Euclidean distance. The Euclidean distance is given by (1):

$$d_{xw} = \sqrt{\sum_{j=1}^{n}(x_j - w_{ij})^2}, \quad (1)$$

where $d_{xw}$ is the Euclidean distance, $x_j$ is the input sample, $w_{ij}$ is the synaptic weights connecting the input to neurons of the output grid and $n$ is the number of inputs. The Euclidean distance allows calculating the similarity between the input samples and the set of neurons weights.

The competitive training can be done in two ways: The winner-take-all and the winner-take-quota. In winner-take-quota training, used in this study, both winner weight and its neighbors are readjusted, as shown in (2), according to a neighborhood region.

$$w_{ij}(new) = w_{ij}(old) + \alpha.h_i^v.(x_j - w_{ij}^v(old)), \quad (2)$$

This region is defined by the function which is centered on the winner neuron and is expressed according to (3). There are several formats of neighborhood, including the circular and rectangular. In (3) a Gaussian function is defined. Over the iterations, it is necessary to reduce the size of the neighborhood region for the algorithm to converge.

$$h_i^v(n) = e^{-[\frac{d_i^v}{2r^2}]} \quad (3)$$

The function $h_i^v(n)$ is the neighborhood value between a neighbor neuron $i$ and the winner neuron $v$, $d_i^v$ is the Euclidean distance between the winner neuron $v$ and the excited neighboring neuron whose weights will be readjusted and, finally, $r$ is the width of the neighborhood function. A learning rate for the SOM training is defined by $\alpha$.

### B. Association Rules

Association rules show how the occurrence of an item set implies the occurrence of some other distinct item set in records of the same database. The main objective of

association rules is to find items that occur simultaneously and often in large transaction databases, facilitating the understanding of data behavior, since the use of association rules techniques make it possible to predict not only the class but any other attributes, like, for example, a network traffic behavioral profile.

An association rule is represented as follows:

$$X(antecedent) \Rightarrow Y(consequent) \quad (4)$$

Agrawal et al. [12] presented a mathematical model in which parameters such as support and confidence are taken into account. The support corresponds to the frequency of the occurrence of patterns throughout the database, while confidence is a measure of the force of rules, which indicates the frequency at which items in Y appear in occurrences containing X.

Analysis by means of association rules using Apriori has been studied and applied in a variety of fields such as web mining [15] and intrusion detection systems [16]. Apriori is able to find all frequent itemsets of any database and subsequently generate association rules. The algorithm, in fact, is based on two main subtasks: The frequent itemsets generation and the generation of rules themselves.

Machine learning algorithms can be used for grouping and extraction of patterns of network traffic, which can be used, after being trained, to classify traffic. Therefore, the advantage of using association rules is that associations between different databases attributes can be explored in the task of network traffic patterns extraction.

### C. Information Gain

Before extracting the rules, the stored data must be evaluated according to their relevance by measuring the information gain shown in (5). Information gain is a measure based on the entropy of a system, that is, on the disorder degree of a system. This measure indicates to what extent the whole systems entropy is reduced if we know the value of a specific attribute. Thus, it can show us how the whole system is related to an attribute; in other words, how much information this attribute contributes to the system [17]. The information gain is then used in the process of feature selection.

$$\triangle info = I(parent) - \sum_{j=1}^{k} \frac{N(V_j)}{N} I(V_j) \quad (5)$$

## III. Proposed Model

The proposed techniques in the literature cover several models to deal with traffic identification and pattern extraction. However, the number of new Internet applications increases at a high speed, and to extract patterns and identify applications is a complex task, especially when it comes to new applications done without the analysis of the payload and done in real time. To deal with this problem, this paper presents a hybrid model for traffic classification based on Apriori and SOM algorithms.

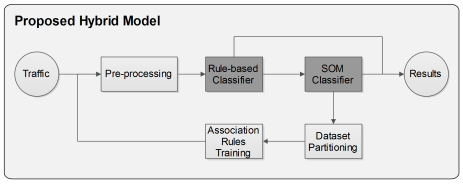


Figure 2. The proposed hybrid model.

### A. Architecture

The proposed model is divided into two main modules: a rule-based classifier and a second classifier named SOM Classifier, forming a hybrid model. In Fig. 2, we present an overview of the proposed hybrid model for automatic pattern extraction and real time traffic classification.

The proposed model objective is to generate association rules, i.e., knowledge from network traffic. One of the advantages of a rule-based classifier is that the generated knowledge is easily readable and understood by experts, unlike the case of complex mathematical models. The rules are used to classify traffic with low computational cost, without the need to perform complex calculations. In the proposed hybrid model of Fig. 2, the rule-based classifier was previously trained and is described in Section III-B2. In Fig. 2, after the data is pre-processed (Section III-B1), if traffic is known, the results are presented, and otherwise an SOM classifier is used to aid in traffic classification. The SOM model was previously trained as described in Section Section III-B3.

Unknown traffic can be subjected to the model. In this case, none of the classifiers would identify the traffic. In this scenario, a continuous and automatic process of training is performed. After being subjected to the two classifiers without success in classification, the unknown traffic is accumulated into a database, automatically labeled by the model. This label can be relabeled by an expert. The database is divided into similar statistical distributions, with the aid of a second SOM network which is trained in each cycle. Many clusters are generated with distinct datasets, where each dataset is trained by an Apriori algorithm, thus creating new rules for unknown traffic. At this moment, the expert takes action to label each generated cluster. In a new capture cycle, the traffic that was unknown would be known from an auto-labeling system or a definitive label after the expert analysis.

The analyzed applications and protocols were HTTP, HTTPS, FTP, SSH, Ares, Gnutella, eMule, BitTorrent and Skype. The defined model was implemented in Java using the Jpcap API and some Weka API classes. Furthermore, the Java Kohonen Neural Network Library (JKNNL) was used

 ISBN: 978-1-61208-256-1 

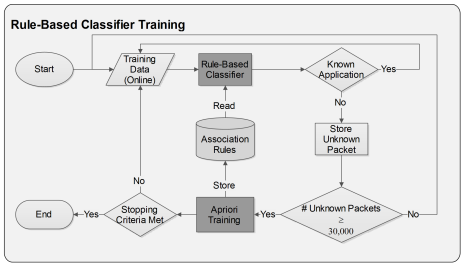


Figure 3. Rule-based classifier training process.

as reference [18].

### B. Training Process

The assembly of the proposed model is explained in the next subsections.

*1) Pre-processing:* The datagram header has several fields that could be used in the network traffic training and classification process. But not all fields are relevant for association rules generation. Thus, an algorithm based on information gain [19] was used to refine the most significant attributes to be used in the proposed hybrid model.

The most relevant attributes determined by the information gain algorithm were: DataLength (referring to the datagram length field), Flag (associated with the flags field), Lay (associated with the upper layer protocol field), TcpLength (relative to the packet payload size (TCP)) and SegmentLength (relative to the packet payload size (UDP)), totaling five attributes. The attributes TcpLength and SegmentLength were calculated since they are not part of the header. Note that neither payload nor port information is used in the process,. The traffic was pre-processed so that only these five attributes were used as input data. In the training of the SOM model (Section III-B3), the duplicated records were removed from the database and then all dataset was normalized within range [0.15,0.85]. The process of database normalization improves the effectiveness and performance of computational intelligence algorithms.

*2) Rule-based Classifier:* The rule-based classifier used in the proposed hybrid model in Fig. 2 was generated from the training process of Fig. 3. The training was conducted using real time traffic as input (online). The entire process was repeated for each application and protocol, ensuring that only particular application traffic would be trained at any time.

At first, the rules database is empty, and therefore, the application is labeled as unknown. After storing 30,000 unknown packets, new rules are generated to classify this initial unknown traffic. As it is ensured that what is passing through the network traffic belongs to the application that is being trained at the moment, the rules are labeled according to the corresponding application. Each passage through the Apriori training stage corresponds to a training cycle. The training process ends when the result of the classification accuracy rate (see Table I) and the number of rules in the database does not increase after 10 consecutive cycles.

TABLE I. METRICS USED IN THE EXPERIMENTS

| Metric | Equation |
|---|---|
| Accuracy (AC) | $AC = \frac{AD}{D}$<br>AD = number of packets correctly detected.<br>D = number of packets detected. |
| Correctness (CR) | $CR = \frac{AD}{A}$<br>AD = number of packets correctly detected.<br>A = real number of packets of the desired application. |
| Completeness (CP) | $CP = \frac{D}{A}$<br>D = number of packets detected.<br>A = real number of packets of the desired application. |
| False Negative Rate (FN) | $FN = \frac{FN}{A}$<br>FN = number of packets of a desired application that was not detected.<br>A = real number of packets of the desired application. |
| False Positive Rate (FP) | $FP = \frac{FP}{D}$<br>FP = number of packets incorrectly detected as a desired application.<br>D = number of packets detected. |

After the training process execution for all applications and protocols used in this work, a single rules database is created by unifying all rules databases found, which is then used as the rules database on the rule-based classifier on the proposed hybrid model (Fig. 2). The Apriori algorithm generates a huge number of rules, so it was necessary to realize a filter process to determine the most relevant rules. Relevant rules were defined as the rules with the greatest number of used attributes in their composition that represent an application or protocol.

The parameters used in the model were support and confidence. Many parameter values were tested during the training process, and the best results were obtained when using 30% for support and 90% for confidence.

*3) SOM Classifier:* The SOM model used in the proposed hybrid model of Fig. 2 was generated from the SOM network training according to Fig. 4. An offline and stratified dataset with 100,000 packets was used during the training for each application and protocol and was randomly mixed. Several trainings were conducted while varying the SOM network parameters until the largest number of representative clusters formed was obtained, since the SOM network used in this work has the functionality of increasing the grid size dynamically. A cluster is defined as representative when a particular application has predominance over others in 70% of the database. Each cluster represents a particular application or protocol.

As already mentioned, during the training process, diverse parameters of the SOM network were tested. The parameters

 ISBN: 978-1-61208-256-1 

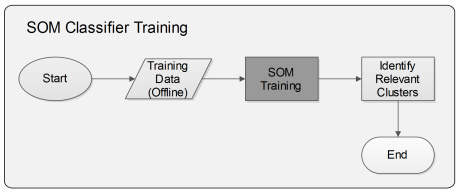


Figure 4. SOM classifier training process.

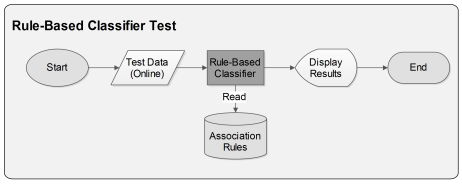


Figure 5. Rule-based classification process.

were the number of iterations and the radius used by the Gaussian neighborhood function. The sets used during the training were [5000, 10000, 15000] and [100, 150, 200] for the iterations and radius parameters respectively. The best results were obtained when using 15000 for the number of iterations and 100 for the radius. The learning rate was fixed in 0.1. The initial weight vector was randomly initialized with values between [0, 0.85].

## IV. Experimental Results

This section introduces the experimental results when applying the model against some test datasets and an analysis is performed when the model is compared to another network traffic classifier also based on computational intelligence.

### A. Metrics

The described models in Section III-A were measured using the metrics described in Table I.

In the next subsections, we present the obtained results for each metric in order to compare the proposed hybrid model in Section III-A with a rule-based model only described in Section III-B2. We also investigate the results on classifying commonly used public traces like the Measurement and Analysis on the WIDE Internet (MAWI) [20] database with the proposed model. Besides that, a comparative investigation between the model proposed by [3] and the proposed hybrid model described in this work was performed.

### B. Dataset

The dataset used in the experiments was generated by exclusively running the protocols and applications investigated in this work, assuring the collection of the ground-truth data. The experiments were performed in real time and it was ensured that only a determined application or protocol was being captured at that time, therefore, the data are submitted without label (unknown data) to the models analyzed in the next subsections. The dataset contains an average of 1,000,000 packets for each application. The packets were pre-processed to extract only the five attributes, serving as input to the classifiers. For the SOM Classifier, the data was normalized. The experiments were performed at the Point of Presence of Pernambuco (PoP-PE) of the National Network on Teaching and Research (RNP).

TABLE II. Rule-based Model Results

| Aplications / Protocols | AC % | CR % | CP % | FN % | FP % |
|---|---|---|---|---|---|
| BitTorrent | 95.98 | 89.35 | 93.09 | 4.07 | 0.23 |
| eMule | 87.86 | 76.88 | 87.50 | 3.95 | 0.46 |
| Ares | 94.01 | 82.53 | 87.78 | 5.25 | 0.90 |
| Gnutella | 99.82 | 89.10 | 89.25 | 0.16 | 0.00 |
| HTTPS | 99.98 | 98.46 | 98.48 | 0.02 | 0.00 |
| SSH | 99.96 | 98.91 | 98.95 | 0.03 | 0.00 |
| FTP | 99.93 | 99.91 | 99.97 | 0.06 | 0.00 |
| HTTP | 99.96 | 99.17 | 99.20 | 0.03 | 2.14 |
| Skype | 53.16 | 53.02 | 99.72 | 46.70 | 3.73 |

The proposed hybrid model was also validated against a public database (MAWI). The trace was randomly selected, comprising traces collected in June 8, 2012. The comparative investigation between the model proposed by [3] and the hybrid model proposed in this work was performed using approximately 12GB real time network traffic.

### C. Rule-based Model

The rule-based model was first validated as described in Fig. 5. The association rules were previously generated from the training phase (Section III-B2). The dataset was classified by the generated rules and the results were analyzed.

The results are shown in Table II. This shows the obtained results for the evaluated applications and protocols, with their respective rates. For example, for the BitTorrent application, using the found rules, an accuracy of 95.98% was obtained, with a correctness of 89.35%, completeness of 93.09%, false negatives of 4.07% and 0.23% false positives. Thus, with the found rules, it was possible to classify approximately 96% of the traffic generated by the BitTorrent application correctly. False positives presented low values for all applications and protocols, except for Skype and HTTP. The high false positive rate of HTTP was due to the fact that P2P applications such as eMule, Ares and Gnutella accessed web servers during the test. When an application has its datagram size varying in very short intervals, such as Skype, the model does not provide a high accuracy rate, because it does not work with such intervals.

### D. Proposed Hybrid Model

Due to the bad results obtained by the Skype application when evaluating the rule-based model, we proposed a hybrid model with the aid of a SOM network model so that the

 ISBN: 978-1-61208-256-1 

TABLE III. PROPOSED HYBRID MODEL RESULTS

| Aplications / Protocols | AC % | CR % | CP % | FN % | FP % |
|---|---|---|---|---|---|
| BitTorrent | 94.41 | 91.37 | 96.77 | 4.50 | 0.03 |
| eMule | 91.52 | 76.23 | 83.29 | 7.06 | 0.25 |
| Ares | 95.98 | 92.32 | 95.83 | 2.08 | 0.35 |
| Gnutella | 99.91 | 98.13 | 98.25 | 0.02 | 0.00 |
| HTTPS | 99.61 | 99.59 | 99.97 | 0.00 | 0.00 |
| SSH | 99.62 | 99.58 | 99.96 | 0.00 | 0.00 |
| FTP | 99.78 | 97.74 | 97.96 | 0.00 | 0.00 |
| HTTP | 99.70 | 99.58 | 99.88 | 0.22 | 0.00 |
| Skype | 94.97 | 86.66 | 90.25 | 3.58 | 0.00 |

TABLE IV. RESULTS OBTAINED FROM MAWI DATABASE

| Protocols | TP % | FP % |
|---|---|---|
| SSH | 99.73 | 0.00 |
| FTP | 99.64 | 0.00 |
| HTTP | 100.00 | 0.00 |

results could be compared. A SOM network reduces the dimensionality of the data and assists in pattern extraction.

The hybrid model was discussed in Section III-A and the results obtained in validation, i.e., using the dataset described in Section IV-B are displayed in Table III. Notice that the results of accuracy on the Skype application proved to be superior in the hybrid model (94.97%) against a 53.16% (see Table II) rate when the rule-based model is used. Besides that, the results for other applications was superior or equivalent. Rates of FP were better tackled by the proposed hybrid model in all applications and protocols, while FN rates were superior in 67% of all protocols and applications investigated.

The hybrid model performance was also evaluated against public traces from MAWI. The FTP, SSH and FTP protocols were analyzed, since these are the protocols in common with the protocols used in this work. The results were evaluated by two metrics and are shown in Table IV. The first metric is the true positive (TP), which is used to measure the traffic fraction of a certain application that is recognized by the model for this application. Also, the false positive (FP) was used, which measures the traffic fraction that does not belong to a certain evaluated application. The classification rate results exceeded 99% of true positive rate. No false positives were detected during the test.

Besides that, an investigation was performed to determine the effectiveness of the proposed hybrid model (PHM) against the model proposed by [3] (RTM), since it also uses some well-known computational intelligence techniques to classify network traffic data. The experimental results, using the protocols, applications and metric (AC (%)) in common with both works, are shown in TABLE V.

It can be observed that when both models were validated against a real time traffic scenario, our proposed hybrid model was far superior when compared to the model proposed by [3].

TABLE V. RESULTS FOR THE REALTIME MODEL (RTM) [3] AND THE PROPOSED HYBRID MODEL (PHM)

| Applications / Protocols | RTM | PHM |
|---|---|---|
| HTTP | 93.75 | 99.7 |
| Skype | 90.54 | 94.97 |
| eMule | 94.57 | 97.4 |
| BitTorrent | 84.55 | 94.41 |

## V. CONCLUSION

This work proposed a hybrid model based on computational intelligence techniques, consisting of a rule-based model and a self-organizing map (SOM) model. The model was proposed to deal with the problem of encrypted data, the constantly changing ports behavior of some applications and the identification of new protocols or applications (unknown network traffic) for future labeling by experts, since the architecture has the ability of being in a constant learning process to extract patterns from new applications or protocols. Besides that, the aim of the method is to extract association rules for a network traffic classification purpose, since rules are easily understood by experts and can be easily applied, for example, in a firewall system for security purposes.

The experimental results showed that the proposed hybrid model is superior to a rule-based model only, with results that exceed 91% of precision, maintaining low rates of false positives and false negatives for most applications and protocols. The hybrid model was also able to better tackle the problem of Skype identification, which presented bad results when classified by a rule-based model only. It was also validated against a known public network traffic database known as Measurement and Analysis on the Wide Internet (MAWI). The proposed model reached levels superior to 99% for true positive rates and 0% for false positive rates for the investigated protocols. The proposed hybrid model was also superior to the Realtime model (RTM) [3] when evaluated against a real time network traffic.

## REFERENCES

[1] G. Szabó, Z. Turányi, L. Toka, S. Molnár, and A. Santos, "Automatic protocol signature generation framework for deep packet inspection," in Proceedings of the 5th International ICST Conference on Performance Evaluation Methodologies and Tools, Brussels, Belgium, May 2011, pp. 291–299.

[2] V. Carela-Español, P. Barlet-Ros, M. Solé-Simó, A. Dainotti, W. de Donato, and A. Pescapé, "K-dimensional trees for continuous traffic classification," in Proceedings of the Second international conference on Traffic Monitoring and Analysis, ser. Lecture Notes in Computer Science, F. Ricciato, M. Mellia, and E. Biersack, Eds., vol. 6003. Berlin, Heidelberg: Springer Berlin Heidelberg, Apr. 2010, pp. 141–154.

 ISBN: 978-1-61208-256-1 

[3] R. Bar - Yanai, M. Langberg, D. Peleg, and L. Roditty, "Realtime classification for encrypted traffic," in Proceedings of the 9th international conference on Experimental Algorithms, ser. Lecture Notes in Computer Science, P. Festa, Ed., vol. 6049. Berlin, Heidelberg: Springer Berlin Heidelberg, May 2010, pp. 373–385.

[4] A. Dainotti, A. Pescape, and K. Claffy, "Issues and future directions in traffic classification," IEEE Network, vol. 26, no. 1, Jan. 2012, pp. 35–40.

[5] J. Summers, T. Brecht, D. Eager, and B. Wong, "Methodologies for generating HTTP streaming video workloads to evaluate web server performance," in Proceedings of the 5th Annual International Systems and Storage Conference on - SYSTOR '12. New York, New York, USA: ACM Press, Jun. 2012, pp. 1–12.

[6] G. La Mantia, D. Rossi, A. Finamore, M. Mellia, and M. Meo, "Stochastic Packet Inspection for TCP Traffic," in 2010 IEEE International Conference on Communications. IEEE, May 2010, pp. 1–6.

[7] M. Ye, K. Xu, J. Wu, and H. Po, "AutoSig-Automatically Generating Signatures for Applications," in 2009 Ninth IEEE International Conference on Computer and Information Technology, vol. 2. IEEE, 2009, pp. 104–109.

[8] U. K. Chaudhary, I. Papapanagiotou, and M. Devetsikiotis, "Flow classification using clustering and association rule mining," in 2010 15th IEEE International Workshop on Computer Aided Modeling, Analysis and Design of Communication Links and Networks (CAMAD). IEEE, Dec. 2010, pp. 76–80.

[9] T. Cover and P. Hart, "Nearest neighbor pattern classification," 1967, pp. 21–27.

[10] J. Hartigan and M. Wong, "A k-means clustering algorithm," 1 ACM/IEEE-CS Joint Conference, Applied Statistics, vol. 28, 1979, pp. 100–108.

[11] M. Soysal and E. G. Schmidt, "Machine learning algorithms for accurate flow-based network traffic classification: Evaluation and comparison," Performance Evaluation, vol. 67, no. 6, Jun. 2010, pp. 451–467.

[12] R. Agrawal and R. Srikant, "Fast Algorithms for Mining Association Rules in Large Databases," Sep. 1994, pp. 487–499.

[13] S. Haykin, "Neural Networks: A Comprehensive Foundation," Jul. 1998.

[14] E. Kita, S. Kan, and Z. Fei, "Investigation of self-organizing map for genetic algorithm," Advances in Engineering Software, vol. 41, no. 2, Feb. 2010, pp. 148–153.

[15] J. S. Ryu, W. Y. Kim, K. I. Kim, and U. M. Kim, "Mining opinions from messenger," in Proceedings of the 2nd International Conference on Interaction Sciences Information Technology, Culture and Human - ICIS '09. New York, New York, USA: ACM Press, Nov. 2009, pp. 287–290.

[16] L. Li, D.-Z. Yang, and F.-C. Shen, "A novel rule-based Intrusion Detection System using data mining," in 2010 3rd International Conference on Computer Science and Information Technology. IEEE, Jul. 2010, pp. 169–172.

[17] M. Martín-Valdivia, M. Díaz-Galiano, A. Montejo-Raez, and L. Ureña López, "Using information gain to improve multimodal information retrieval systems," Information Processing & Management, vol. 44, no. 3, May 2008, pp. 1146–1158.

[18] JKNNL, "Java Kohonen Neural Network Library." [retrieved: March, 2013]. Available: http://jknnl.sourceforge.net/

[19] P.-N. Tan, M. Steinbach, and V. Kumar, "Introduction to Data Mining, (First Edition)," May 2005.

[20] MAWI, "MAWI Working Group Traffic Archive." [retrieved: March, 2013]. Available: http://mawi.wide.ad.jp/mawi/

 ISBN: 978-1-61208-256-1 