Improving P2P Network Traffic Classification with ML multi-classifiers

Haitham A. Jamil, Bushra M. A, Ahmed Abdalla, Ban M. K,Sulaiman M. Nor, Muhammad N. Marsono

Department of Electronic and Computer Engineering, FKE, UTM
Skudai, Johor, Malaysia

Abstract— Machine learning (ML) techniques have been known to be a promising method to classify Internet traffic. These techniques have the capability to detect encrypted communication and unknown traffic. However, the generation of examples, feature selection and classifier design have a significant impact in classification results. This paper proposes approach based on multiple ML classifiers in order to provide a robust model for online P2P Internet traffic classification. The process of validation and analysis were done through experimentation on traces captured from Universiti Technologi Malaysia. The results show that the generation of the training model using ML on P2P classification resulted in a high accuracy, low false negative and low classifying time.

Keywords— P2P traffic, traffic classification, Machine Learning, Multi-classifiers.

I. INTRODUCTION

With the recent evolution of Internet to content distribution oriented architecture, computer communications have gradually migrated from the client-server paradigm to the edge services paradigm, and more recently to the Peer-to-Peer (P2P) computing model. This evolution of the new network paradigms and applications has changed the traffic characteristics of the network. Traffic classification plays a vital role in monitoring the performance and ensuring fairness of the network[1]. The management of bandwidth in a heterogeneous network with limited bandwidth, as in campus networks, is becoming more challenging. Therefore, traffic detection and mitigation are powerful tools used to improve the network performance[2].

Research work on applying statistical Machine Learning in on-line P2P traffic classification is still lacking. Changes in traffic properties will result in variation of the traffic performance. The performance of these classifiers not only depends on the different ML algorithms, but also on the features selected. The performance also depends on the generation of accurate samples of the training portion[3].

This paper proposes approach based on multiple ML classifiers in order to provide a robust model for on-line P2P Internet traffic classification. The objectives of this paper are to maintain and evaluate the effectiveness of the P2P classifier practically in term of accuracy, to measure the efficiency of the classifier practically in term of cost and to validate the classifier output against recent traffic traces.

The remainder of this paper is organized as follows. Section two introduces some related works on the detection and mitigation of P2P network traffic. Section three describes the methodology. The experimental setup is discussed in section four. The experimental results and analysis are given in section five. We conclude the work in section six..

II. MACHINE LEARNING AS A PROMISING STATISTICAL-BASED CLASSIFICATION

Machine learning is a branch of Artificial Intelligence (AI). It has been known as a collection of powerful techniques for data mining and knowledge discovery [4]. Arthur Samuel (1959) defined Machine Learning as: "*Field of study that gives computers the ability to learn without being explicitly programmed*" and in [5] Witten and Frank noted "*Things learn when they change their behaviour in a way that makes them perform better in the future*".

In 1994, Jeremy Frank used ML for intrusion detection [6]. It was the first time to utilize ML for Flow based Internet traffic classification. Basically, machine learning technique involves two steps, learning data samples to generate machine learning model and classifying future data samples using the generated model.

A. ML and concept drift

Concept drift is a field of data mining that has been gaining considerable attention. Wang in [7] described the concept drift in machine learning as "The term concept refers to the quantity that a learning model is trying to predict, i.e., the variable. Concept drift is the situation in which the statistical properties of the target concept change over time". Concept drifts can be characterized in different ways. One is by the speed of change in learning algorithm, and another is the reason of change. Moreover, concept drift can be characterized into virtual concept drift which is drift in data distribution and real concept which is drift in decision concepts [8, 9].

Learning under concept drift poses an additional challenge to existing learning algorithms. Instead of considering all the past training data, or making a permanent distribution hypothesis, a robust learner should be capable to follow these changes and immediately adapt to them. Otherwise, as concept drifts, the induced model may not be relevant to the new data, which may result in an increasing number of

errors. The issue of concept drift refers to the change of distribution underlying the data [10].

Based on the literature, Most of the current traffic detection methods use ML techniques to classify the traffic using flow statistics. However, the classification accuracy mostly fluctuates. The authors in [11-18] discussed the issue of supervised ML algorithms which requires the training data to be identified first before a model could be used for the testing set.

Zarei et al. [12] proposed a retraining process for P2P ML classifier. It uses training dataset generated by the three classes heuristic to create and retrain on-line ML classifier. The overall results shows that the training dataset generation can generate accurate training dataset by classifying P2P flows with high accuracy and low false positive

There are some weaknesses on using statistical classification in P2P traffic classification. The classification accuracy becomes low over time as the traffic behaviour changes. Presently, most of the researchers focus on the quality of generating samples to be used as input but, the Peer-to-Peer detection using machine learning classification is also influenced by its training quality. More so, recent achievements in P2P traffic classification focus on the evaluation of the training and testing data while the evaluation of the validation data is not put into account.

B. *Machine learning and Snort*

SNORT is a free and open source network intrusion detection system (NIDS), created by Martin Roesch in 1998. The difference between this mechanism and machine learning is the basic concept of detection method as illustrated in Figure 3.4. Machine learning needs a prior training of features to form the generative model. Referring to the Figure 3.4(a), once trained, the machine learning can recognize the exact or similar pattern of the future input based on the model and make the decision. However, SNORT monitors network traffic and analyse it against signatures set. The signatures of different types of network traffic protocol, including Transport Control Protocol (TCP), User Datagram Protocol (UDP) and Internet Control Message Protocol (ICMP) are expressed as SNORT rules as shown in Figure 3.4(b). These rules are generated by humans intervention normally after the outbreak of malware has occurred.

Compared to the machine learning in Figure 3.4(a), SNORT in Figure 3.4(b) utilizes an exact pattern P2P matching. According to the SNORT rules, traffic that contains P2P signatures is flagged. The detection engine takes the packet and checked it through a set of rules. If the rules match the data pattern in the packet, then they are sent to the alert processor for further observation. Otherwise, the packet is tagged as nP2P packet. Detection engine and rules set are controlled by a configuration script called snort.conf.

Weka is a collection of open source state-of-the art machine learning algorithms and data pre-processing tools [19]. Weka contains tools for data mining problems such as data pre-processing, classification, regression and clustering. Weka is well-suited for developing new machine learning schemes.

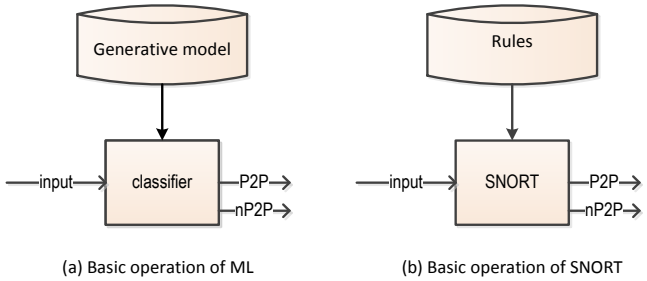

Fig 1 Basic operation for detecting P2P using ML and SNORT

C. *Brief Theoretical background on Machine learning algorithms*

This subsection provides a concise theoretical background for some of the machine learning algorithms. The algorithms are considered in such a way that can help to choose a proper model for an on-line Peer-to-Peer Internet traffic classification.

1) *Support Vector Machines*

SVMs are a group of supervised learning methods that generates input-output mapping functions from a set of labelled training data [20]. The mapping function can be either a classification function or a regression function. For classification, nonlinear core functions are often used to convert input data to a high-dimensional feature space in which the input data become more distinct compared to the original input space.

Maximum-margin hyper-planes are then created. The produced model depends on only a subset of the training data near the class boundaries. SVM is an effective method to solve the classification and pattern recognition problems [21]. This research uses SVM algorithm to classify and identify P2P traffic.

2) *J48 Decision Tree*

J48 is a Machine Learning algorithm. It makes decision trees from a set of training data examples, with the help of information entropy estimation. The training dataset consists of a wide number of training samples which are defined by various features and consists of the target class. J48 selects one perfect feature of the data at each node of the tree which is used to divide its collection of samples into subsets improved in one or another class. It is based upon the concept of normalized information gain that is obtained from selecting a feature for splitting the data.

The feature with the highest normalized information gain is selected, and a decision is made. After that, the J48 algorithm repeats the same action

on the smaller subsets. In the current research work, J48 algorithm is used for Internet traffic classification.

3) Artificial Neural Networks

ANNs have been used in several different fields like information parallel processing, pattern recognition, classification of Internet application, intrusion detection. The ANN model is trained using a set of traffic values associated with fully identified applications. Then, the trained ANN model is used to identify applications relative to new traffic values that are presented as inputs. Packet contents are not inspected and with particular extension this approach can be used for real time.

It is a fact that the training phase using ANN is computationally requiring, but once the classification models are conveniently trained, they can be used on-line and the computational requirements of the testing phase are extremely low [22].

$$f(x) = g(\sum_i v_i g\ (\sum_j w_{ij}\ +\ b_i b_0)) \quad (1)$$

This equationpresents the output computation of a two layered ANN, where x is the input vector, v_i is a weight in the output neuron, g is the activation function, w_{ij} is the weight of a hidden neuron and b_i, b0 is the bias.

D. Combining ML classifiers

A system combining two or more of different techniques is called a multi classifiers system which is supposed to achieve better accuracy than any single classifier. The machine learning community has recently developed multi classifier systems based on intelligent combination algorithms that learn from historical behaviours of individual classifiers on the studied flow objects [23]. Multi classifiers system is more robust which can understand the changing in the nature and mix of applications.

Researchers are only recently beginning to investigate more general and effective techniques [24] that use different classifiers on the same flow object. Although combining classifiers can increase the computational complexity of the process, it can also potentially reduce the amount of traffic information required for accurate classification for example, using five packets per flow rather than ten which can reduce the average classification time.

The authors in [25] proposed an on-line Internet application traffic classification system based on SVM. The proposed system combines a binary classifier SVM and Support Vector Data Description (SVDDs). This system includes three layers: First, the SVM layer which is a binary classifier that performs classification between P2P and non-P2P traffic. The second layer classifies P2P traffic into file sharing, messenger and P2P-TV. Third, individual classifier which is classifies the individual application traffic types. On other hand, the structure of this classifier includes four modules data collection module which is responsible for data preparations, features selection module which is responsible for selecting the optimal flow attributes, training module which performs the training based on the selected features for each classification and classification modules which is used the trained system to classify incoming flow data. The performance of the system is validated with experiments which confirm that its recall is 96.6% and precision is 95.8% when full set of features are used. Whilst, its recall is 98.27% and precision is 96.35% when Correlation Feature Selection algorithm (CFS) is applied.

III. METHODOLOGY

This section proposes methodologies for P2P traffic detection. Since we focus on online flow-based classification, privacy concerns is not an issue here. Figure 1 illustrates the proposed framework for on-line P2P traffic classification.

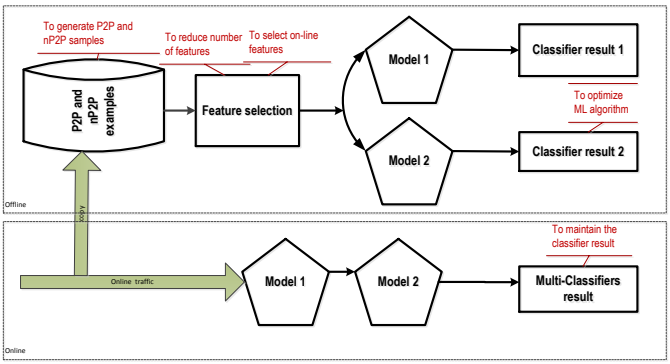

Fig 2 The structure of multi-classifiers system

A. Data pre-processing

Datasets used in this work were downloaded from specific shared resources. Also five datasets were captured from Universiti Teknologi Malaysia academic network. Table 1 to 4 illustrates the datasets consisting of the number of flow instances and size.

(1) The campus datasets were captured from both academic and colleges networks of University Teknologi Malaysia. The traces cover 1834122 packets (15365 flows) from different segments. Our traces consist of five datasets. The dataset one to three were captured between July and October 2011 using Wireshark [26]. Dataset four was captured in October 2012 using Tcpdump and analysing using Snort [27]. We used Snort to get accurate P2P traffic in such way to be used as a labelled data for our model. Dataset five is mix traffic captured in November 2012. This dataset is used for evaluation. Table 1 shows the captured datasets consisting of eMule, PPlive, HTTP, Bittorrent and the Mix network traffic.

TABLE 1 THE SAMPLES OF THE CAMPUS DATA SETS

Dataset	Application	Size
Dataset1	eMule	21.4 MB
Dataset2	PPlive	54.5 MB
Dataset3	HTTP	49.3 MB
Dataset4	BitTorrent	128.0 MB
Dataset5	Mix	588.0 MB

(2) CAIDA datasets [28] contain traces from both active and passive measurement of the Internet connection. The active-measurement measures the connectivity and latency using active probing. The passive-measurement is done in collaboration with organizations that operate network infrastructure in academic, non-profit, commercial, and dark address space to passively monitor traffic on the link. CAIDA provides access to these datasets for researchers in accordance with University of California, San Diego policy. We are allowed to download 2009's and 2013's datasets via secure login. Each flow is associated with the protocol identified by the destination port number. The TCP server port is a relatively reliable source of ground truth to identify the usual protocols. Tcptrace was used to generate the features from the first quarter of the flow [28].

TABLE 2 THE TRACES OF CAIDA DATA SETS

Dataset	Size
equinix-sanjose.dirA.20130117-125912.UTC.anon	897 MB
equinix-sanjose.dirA.20130117-130000.UTC.anon	1.11 GB
equinix-sanjose.dirA.20130221-130100.UTC.anon	703 MB

(3) Cambridge data sets [28] are based on the traces captured on the Genome Campus network in August 2003. They are published by the computer laboratory in the University of Cambridge. There are ten different data sets each from a different period of the 24-hour day [29]. The number of flows in each data set is different, due to a variable density of traffic during each constant period. These data sets cover most of the statistics of absolute TCP flows. Moreover, each flow example is high dimensional since it consists of 248 features that are derived from the TCP headers by using tcptrace [30].

TABLE 3 THE SAMPLES OF THE CAMBRIDGE DATA SETS

Dataset	Instances	Size
Dataset1	24863 flows	29.7 MB
Dataset2	23801 flows	28.3 MB
Dataset3	22932 flows	27.5 MB
Dataset4	22285 flows	26.6 MB
Dataset5	21648 flows	25.8 MB
Dataset6	19384 flows	23.1 MB
Dataset7	55835 flows	66.0 MB
Dataset8	55494 flows	65.6 MB
Dataset9	66248 flows	78.3 MB
Dataset10	65036 flows	77.1 MB

(4) UNIBS traces [31] include packets generated by a series of workstations, located at the University of Brescia (UNIBS) in Italy between September and October 2009. These traces were captured by Tcpdump on the edge router which connects the network to the Internet through a dedicated 100 Mbps uplink. The captured traces were saved as files on a dedicated hard disk that is connected to the router internals through a dedicated ATA controller. The traces occupy around 2.7 GB (78998 flows) which includes Web (61.2%), Mail (5.7%), P2P traffic (32.9%) and other protocols (0.2%).

TABLE 4 THE TRACES OF UNIBS DATA SETS

Dataset	Size
unibs20090930.anon	317 MB
unibs20091001.anon	236 MB
unibs20091002.anon	1.94 GB

B. Evaluation Metric

Benefit and cost are used to evaluate the effectiveness of the proposed approach. These metrics depend on true positive, false positive, true negative and false negative. TP is the number of P2P class that are correctly classified, FP is the number of nP2P class that are classified as P2P class, TN is the number of non-P2P (nP2P) class that are correctly classified, and FN is the number of P2P class that are classified as nP2P class. Training and testing times are used to illustrate the efficiency improvement.

IV. RESULTS AND DISCUSSION

In order to realize the P2P on-line traffic classification, P2P examples are generated using Snort and a feature subset is created using our algorithm in[3]. Then, we built an algorithm that can able to detect P2P over non P2P (nP2P) accurately.

Table 5 defines the classification performance of the proposed approach. The performance accuracy of the training part using individual model is 98.58% using decision tree and 98.00% using SVM, whilst it is 98.46% and 97.90% respectively for the testing. The accuracy of the multi classifier has shown significant

improvement. The accuracy is 99.35% for the training and 99.29% for the testing with error of 0.71%.

TABLE 5 THE EVALUATION RESULTS

Partition	Classifier	TP	FP
Training	SVM	98.43 %	1.57%
	J48	98.58 %	1.42%
	ANN	98.00 %	2.00%
	SVM +J48	99.9%	0.1%
Testing	SVM	98.31 %	1.69%
	J48	98.46 %	1.54%
	ANN	97.90 %	2.10%
	SVM +J48	99.72 %	0.28%

Algorithm 1 P2P classifier

```
INPUT
Let P = {p1, p2, ..., pn}; P is packet traffic dataset,
X = {F1, F2, ..., Fn}; X is flow info
D = {f1^k, f2^k, ..., fn^k, c}; D is labeled dataset k = 0 (offline) or 1 (online)
F = {f1^I, f2^I, ..., fn^I, c}; F is the online features
OUPUT
System sudo tcpdump tcp or udp; captured packet
for each packet (p) do
    Extract packet level information
    Read row
    Capture = file1; save capture data
end for
Sniff = file2; save sniffed data
open file1
for each row do
    Add class c; create labeled datase
    if the (IPsrc/IPdst) belong to existing flow then
        Add packet
    else
        read X Array to save flows
        New flow; split flow information
    end if
end for
Read D Array
Run algorithm1
Read F Array
Build J48,SVM model
Evaluate the model using multi algorithm
Read TP, FP, TN and FN
Accuracy = (TP+TN)/(TP+FP+TN+FN)
Print TP, FP, TN, FN and accuracy
```

V. CONCLUSIONS

The evolution of new network models and applications has changed the traffic properties of the network, so a better understanding of these applications is important especially for the purpose of network capacity planning and traffic engineering. In this paper, we propose a technique based on ML in order to provide a robust model for online P2P Internet traffic detection. We evaluate the using our construction classifier to detect P2P traffic in terms of effectiveness and efficiency. The experimental results indicate that construction classifier result in a higher accuracy and uses smaller detection time. The accuracy and testing time for the multi classifiers using SVM and decision tree are 99.29% and 2 second, respectively.

ACKNOWLEDGMENT

The authors would like to thank University of Cambridge, University of Brescia and CAIDA for providing their datasets. Also we acknowledge the support of University of Elimam Elmahdi.

REFERENCES

[1] H. A. Jamil, R. Zarei, N. O. Fadlelssied, A. M, S. M. Nor, and M. N. Marsono, "Analysis of Features Selection for P2P Traffic Detection Using Support Vector Machine," presented at the International Conference of Information and Communication Technology (ICoICT), Bandung, Indonesia, 2013.

[2] H. A. Jamil, Bushra M. Ali, Ghada A. A., Sulaiman M. Nor, and M. N. Marsono, "Detection and Mitigation Framework of Peer-to-Peer Traffic in Campus Networks," *International Review on Computers and Software (IRECOS)*, vol. 8, pp. 1734-1743, 31 July 2013 2013.

[3] H. A. Jamil, A. M, A. Hamza, S. M. Nor, and M. N. Marsono, "Selection of online Features for Peer-to-Peer Network Traffic Classification," in *Recent Advances in Intelligent Informatics*. vol. 235, ed: Springer International Publishing, 2014, pp. 379-390.

[4] T. T. Nguyen and G. Armitage, "A survey of techniques for internet traffic classification using machine learning," *Communications Surveys & Tutorials, IEEE*, vol. 10, pp. 56-76, 2008.

[5] I. H. Witten and E. Frank, *Data Mining: Practical machine learning tools and techniques*: Morgan Kaufmann, 2005.

[6] J. Frank, "Artificial intelligence and intrusion detection: Current and future directions," in *Proceedings of the 17th National Computer Security Conference*, 1994.

[7] S. Wang, S. Schlobach, and M. Klein, "Concept drift and how to identify it," *Web Semantics: Science, Services and Agents on the World Wide Web*, vol. 9, pp. 247-265, 2011.

[8] P. M. Gonçalves Jr and R. S. M. d. Barros, "RCD: A recurring concept drift framework," *Pattern Recognition Letters*, vol. 34, pp. 1018-1025, 2013.

[9] L. L. Minku, A. P. White, and X. Yao, "The impact of diversity on online ensemble learning in the presence of concept drift," *Knowledge and Data Engineering, IEEE Transactions on*, vol. 22, pp. 730-742, 2010.

[10] N. Lu, G. Zhang, and J. Lu, "Concept drift detection via competence models," *Artificial intelligence*, 2014.

[11] L. Bernaille, R. Teixeira, I. Akodkenou, A. Soule, and K. Salamatian, "Traffic classification on the fly," *ACM SIGCOMM Computer Communication Review*, vol. 36, pp. 23-26, 2006.

[12] R. Zarei, A. Monemi, and M. Marsono, "Retraining Mechanism for On-Line Peer-to-Peer Traffic Classification," in *Intelligent Informatics* vol. 182, ed: Springer Berlin Heidelberg, 2013, pp. 373-382.

[13] A. W. Moore and D. Zuev, "Internet traffic classification using bayesian analysis techniques," 2005, pp. 50-60.

[14] N. Williams, S. Zander, and G. Armitage, "A preliminary performance comparison of five machine learning algorithms for practical IP traffic flow classification," *ACM SIGCOMM Computer Communication Review*, vol. 36, pp. 5-16, 2006.

[15] T. Auld, A. W. Moore, and S. F. Gull, "Bayesian neural networks for internet traffic classification," *Neural*

Networks, IEEE Transactions on, vol. 18, pp. 223-239, 2007.

[16] Y. Ma, Z. Qian, G. Shou, and Y. Hu, "Study of information network traffic identification based on C4. 5 algorithm," 2008, pp. 1-5.

[17] J. Erman, M. Arlitt, and A. Mahanti, "Traffic classification using clustering algorithms," in *ACM SIGCOMM 2006 - Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication, September 11, 2006 - September 15, 2006*, Pisa, Italy, 2006, pp. 281-286.

[18] R. Zarei, A. Monemi, and M. N. Marsono, "Automated Dataset Generation for Training Peer-to-Peer Machine Learning Classifiers," *Journal of Network and Systems Management*, pp. 1-22, 2013.

[19] WEKA. (2012). Available: http://www.cs.waikato.ac.nz/ml/weka/

[20] SVM. (2012). *Support vector machines (SVM)*. Available: http://www.support-vector-machines.org

[21] R. Wang, Y. Liu, Y. Yang, and H. Wang, "A new method for P2P traffic identification based on support vector machine," *Artificial Intelligence Markup Language. Egypt: IEEE Computer Society*, pp. 58-63, 2006.

[22] A. Nogueira, P. Salvador, A. Couto, and R. Valadas, "Towards the On-line Identification of Peer-to-peer Flow Patterns," *Journal of Networks*, vol. 4, 2009.

[23] A. Dainotti, A. Pescapè, and K. C. Claffy, "Issues and future directions in traffic classification," *Network, IEEE*, vol. 26, pp. 35-40, 2012.

[24] A. Callado, J. Kelner, D. Sadok, C. Alberto Kamienski, and S. Fernandes, "Better network traffic identification through the independent combination of techniques," *Journal of Network and Computer Applications*, vol. 33, pp. 433-446, 2010.

[25] D. Park, "Real-time classification of Internet application traffic using a hierarchical multi-class SVM," *KSII Transactions on Internet and Information Systems (TIIS)*, vol. 4, pp. 859-876, 2010.

[26] Wireshark. (2010). Available: http://www.wireshark.org

[27] Snort. (2010). *SNORT Network Intrusion Detection System.* Available: http://www.snort.org

[28] CAIDA. (2013, 10 April 2013). *The Cooperative Association for Internet Data Analysis*. Available: http://www.caida.org/data

[29] Cambridge, data, and sets. (2012, 18 nov). Available: http://www.cl.cam.ac.uk/research/srg/netos/nprobe/data/papers/sigmetrics/index.html

[30] H. L. Zhang, G. Lu, M. T. Qassrawi, Y. Zhang, and X. Z. Yu, "Feature selection for optimizing traffic classification," *Computer Communications*, vol. 35, pp. 1457-1471, Jul 1 2012.

[31] UNIBS. (2012, 19 Nov). *Università Brescia data sets*. Available: http://www.ing.unibs.it/ntw/tools/traces/download/