Websites:
http://www.sciencepub.net
http://www.sciencepub.net/researcher

Emails:
editor@sciencepub.net
marslandresearcher@gmail.com

Researcher

MARSLAND PRESS
Multidisciplinary Academic Journal Publisher

Internet traffic classification using machine learning approach: datasets validation issues

Hamza Awad Hamza Ibrahim[1], Sulaiman Mohd Nor[2], Izzeldin Ibrahim Abdelaziz[3], Ahmed Abdalla[4], *Haitham A. Jamil*[5] Mohamed Sad[6]

Faculty of Electrical Engineering, Universiti Teknologi Malaysia
hamysra76@hotmail.com

Abstract: Internet traffic classification is an area of current research interest. The failure of port and payload based classification motivates researchers to head towards a machine learning (ML) approach. However, training and testing dataset validation has not been formally addressed. This paper discusses the problem of ML dataset validation and highlights three training issues to be considered in ML classification. The first issue is based on different network characteristics being training and testing datasets which are collected from two different networks. The second issue considers training dataset classes whose real online traffic classes are not presented. The third issue is the geographic place where the network traffic is captured. Real Internet traffic datasets collected from a campus network are used to study the traffic features and classification accuracy for each validation training issue. The experimental results demonstrate that there are differences in some traffic features such as inter-arrival time when training and testing data were collected from different networks. Furthermore, the experiment of the second issue shows that the online classifier achieved the highest accuracy (92.22%) when the ML classifier was trained by dataset classes which have the same ratio of the real online traffic. For the geographic capturing level, the results indicate that there is a difference in the traffic statistical features when the capturing level is different.

[Hamza Awad Hamza Ibrahim[1], Sulaiman Mohd Nor, Izzeldin Ibrahim Abdelaziz, Ahmed Abdalla, *Haitham A. Jamil* Mohamed Sad. **Internet traffic classification using machine learning approach: datasets validation issues**. *Researcher* 2021;14(6):15-25]. ISSN 1553-9865 (print); ISSN 2163-8950 (online). http://www.sciencepub.net/researcher. 3. doi:10.7537/marsrsj130621.03.

Keywords: Internet traffic classification; machine learning; datasets validation issues; online classification

1. Introduction

Internet service providers (ISPs) and network operators are mostly interested in knowing the amount of traffic carried by their networks for the purposes of optimizing network performance and security issues. Therefore, Internet traffic classification is something valuable, particularly for interactive traffic applications such as VoIP and online games.

Simple classification assumes that most applications use well-known port numbers, and the classifier uses this port number to determine the application type. However, most Internet applications use unknown port numbers, or more than one application uses the same port number, which indicates the failure of port base classification [1]. Another classification method is payload based (deep packet inspection) [2], which is individual packet inspection, looking for unique signatures. However, this technique faces two problems. The first is that it is difficult to detect non-standard ports by using packet inspection because of packet encryption. The second is that deep packet inspection touches on users' privacy. In order to solve the problem of past classification methods, machine learning (ML) technique is developed. ML [3] [4] uses artificial intelligence to classify IP traffic, which provides a powerful solution by extracting the right information from application features [5]. Moreover, some of ML algorithms are suitable for Internet traffic flow classification at high speed [6]. Some ML algorithms such SVM can prepare an excellent learning method because of its fast learning rate and the acceptable complexity [7]. ML technique performs in several steps. First, selects of informative features as the attributes of the traffic flow such as packet length, inter-arrival time, protocol, idle time, etc. The second step is training the selected features to establish classification rules. Finally, ML classification is applied for unknown packets/flows using training rules. ML consists of different algorithms which are categorized into two main types: supervised and unsupervised learning.

This paper highlights the problem of ML training and testing datasets from three points of view and we called these points validation issues. Firstly, the effect of having training and testing datasets collected from the same or different networks environment. Secondly, consideration of the real majority and minority classes on the training datasets in case of online classification. Thirdly, the effect of having training and testing datasets which captured from the same or different network geographic levels (switch). Therefore, we utilize real Internet traffic through three experiments to investigate the impacts of each case in ML dataset features and classification performance.

The rest of this paper is organized as follows. Section 2 discusses the three ML datasets validation issues. Section 3 highlights some of related works from validity of the data point of view. Section 4 presents the experimental results and analysis. The conclusion of this study is shown in section 5.

2. ML datasets validation issues

One of the main problems encountered in machine learning Internet traffic classification is the validation of training and testing datasets. Normally, dataset characteristics are supposed to be similar to the real network environment. In this section, we analyze each one of the three previous issues to highlight the factors that can affect the relationship between ML training and testing datasets. It is important to note that all the following three issues have more effect with online classification and less effect with offline classification.

2.1 Training and testing ML dataset collected from the same/different networks environment

The networks include different configurations such as using real IP, using NATed IP[8], etc. The question arises here: are the statistical features of Internet application the same in different network scenarios? In other words, what is the effect when we collect training and testing datasets from the same/different network? Many sub-questions can arise from this main question, such as:

- What is the effect if the training and testing datasets are collected from different network environments?
- What are the benefits if the training and testing datasets are collected from the same network?
- What are the traffic features that will be affected when we change the network characteristics?
- What information needs to be added in ML research papers about training and testing datasets?

For the same class in ML Internet traffic classification, the training dataset is assumed to represent the testing dataset. Figure 1 shows a basic example for two users running the same application class (http) in two different networks. The question here is: does http traffic generated by user 1 and http traffic generated by user 2 have the same traffic features?

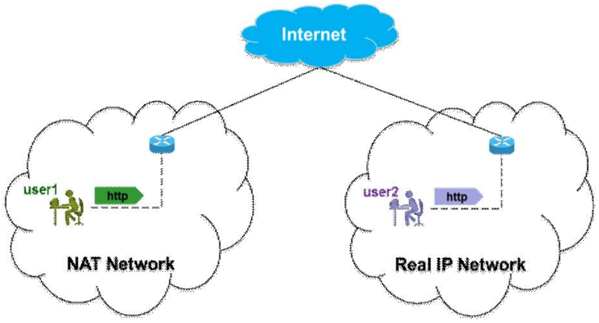

Figure 1. Two users in two different networks run the same application

We believe that different network scenarios can generate different traffic patterns. This variation means the values of the traffic features (such as inter-arrival time and packet length) can be different when the network segments are different. According to Nguyen et al. [9] many Internet applications change their statistical properties over time. Alshammari et al. [10] did a good comparison between classification accuracies of Skype traffic when the training and testing datasets were collected from different networks as well over different years. The training dataset (Univ07) was collected in 2007 from a university in Canada. Three testing datasets were considered: the first when the training and testing datasets were collected from the same network; the second when testing dataset was collected from the same network as training datasets but for different year (Univ10); the last when the testing dataset was collected from different country (Italy). The results show that the Detection Rate (DR) is high and False Positive (FP) is low when the training and testing datasets come from the same network and at the same time.

2.2 Considering the real majority and minority classes on the training datasets for online classification

The second ML datasets validation issue is the effect of using imbalanced training datasets. In Internet traffic, some application classes generate a lot of flows (majority) and others only generate a few flows (minority). WWW applications in University of Cambridge generate about 55.06% of flows, while interactive applications only generate 0.15% [11]. In particular for online classification, imbalance means that the distribution percentage of the classes in the training datasets does not represent the real distribution environment. Figure 2 shows an example of an ML classifier that did not consider the real dataset distribution in the offline training stage. Considering majority and minority classes in the training stage which is totally different from the real online traffic leads to an axiomatic question: what are the effects if the ML classifier did not consider the real online majority and minority classes on the training datasets?

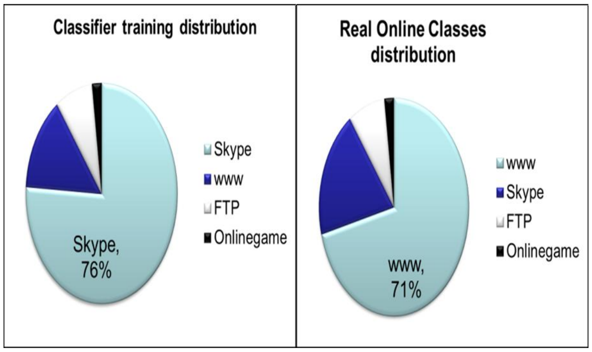

Figure 2. Between training and real classes dataset distribution

The ML classifier results bias towards majority classes and misclassify the traffic of minority classes seriously [12]. In order to build balanced training datasets, some articles [13] select equal numbers of flows for each class; however, it would modify the real distribution of flows [12]. There is a research gap on the problem of imbalance learning [14]. All the state-of-the-art works that we covered studied the imbalanced problem to enhance the accuracy of minority classes, but they did not look from the online classification point of view. Our study differs from others since it evaluates the used training dataset itself. This evaluation is to show the impact of an imbalanced training dataset on the online classification performance. We believe that the online ML Internet traffic classifier is supposed to be trained offline by the same/near distributed percentage of the real online traffic. As an example, if the WWW applications represent 80% of the real traffic, then the classifier is supposed to consider the same percentage (80% or near) for WWW applications in the training stage. It is known that the ML algorithms have less impact when they consider skew dataset distribution [14].

2.3 The effect of capturing the training and testing datasets from the same/different network level

The capturing level means the geographic switches/routers where the data was captured. In this section we need to highlight the question: does the value of the traffic features (such as inter-arrival time and packet length) change if we change the geographic capturing place in the same network? Figure 3 shows two different geographic capturing places for the same user traffic. From this figure, the previous question can be written as: Are the statistical features the same in Usr_traffic_level1 and Usr_traffic_level3? In particular, can we train the ML classifier by data collected from switch of level 1 to classify the traffic that passes through the switch of level 3?

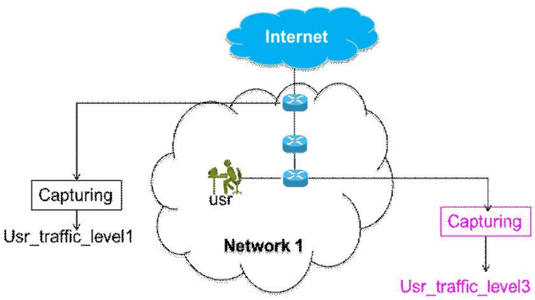

Figure 3. Two different geographic capturing places for the same user traffic

The traffic time characteristics can change rapidly during traffic congestion [15]. There are two factors that can affecting router congestion: the forwarding packet rate and the rate that packets enter the router input queue [16]. Therefore, some traffic feature such as inter-arrival time of the same flow can differ from switch level to another in the same network. Therefore, in Figure 3, Usr_traffic_level1 and Usr_traffic_level3 can differ in case of congestion occurring in any of the switches. The ML dataset instance is defined as traffic features such as Inter-arrival time, packet length, flow duration, etc. If these features are affected by changing of the capturing level, then the classifier performance will be affected.

The previous three sections can summarized in the following question: Based on network characteristics, multi-class balancing (multi-class percentage distribution), and geographic capturing places; how to increase traffic characteristics similarity between training and testing datasets?

3. Related works

In this section, we will discuss the Internet traffic classification works to highlight the lack of work which considers the three ML datasets validation issues mentioned before. In addition, Table 1 summarizes some of previous works based on these ML validation issues.

Alshammari et al. [17] aimed to classify encrypted traffic; SSH and Skype were considered as the case study. The authors developed classifier trained data from one network to test on data from an entirely different network. The testing data is collected from three different places (Dalhousie traces, public traces and DARPA99 traces). Each of these traces is trained from the Dalhousie network. However, the question here is how to ensure the validity of output of the testing stage when the training and testing data for both sets of data is totally different.

The authors in [18] used AdaBoost and C4.5 to classify the traffic into Skype and non-Skype. The Skype traces were collected as labelled data and taken from the campus network. The data were separated into UDP and TCP and classified independently. The classification results are 98% and 94% for UDP and TCP respectively. However, the labelled datasets were collected at different classification times. This causes a difference between classification environment and dataset collection.

The researchers in [4] used ML in their work to identify Internet traffic applications. The authors used ten-fold cross-validation with 100 packets sub-flow. The results showed about 98% precision and 86% recall. As has been the case with the previous works, the problem is the use of training and testing datasets which were collected from two different environments. The first traces collected in real-time using Tcpdump (of unknown origin). The second group comprises the offline pcap files which were obtained from University of Twente (saved files). Again the question is how to train the classifier by datasets collected from one network and to examine this classifier by datasets from another network, where the characteristics of the two networks may be different.

The authors of [19] proposed a method to train the classifier on a combination of short sub-flows to optimize the use of ML classifiers. This is to achieve classification decision before flow expiring, which is very helpful in real-time classification. Some traffic features were derived from the first n packets of the flow to identify the flow. The proposed method was used to classify an online game (Wolfenstein Enemy Territory). However, the ML dataset includes a mix of two traces. The first traces were collected during May and September 2005 from a public game server in Australia. The second traces were two 24-hour periods collected from the University of Twente, Germany, on 6-7 February 2004. Thus, the problem was how to mix two datasets collected from different networks at different period of times. This can be done if the authors ensure that both networks have the same characteristics.

The researchers of [20] proposed a wireless mesh network traffic classification using C4.5. Sub-flow with application behaviors was applied to solve the problem of how to select represented sub-flow. Based on the statistical features of the first n packets, the classifier clusters the flow to one of the defined applications. The proposed method was used to identify some Internet applications such as http, SMTP, FTP, Kazza etc. Similar to the previous study, the ML datasets were collected from two different networks (campus and residential) which can have different characteristics. Thus, as an example, how can it be ensured that the inter-arrival time of http traffic is the same in both networks?

Sun and Chen [21] proposed a method which is suitable to identify the application association with TCP flows. This is based on total data length sent by client (ACK-Len ab) or server (ACK-Len ba) before it received ACK packets. The proposed method was verified by using an ML classifier (C4.5) to classify four types of Internet applications (WWW, FTP, EMAIL and P2P). These applications were collected from three different places as follows: first trace was provided in London UK. in 2006, the second trace was used by [22], which was collected from university in France. The last trace was gained from the author work environment (China). In the same manner as other researches, the traces were collected from different network environments as well different times. Because the classifier was trained by datasets collected from three different networks, it's difficult to use this classifier online to identify the traffic of any one of the same networks.

Nguyen, Armitage et al. [23] proposed and analyzed an approach of training ML classifier using sets of features calculated from multiple sub-flows at different points. This allows the classifier to quickly identify the application at any point of a flow's lifetime. Forward and backward flows are differentiated by features swapped called synthetic sub-flow pairs (SSP). Three training methods were considered: training with full flows, training with one sub-flow, and training with multiple sub-flows. As with [19], the datasets were collected from two distinct networks: a public game server in Australia (2005) and a home network (2007). The same question can arise: how to ensure the similarity between the training and testing datasets for the same class in case of online classification.

In order to achieve the requirements of the network activities, a traffic classifier based on support vector machines (SVM) was presented in [24]. The dataset included three traces collected from three different places: 1) Moore_Set which collected from Cambridge University; 2) Handmade_Set which was labelled in the author's laboratory (China); 3) University_Set which collected from Nanjing University of Posts and Telecommunications. Based on statistical features, the classifier used the first ten packets to identify the flow. However, the authors did not consider the capturing of training and testing datasets from the same place, which is an important datasets validation issue. In addition, how to classify flow includes a large number of packets based on only ten packets.

The authors of [25] proposed an approach that used inter-packet times (IPT) to classify VOIP and online gaming traffic. The method was built based on the hypothesis that real-time interactive applications normally send out a constant-packet-rate (CPR). The authors used deviation-based CPR-traffic to perform the classification. Different deviation metrics, such as standard deviation and coefficient of variation were analyzed to builds an estimator for short-term IPT deviation. The method is useful if the packet-rate of the online game is constant in all networks types. However, the characteristics of the same internet applications differ from each other based on network characteristics [26]. Furthermore, the considered traces were collected from five different networks as well different times.

In order to utilize comprehensive resources of all studies on online games traffic, the authors of [27] provide review of all studies on Massively Multi Player Online Games (MMOG) which became popular type of games in recent years. The authors categorize MMOG into different groups, each one includes several games. Packet inter-arrival time and packet size of games traffic which are related to some articles are presented to study and compare games characteristics. The paper concluded some important issues such as: firstly, there are some discrepancies in previous games classification articles outcome; secondly, each game group produces network traffic with unique characteristics which distinguishes itself from others. This work has an advantage against others since it highlights the different characteristics of the same Internet application (game).

Bujlow et al. [15] proposed a classification method based on the C5.0 ML algorithm. The authors recruited volunteers from among the users to generate the real labelled traffic. Some software was installed on the volunteers' computers to capture the relevant traffic and submit the datasets to the server. C5.0 ML algorithm was used as statistical classifier to distinguish between seven types of applications. We totally agree with the authors when they develop a classifier to be network-dependent, which means it will train in each network independently. However, the traffic flows were collected from volunteers' NIC; the characteristics of this traffic can change when passing through network switches. In addition, the online classifier normally installed in the switch/router to identify the total traffic passes through this device; this means the training datasets should be collected from the same level.

Table 1 summarizes some of the related works from the datasets validation issues point of view. The term 'yes' means that the work achieved the issue, while 'no' means the work did not achieve the issue. From our best knowledge, we did not find in state-of-the-art studies any work that achieved the three validations issues and performed the online classification at the same time.

Table 1. The datasets validation issues in some related works

Works	Are the training and testing datasets collected from the same network?	Does the work consider the real majority and minority classes on? the training datasets	Does the work capture the training and testing from the same geographic level?	Does the work apply an online Classification?
(Min, Xingshu et al. 2013) [28]	yes	yes	yes	no
(Adami, Callegari et al. 2012) [2]	no	ignored	no	yes
(Chen, Yang et al. 2009) [29]	yes	no	yes	no
(Molnar and Perenyi 2011) [30]	no	ignored	no	no
(Nguyen, Armitage et al. 2012) [23]	no	yes	no	no
(Gu, Zhang et al. 2011) [24]	no	yes	no	no
(Gu, Wang et al. 2010) [31]	yes	yes	yes	no
(Sun and Chen 2011) [21]	no	yes	no	no
(Gu, Zhang et al. 2011) [20]	no	yes	no	no
(Hong, Gu et al. 2009) [32]	yes	yes	yes	no
(Xu, Qiong et al. 2009) [33]	yes	ignored	no	no
(Bujlow, Riaz et al. 2012) [15]	yes	ignored	yes	no
(Bonfiglio, Mellia et al. 2007) [34]	no	ignored	no	no
(Weirong and Gokhale 2010) [35]	yes	yes	yes	no
(Alshammari and Zincir-Heywood 2010) [3]	yes	ignored	yes	no
(Wang, Xiang et al.2011) [36]	no	yes	yes	no
(Shrivastav and Tiwari 2010) [37]	yes	ignored	yes	no
(Yuan, Li et al. 2010) [38]	yes	yes	ignored	no

4. Experiments and analysis

In order to check the effect of the ML datasets validation issues (discussed in section 2), three experiments were considered. The first experiment was to find the change of traffic features when the network characteristics were changed. The second experiment was to highlight the impact of online ML classification accuracy when we did not consider the real majority and minority classes on the training datasets. The last experiment was to answer the question: does the value of the traffic features (such as inter-arrival time and packet length) change if we change the geographic capturing place in the same network?

4.1 Are the Internet traffic features of one class (application) same in different network characteristics?

Traffic features mean the traffic patterns used in ML classifier datasets such as inter-arrival time and packet length. Skype traffic has gained significant attention and has become one of the most popular forms of VoIP software and it is used in our campus network; in addition Skype can represent P2P applications. Therefore, we consider full real Skype session (call) datasets to answer the question: are Internet application (Skype in particular) traffic features the same when the traffic is collected from different network environments? All data were collected by Wireshark [39], and the statistical values are summarized from the same software.

Eight different Skype calls were considered and divided into two groups. The first group consists of four different calls (calls 1.1–1.4). In this group, the Skype sessions are full Skype calls between two SCs located in two different countries. This means that both Skype clients are located behind a firewall and thus using NATed IP. The second group consists of the other four calls (calls 2.1–2.4) of Skype session between two SCs located inside our campus area. This group of calls is configured with no firewall between both clients (the two clients used real IPs). We aimed to generate two different datasets to study Skype traffic features in two different network scenarios.

Table 2 and Figure 4 show statistical features of the results of the two groups. For all calls of group one, the TCP rate, UDP rate, average packets per second and average packets per size (bytes) were observed to have almost similar values. This means that the same network environments produce the same traffic features. However, when any call of group one is compared with other call from group two, clear differences appear in the TCP rate, UDP rate, and average packets per second. This means that the statistical features of the same Internet application traffic (Skype in particular) are not the same when the network environments are different.

Table 2: Some features' values when network characteristics are different

Full call Connection	TCP rate	UDP rate	Avg.pckt/sec (S)	Avg.pckt/size(bytes)
call1.1	99.89%	0.11%	138.244	128.170
call1.2	98.73%	1.24%	112.738	130.039
call1.3	83.16%	16.84%	195.569	70.633
call1.4	99.43%	0.56%	162.724	75.128
call 2.1	5.29%	90.66%	20.782	147.465
call 2.2	1.08%	98.92%	84.411	120.204
call 2.3	0.30%	99.70%	87.290	121.991
call 2.4	1.83	98.12	61.613	120.158

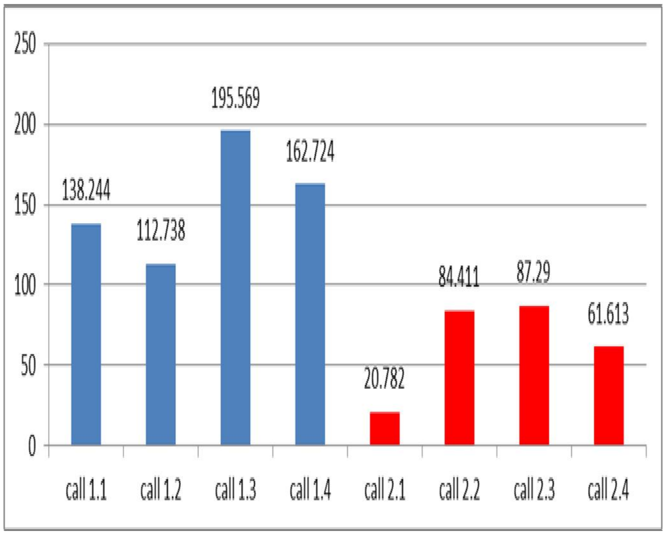

Figure 4. Average packets per second for same application (Skype) when the network characteristics are different

4.2 The effect of the real majority and minority training datasets on the online classification

As discussed earlier, the important issue in ML classifier is to use valid training and testing datasets. In this section, we will discuss three experiment scenarios. In the first scenario, the online classifier was trained offline by datasets that do not have the same ratio as the real online traffic (discussed in section 2.2). This means that we used imbalanced training datasets, which did not represent the real online traffic distribution. In the second scenario, we used equal number of flows for each class which also did not represent the real online traffic distribution. In the last scenario, the ML classifier was trained by dataset classes which have the same ratio as the real online traffic. Real time Internet traffic was collected from the campus network. This includes two types of Internet application classes (WWW and Skype). The WWW class includes http and https applications which have the higher percentage of the campus traffic. Skype traffic was generated by real communication sessions (calls) between Skype clients (SC), which are located within and outside the campus area. The selected applications were run manually through some monitored clients (users). The real online traffic generated by the monitored users is controlled as ~86% www class and ~14% Skype class. In the first training scenario, the classifier was trained imbalanced by 90% Skype and 10% www, which is the reverse of the real traffic and did not represent the real online traffic distribution. In the second scenario, we trained the ML by an equal number of flows from each class (50% www and 50% Skype). In the third scenario, the classifier was trained by 90% www and 10% Skype which is near to the real online traffic distribution (Table 3).

Table 3: Real classes distribution and training classes distribution

Class	Real online classes traffic distribution	Training classes distribution (the three experiments scenarios)		
		Using imbalanced training dataset	Using equal number of flows for each class	Using near online real traffic ratio
WWW	~86%	10%	50%	90%
Skype	~14%	90%	50%	10%

Online classification means that the decision of which packet/flow belongs to which class is supposed to be based on the traffic speed, just like any hardware classifier (Packet Shaper, SANGFOR) which is installed on the network path to classify the traffic in the network speed. We used the online hybrid classifier proposed in [40], which makes a classification decision within traffic speed. On this hybrid classifier, the ML classifier has an essential role in the online classification decisions. In all experimental scenarios, the volunteer users in the real online classification stage run the same applications distributed as: ~86% is www and ~14% is Skype. This continuously generates a dataset which is totally different from the training dataset. We aim to study which training scenario can give a higher accuracy with online classification. Table 4 and Figure 5 show the classification results of the three considered scenarios.

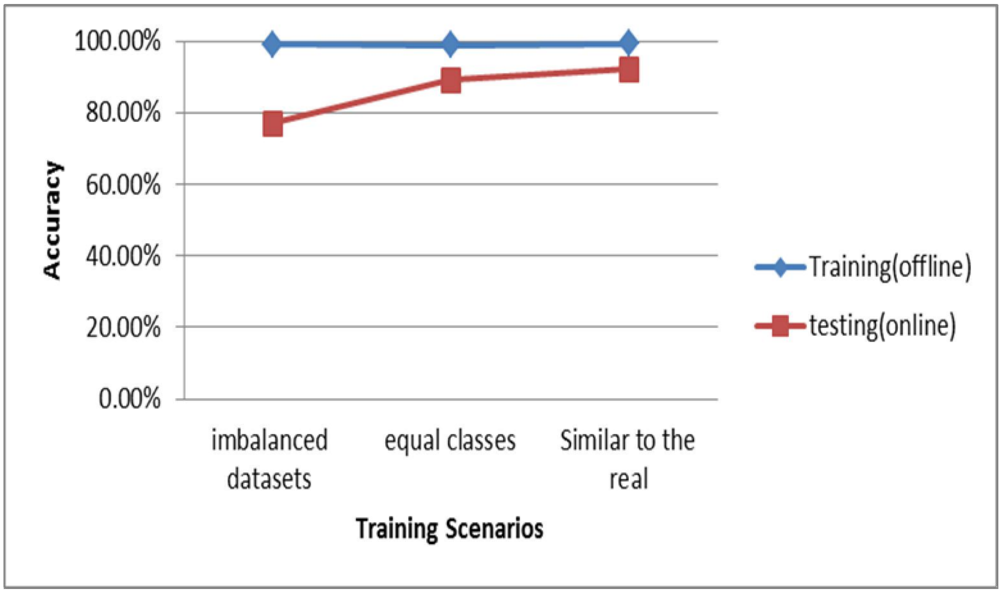

Figure 5. The classification results of the three training scenarios

4.3 Traffic features when capturing from two different geographic network levels

The network structure can include several geographic switches/routers levels (section 2.3). In this section, we need to answer the question: do the values of the traffic features (such as inter-arrival time and packet length) change if we change the geographic capturing place in the same network? Typically, ML classifier used datasets which present the statistical properties of Internet traffic flows. Therefore, the appointed feature value is supposed to be the same in the two classification stages (training and testing).

We captured the same user's traffic from two different geographic network levels and then this traffic to see the feature status of each level. The first level is the user's network interface card (NIC) and the second level is a traffic mirror which reflects the same traffic flow. Inter-arrival time and packet length are the common features used for the Internet traffic classification. Therefore we calculate five statistical factors for each feature which are: min, max, mean, variance, and standard deviation. For fairness, we start the capturing of both levels at the same time, and we capture for the same period of time. In the mirror level the captured traffic was filtered by users' IPs. This experiment considers WWW class (http and https) which consisted of 989 flows includes more than 0.04GB. The results of statistical factors for each

capturing level are shown in Table 5 and Figure 5. The results indicate that there are some differences in traffic features for the same user traffic when we capture traffic from two different geographic levels. This difference is not significant in inter-arrival time mean (Tmean) nor in length variance (Lvar). It is important to highlight that there are a lot of traffic flows seen in NIC capturing level which are not seen in the mirror level. These flows can affect the classification performance because they affect the statistical features. In other words, some communications were generated between the clients (NIC) and the near switch/router and this will not be seen in upper level switch/router

5. Conclusion

The ability to accurately identify the network traffic is a very important factor to achieve accurate and fast management tasks. Machine learning plays an essential role in network traffic classification; however, the validation of training and testing datasets has not been formally addressed. In this paper, we performed an analysis of ML training and testing datasets to highlight three important validation issues for ML network traffic classification.

These issues are defined as follows: the first issue is when training and testing datasets were collected from the same/different network; the second issue is when the ML classifier did not consider the real online classes' distribution in the training stage; the third issue is when training and testing datasets were captured from the same/different geographic network level. For the first validation issue, real campus traffic was used to study the traffic features status when the network characteristics are different. The results indicate that there are differences in some features such as inter-arrival time which can affect the classification performance. In the second validation issue, three experimental scenarios were performed to evaluate ML classification accuracy in case of capturing from different traffic levels. Between the three scenarios, the online classifier achieved the highest accuracy (92.22%) when the ML classifier was trained by dataset classes which have the same ratio as the real online traffic. For the third validation issue, we captured the same user's traffic from two different geographic network levels and then analyzed this traffic to see the feature status of each level. The results indicate that there is a difference in the traffic statistical features (such as flow variance) when the capturing level is different.

Therefore, we conclude three points for ML classification: i) training and testing traffic datasets should be collected from the same network segment and at the same/near time; ii) the online classifier should be trained by dataset classes which have the same/near distribution percentages of the real online classes; iii) training datasets should be captured from the same network geographic level (switch/router) as the testing datasets.

Table 4: classification results of the three considered scenarios

Scenario	Number of flows		Accuracy	
	Training	~ 4 minutes online Testing	Training(offline)	testing(online)

First scenario (imbalanced training dataset): training datasets did not represent the real online distribution	www = 22;
	Skype= 220
		WWW = 315;
		Skype= 45
			99.17%	76.94%

Second scenario (application classes equal each other): training datasets did not represent the real online distribution	www = 220
	Skype= 220
		WWW = 315;
		Skype= 45
			98.86%	89.17%

Third scenario: training datasets represent the real online distribution	www = 220
	Skype= 22
		WWW = 315;
		Skype= 45
			99.29%	92.22%

Table 5: Statistical features values for the two capturing levels

	Tmax	Tmean	Tvar	Tstd	Lmin	Lmax	Lmean	Lvar	Lstd
mirror	34.25	4.70	347.19	11.01	59.74	510.22	247.47	76577.94	170.49
NIC	28.21	4.94	442.25	9.41	73.23	335.03	182.22	44304.06	99.18

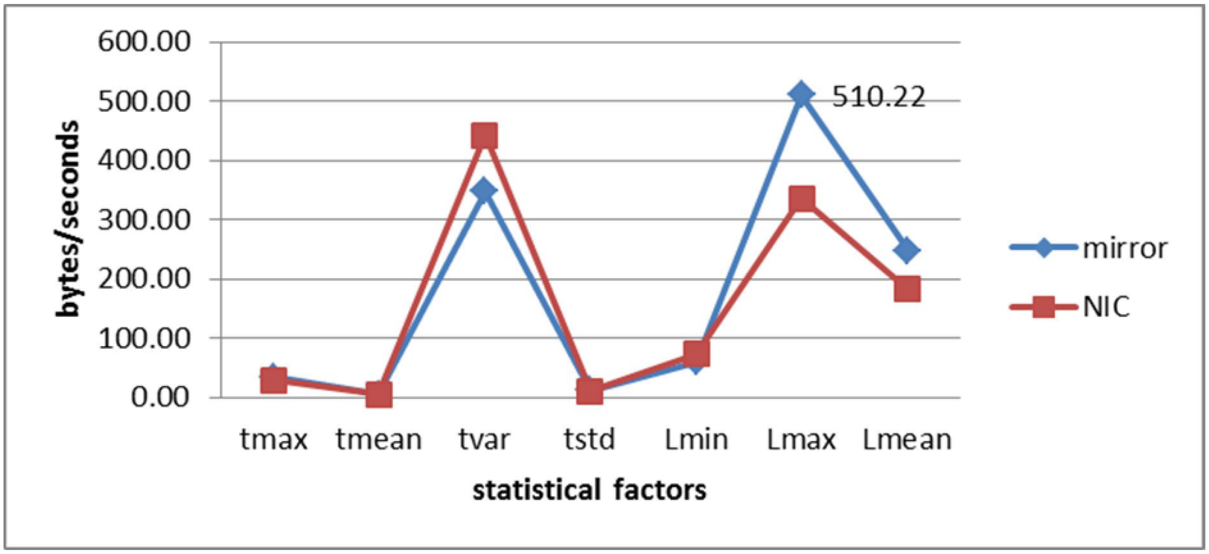

Figure 6: Statistical feature results for the two capturing levels

REFERENCES

1. Nguyen, T.T.T., Armitage, G.: A Survey of Techniques for Internet Traffic Classification using Machine Learning. Ieee Commun Surv Tut 10(4), 56-76 (2008). doi:Doi 10.1109/Surv.2008.080406
2. Adami, D., Callegari, C., Giordano, S., Pagano, M., Pepe, T.: Skype-Hunter: A real-time system for the detection and classification of Skype traffic. Int J Commun Syst 25, 386–403 (2012).
3. Alshammari, R., Zincir-Heywood, A.N.: An investigation on the identification of VoIP traffic: Case study on Gtalk and Skype. In: Network and Service Management (CNSM), 2010 International Conference on, 25-29 Oct. 2010 2010, pp. 310-313
4. Jesudasan, R.N., Branch, P., But, J.: Generic Attributes for Skype Identification Using Machine Learning. Technical Report 100820A (2010).
5. Yu, J., Lee, H., Im, Y., Kim, M.S., Park, D.: Real-time Classification of Internet Application Traffic using a Hierarchical Multi-class SVM. Ksii T Internet Inf 4(5), 859-876 (2010). doi:DOI 10.3837/tiis.2010.10.009
6. Soysal, M., Schmidt, E.G.: Machine learning algorithms for accurate flow-based network traffic classification: Evaluation and comparison. Perform Evaluation 67(6), 451-467 (2010). doi:DOI 10.1016/j.peva.2010.01.001
7. Liu, B.Y., Feng, V.S., Chang, S.Y.: Performance analysis for relay networks with hierarchical support vector machines. Int J Commun Syst 26(3), 337-355 (2013).
8. Egevang, K., Francis, P.: The IP network address translator (NAT). In. RFC 1631, may, (1994)
9. Nguyen, T.T.T., Armitage, G.: Clustering to assist supervised machine learning for real-time IP traffic classification. Ieee Icc, 5857-5862 (2008).
10. Alshammari, R., Zincir-Heywood, A.N.: Is machine learning losing the battle to produce transportable signatures against VoIP traffic? In: Evolutionary Computation (CEC), 2011 IEEE Congress on, 5-8 June 2011 2011, pp. 1543-1550
11. Liu, Q., Liu, Z.: A comparison of improving multi-class imbalance for internet traffic classification. Inform Syst Front, 1-13 (2012).
12. Liu, Z., Liu, Q.: Studying cost-sensitive learning for multi-class imbalance in Internet traffic classification. Journal of China Universities of Posts and Telecommunications 19(6), 63-72 (2012).
13. Jin, Y., Duffield, N., Erman, J., Haffner, P., Sen, S., Zhang, Z.L.: A modular machine learning system for flow-level traffic classification in large networks. ACM Transactions on Knowledge Discovery from Data 6(1) (2012).
14. Wang, S., Yao, X.: Multiclass Imbalance Problems: Analysis and Potential Solutions. Ieee T Syst Man Cy B 42(4), 1119-1130 (2012). doi:Doi 10.1109/Tsmcb.2012.2187280
15. Bujlow, T., Riaz, T., Pedersen, J.M.: A method for classification of network traffic based on C5.0 Machine Learning Algorithm. In: Computing, Networking and Communications (ICNC), 2012 International Conference on, Jan. 30 2012-Feb. 2 2012 2012, pp. 237-241
16. Russell, B., Littman, M.L., Trappe, W.: Integrating machine learning in ad hoc routing: A wireless adaptive routing protocol. Int J Commun Syst 24(7), 950-966 (2011).
17. Alshammari, R., Zincir-Heywood, A.N.: Machine learning based encrypted traffic classification: Identifying SSH and Skype. In: Computational Intelligence for Security and Defense Applications, 2009. CISDA 2009. IEEE Symposium on, 8-10 July 2009 2009, pp. 1-8
18. Angevine, D., Zincir-Heywood, A.N.: A Preliminary Investigation of Skype Traffic Classification Using a Minimalist Feature Set. In: Availability, Reliability and Security, 2008. ARES 08. Third International Conference on, 4-7 March 2008 2008, pp. 1075-1079
19. Nguyen, T.T.T., Armitage, G.: Training on multiple sub-flows to optimise the use of Machine Learning classifiers in real-world IP networks. Conf Local Comput Ne, 369-376 (2006).
20. Gu, C., Zhang, S., Xue, X., Huang, H.: Online wireless mesh network traffic classification using machine learning. Journal of Computational Information Systems 7(5), 1524-1532 (2011).
21. Sun, M.F., Chen, J.T.: Research of the traffic characteristics for the real time online traffic classification. Journal of China Universities of Posts and Telecommunications 18(3), 92-98 (2011).
22. Bernaille, L., Teixeira, R., Akodkenou, I., Soule, A., Salamatian, K.: Traffic classification on the fly. Comput Commun Rev 36(2), 23-26 (2006).
23. Nguyen, T.T.T., Armitage, G., Branch, P., Zander, S.: Timely and Continuous Machine-Learning-Based Classification for Interactive IP Traffic. Networking, IEEE/ACM Transactions on PP(99), 1-1 (2012). doi:10.1109/tnet.2012.2187305
24. Gu, C., Zhang, S., Huang, H.: Online internet traffic classification based on proximal SVM. Journal of Computational Information Systems 7(6), 2078-2086 (2011).

25. Kuan-Ta, C., Jing-Kai, L.: Rapid Detection of Constant-Packet-Rate Flows. In: Availability, Reliability and Security, 2008. ARES 08. Third International Conference on, 4-7 March 2008 2008, pp. 212-220
26. But, J., Nguyen, T., Stewart, L., Williams, N., Armitage, G.: Performance analysis of the ANGEL system for automated control of game traffic prioritisation. In: 2007, pp. 123-128
27. Che, X.H., Ip, B.: Packet-level traffic analysis of online games from the genre characteristics perspective. J Netw Comput Appl 35(1), 240-252 (2012). doi:DOI 10.1016/j.jnca.2011.08.005
28. Min, D., Xingshu, C., Jun, T.: Online Internet traffic identification algorithm based on multistage classifier. Communications, China 10(2), 89-97 (2013). doi:10.1109/cc.2013.6472861
29. Chen, Z.X., Yang, B., Chen, Y.H., Abraham, A., Grosan, C., Peng, L.Z.: Online hybrid traffic classifier for Peer-to-Peer systems based on network processors. Appl Soft Comput 9(2), 685-694 (2009). doi:DOI 10.1016/j.asoc.2008.09.010
30. Molnar, S., Perenyi, M.: On the identification and analysis of Skype traffic. Int J Commun Syst 24(1), 94-117 (2011). doi:Doi 10.1002/Dac.1142
31. Gu, R., Wang, H., Ji, Y.: Early traffic identification using Bayesian networks. In: 2010, pp. 564-568
32. Hong, M.-h., Gu, R.-t., Wang, H.-x., Sun, Y.-m., Ji, Y.-f.: Identifying online traffic based on property of TCP flow. The Journal of China Universities of Posts and Telecommunications 16(3), 84-88 (2009). doi:http://dx.doi.org/10.1016/S1005-8885(08)60231-9
33. Xu, T., Qiong, S., Xiaohong, H., Yan, M.: A Dynamic Online Traffic Classification Methodology Based on Data Stream Mining. In: Computer Science and Information Engineering, 2009 WRI World Congress on, March 31 2009-April 2 2009 2009, pp. 298-302
34. Bonfiglio, D., Mellia, M., Meo, M., Rossi, D., Tofanelli, P.: Revealing Skype traffic: When randomness plays with you. Comput Commun Rev 37(4), 37-48 (2007).
35. Weirong, J., Gokhale, M.: Real-Time Classification of Multimedia Traffic Using FPGA. In: Field Programmable Logic and Applications (FPL), 2010 International Conference on, Aug. 31 2010-Sept. 2 2010 2010, pp. 56-63
36. Wang, Y., Xiang, Y., Yu, S.: Internet Traffic Classification Using Machine Learning: A Token-based Approach. Computational Science and Engineering (CSE), 2011 IEEE 14th International Conference on (2011).
37. Shrivastav, A., Tiwari, A.: Network Traffic Classification Using Semi-Supervised Approach. In: Machine Learning and Computing (ICMLC), 2010 Second International Conference on, 9-11 Feb. 2010 2010, pp. 345-349
38. Yuan, R.X., Li, Z., Guan, X.H., Xu, L.: An SVM-based machine learning method for accurate internet traffic classification. Inform Syst Front 12(2), 149-156 (2010). doi:DOI 10.1007/s10796-008-9131-2
39. Orebaugh, A., Ramirez, G., Burke, J., Pesce, L., Wright, J., Morris, G.: Wireshark & Ethereal Network Protocol Analyzer Toolkit. Syngress, (2007)
40. Hamza Ibrahim, H.A., Mohd Nor, S., Mohamed Abdelaziz, I.I., Alfaki Abdalla, A.A.: SSPC algorithm based on three different methods for online Skype traffic classification. Journal of Theoretical and Applied Information Technology 53(3), 422-429 (2013).

6/2/2021