Computer Networks 144 (2018) 271–279

Contents lists available at ScienceDirect

Computer Networks

journal homepage: www.elsevier.com/locate/comnet

QoS-Classifier for VPN and Non-VPN traffic based on time-related features

Julian Andres Caicedo-Muñoz[a,*], Agapito Ledezma Espino[b], Juan Carlos Corrales[a], Alvaro Rendón[a]

[a] Telematics Engineering Group (GIT), Universidad del Cauca, Campus Tulcan, Popayán-Cauca, Colombia
[b] Department of Computer Science and Engineering, Universidad Carlos III de Madrid, Avenida de la Universidad 30, Leganes-Madrid 28911, Spain

ARTICLE INFO

Article history:
Received 26 January 2018
Revised 9 August 2018
Accepted 12 August 2018
Available online 13 August 2018

Keywords:
QoS classifier
Per-hop behavior
VPN traffic characterization
Machine learning

ABSTRACT

The Quality of Service (QoS) is a continuous challenge issue in the telecommunication industry, mainly for having an impact on telco services provision. Traffic Classification, Traffic Marking, and Policing are general stages of QoS managing. Different approaches have focused on Traffic Classification and Traffic Marking, which machine learning algorithms arise as promising techniques ones. However, Traffic Marking overtime-related features is not widely explored, especially for Virtual Private Network (VPN) traffic. Hence, a specific QoS classifier for VPN traffic based on per-hop behavior (PHB) for a specific domain was proposed. To this end, a baseline QoS-Marked dataset was generated from a characterized VPN traffic; to which some machine learning algorithms were compared and a T-Tester was performed. As a result, Bagging-based learning model has the best behavior for all scenarios in which the higher value achieved was a 94,42% accuracy. Consequently, a QoS classifier is an effective approach for traffic treatment on Differentiated Services (DiffServ) networks.

1. Introduction

The Quality of Service (QoS) is a continuous challenging issue in the telecommunication industry, mainly for having an impact on telco services provision. In that sense, ITU-T has defined QoS as *"Totality of characteristics of a telecommunications service that bear on its ability to satisfy stated and implied needs of the user of the service"* [1]. In a narrow sense, QoS has a tight relationship with network performance, particularly, jitter, bandwidth, delay, and throughput which are a few quantitative parameters in the network management, but also, QoS is concerned of how managing any congestion in the network. Hence, Traffic Classification, Traffic Marking and Policing are the general stages of controlling traffic and preventing any network collapse.

Some works as [2–4] are promising approaches for Traffic Classification based on machine learning (ML) algorithms, but these are considering neither QoS managing nor the implementation of any method of Traffic Marking. Traffic Marking and Policing processes were considered by Liu and coworkers [5,6], wherein each packet is marked based on customer preferences and payload. Payload checking for marking is an effective manner to determinate the type of traffic but is not an efficient solution, specifically for marking Virtual Private Network (VPN) traffic wherein user data is protected.

In this paper, classification and marking processes based on some time-related features for VPN and Non-VPN traffic were proposed. Two initial QoS classifier tasks were defined for treatment of traffic and a QoS-marked dataset was generated for being routed on a Differentiated Services (DiffServ) network. Our major contributions are

- a general approach for a QoS classifier built on machine learning model as from network traffic flow.
- a domain-specific implementation for QoS treatment based on per-hop behavior (PHB) and service classes for a VPN and Non-VPN traffic.
- an evaluation of the ML-based classifiers typically used in the literature for QoS classifier proposed.

A T-Tester as a Weka [7] experiment was conducted for comparing different ML classifiers in the QoS classifier approach. Default settings in T-Tester were applied for the experiment. As a result, a statistical comparison was obtained for the model proposed. Similarly, the experiment was replicated in [8] wherein only a C4.5 and kNN were performed for VPN and Non-VPN Traffic Classification purposes.

* Corresponding author.
E-mail addresses: jacaicedo@unicauca.edu.co (J.A. Caicedo-Muñoz), ledezma@inf.uc3m.es (A. Ledezma Espino), jcorral@unicauca.edu.co (J.C. Corrales), arendon@unicauca.edu.co (A. Rendón).

https://doi.org/10.1016/j.comnet.2018.08.008
1389-1286/

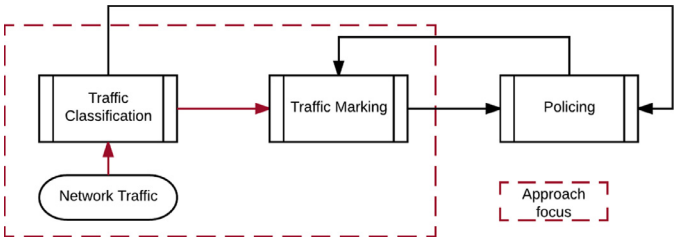


Fig. 1. General stages for QoS managing - adapted from [9].

Table 1
Traffic by applications [8].

Traffic	Application
Web Browsing (BRW)	Firefox and Chrome
Email (Mail)	SMPTS, POP3S, and IMAPS
Chat	ICQ, AIM, Skype, Facebook, and Hangouts
Streaming (STR)	Vimeo and Youtube
File Transfer (FT)	Skype, FTPS, and FTP using Filezilla
VoIP	Facebook, Skype, and Hangouts
P2P	uTorrent and Transmission (Bittorrent)

The rest of this paper is structured as follows. Section 2 presents methods and original dataset used as support elements in this work. Besides, a QoS classifier approach for labeling of characterization VPN and Non-VPN traffic based on PHB method is described. Section 3 outlines the results obtained after having performed some machine learning algorithms. Section 4 discusses the results obtained. Finally, Section 5 conclusion and future works are presented.

2. Materials and methods

According to QoS managing the Traffic Classification, Traffic Marking, and Policing are general stages for controlling network flows and maintaining an acceptable performance in the telco service provision (See Fig. 1).

Traffic Classification stage is the procedure to identify different applications and protocols that exist in a network [10]. Michael Finsterbusch et.al [11] has split out the traffic classification techniques in four categories: port-based, statistical approach, protocol decoding, and pattern-based; both port-based and statistical approaches adopt techniques not intrusive as those applied to protocol decoding and pattern-based approaches. However, port-based depends on the well-known ports for protocols assigned by IANA [12]. In fact, some applications that are not registered can be bypassing any censorship using other protocols or can be using random ports to send data. As such, statistical approach is increasing attention during last decade, especially, the machine learning algorithms are promising alternatives for modeling captured Internet flows.

Traffic Marking stage is a procedure to modify the QoS fields of the incoming and outgoing packets based on service policies [9]. Marking can be done in layer 2 (i.e. Class of Service - CoS) or in layer 3 (i.e. IP Precedence or DSCP) of any packet, particularly, Differentiated Service Code Point (DSCP) is used for managing traffic classes in DiffServ networks in a proper manner [13]. Besides, having a well-defined DSCP as proposed by RFC-2474 [14], allows an effective mapping to EXP field in MPLS networks [15].

Policing stage is a procedure for monitoring of data rates for a particular traffic class [9]. Policing allows a network administrator to have a threshold for re-marking or dropping of packets. Increasing of traffic as regards policing data gathered is taken into account to define new QoS policies for controlling and structure the network.

In this paper, we have focused on Traffic Classification and Traffic Marking stages, principally for encrypted VPN traffic. The characterization VPN traffic is a challenging issue due to being a protocol encapsulation method for maintaining the privacy of data on the network, consequently, traffic classification based on traditional techniques (e.g based on Deep-Packet Inspection) are not completely useful. Recently, an encrypted VPN traffic dataset based on time-related features has been generated by Habibi et al. [8]. The authors have proposed a flow-based classification method and have compared only two classifier algorithms (i.e. C4.5 and kNN). Next is briefly described the dataset, selected algorithms, and scenarios.

2.1. Dataset

The dataset contains real traffic generated by traditional applications in a regular session and a VPN session (e.g. Firefox, SMPTS, Skype, and Facebook, among others). As result, the authors in [8] have labeled a dataset with 14 different traffic classes split out in 7 VPN and 7 Non-VPN traffic classes separately. The type of traffic is shown in Table 1.

2.2. Machine learning algorithms

Habibi et al. [8] have only adopted C4.5 and kNN to evaluate the quality of classification processes based on two common metrics:

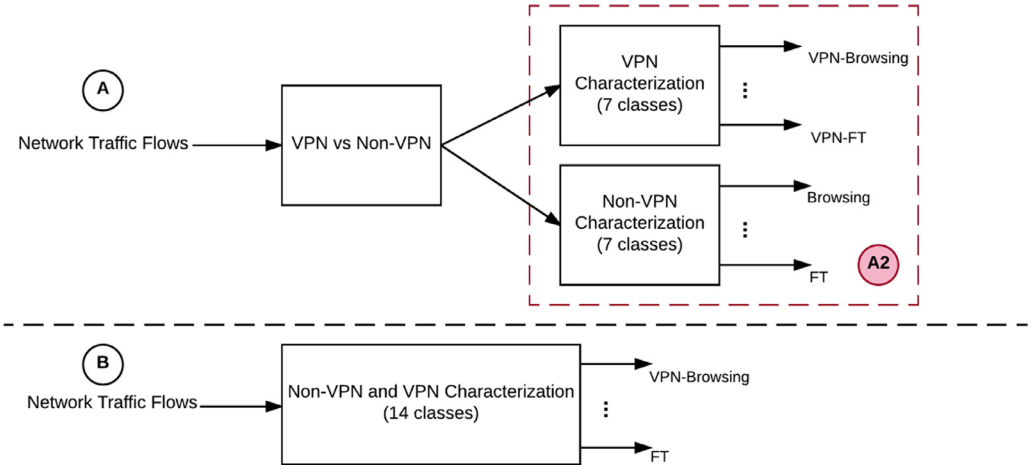


Fig. 2. Scenarios [8] - A2 and B have been analyzed for QoS classifier approach.

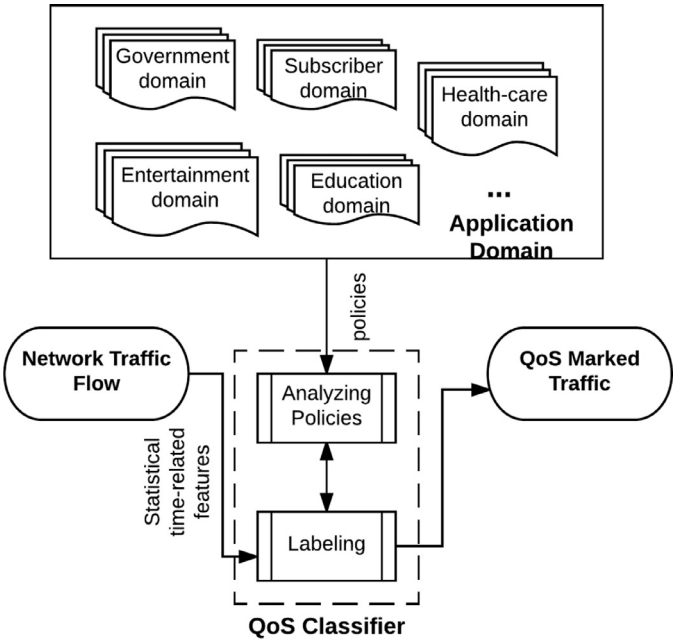


Fig. 3. QoS classifier general approach.

Precision and Recall. As a result, C4.5 algorithm presents the best behavior as classification model in two scenarios.

Finally, two scenarios were implemented to obtain the traffic classes as depicted in Fig. 2. First one (A), distinguishes between only a Non-VPN and VPN traffic, later a specific characterization is realized for labeling each class according to Non-VPN and VPN Traffic (A2). The second one (B), a mixed dataset is obtained in one step and all VPN and Non-VPN traffic are labeled.

2.3. QoS Classifier approach

The approach proposed was focused on the two early stages of QoS managing, Traffic Classification, and Marking. Fig. 3 is depicted as a general approach.

QoS classifier has two inputs. First, traffic with statistical time-related features. Principally, time-related features are obtained in a simple manner by analyzing flow-based data and computationally efficient [8]. Besides, statistical time-related features have a good result by applying some machine learning algorithms techniques [16]. Second, a policy coming from any application domain. An application domain depends on which context a network administrator provides QoS for different services. Indeed, QoS policies in the education domain differ from normal subscriber domain; namely, in the education context can prevail VoIP and File transfer services over Chat or P2P services, meanwhile in the subscriber domain, services can be attended indistinctly or give a bit more score value to Streaming service based on recently consumer trends.

QoS Classifier output is a Marked Traffic in order to QoS managing on the network. Marking process depends on policies established by a particular domain.

QoS Classifier contains two tasks to combining the inputs and generating a QoS marked traffic. Analyzing Policies task was defined for mapping any specific domain traffic rule to a specific QoS field (e.g. DSCP in DiffServ architectures, EXP in MPLS). Labeling task was defined as changing the type of traffic characterized to a well-known label in the network (e.g. AF/EF in DiffServ or E-LSP/L-LSP in MPLS).

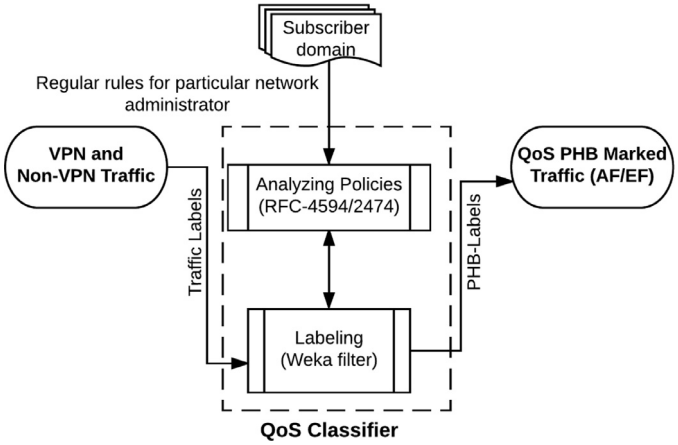


Fig. 4. QoS classifier implementation.

2.4. General approach implementation

To verify general approach, some particular policies on specific VPN traffic were implemented in order to having a QoS Marked traffic (Fig. 4).

2.4.1. Subscriber domain

Regular consumers require all services to work well on the network (e.g home user). Today consumer trends are focused on streaming video as reported by Internet.org [17]. On the other hand, VPN applications are being more popular because of protecting user personal data [18,19].

2.4.2. VPN/NonVPN-traffic

This traffic needs to be processed before getting into QoS classifier. To this end, the normal VPN and NonVPN traffic generated by some application (e.g Skype, Hangouts, Chrome, Facebook) and conveyed on the network were collected and processed by Habibi et al. [8]. As result, a VPN and Non-VPN traffic with some time features were generated as a network traffic flows (Traffic-Labels for short). Actually, VPN traffic is an important issue due to increasing of users are going online using VPN applications for avoiding internet restrictions [20].

2.4.3. QoS classifier

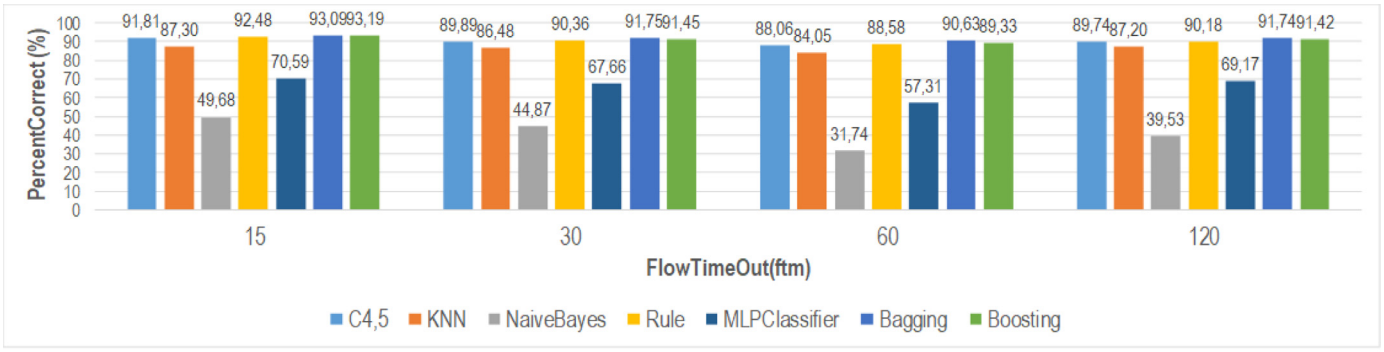
An analyzing policies task has been implemented taking account services class definition by RFC-4594 [21] and PHB as defined by RFC2474 [14]. This task is network administrator knowledge dependent and manual process. On the other hand, a labeling task changes type VPN traffic (i.e. Traffic-Labels) to PHB-Labels using a Weka filter algorithm. As a result, Table 2 shows the relationship between Traffic-Labels implemented by Habibi et al. [8] and PHB-Labels of QoS Classifier based on subscriber domain rules for representing services classes and dropping.

2.4.4. QoS PHB marked

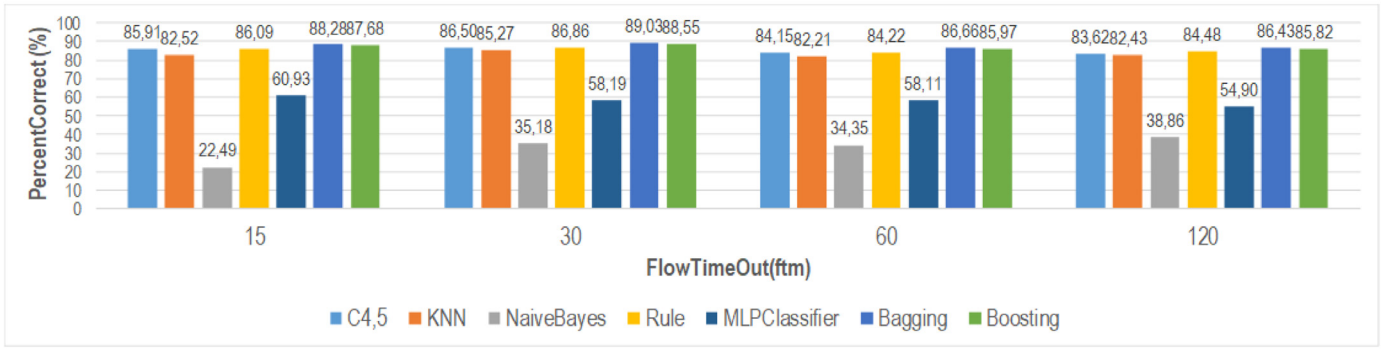
A specific baseline dataset is the outcome of QoS Classifier (i.e. PHB-Labels). An Input dataset with statistical time-related features was modified taking account a network administrator rules for a specific domain and the PHB concept [22].

3. Experiments

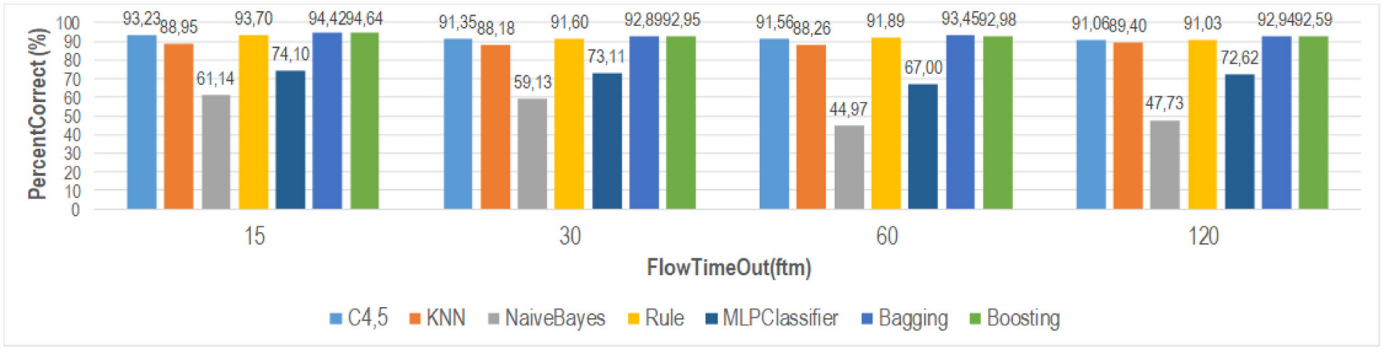
Two baseline dataset was obtained by mapping the initial Traffic-Labels (i.e. QoS-PHB Marked Traffic). The A2 dataset has PHB-Labels ranging from AF12 to AF32 for Non-VPN and AF11 to AF31 for VPN traffic. VoIP traffic was labeled with EF for both (i.e



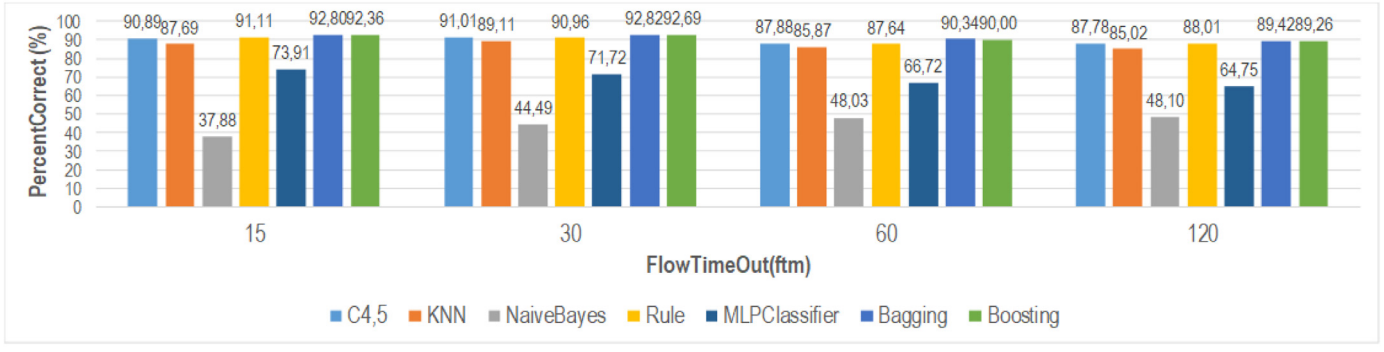
(a) NonVPN Percent Correct Traffic Labels



(b) VPN Percent Correct Traffic Labels



(c) NonVPN Percent Correct PHB Labels



(d) VPN Percent Correct PHB Labels

Fig. 5. Percent correct for Traffic-labels and PHB-labels approaches in Scenario A2 - Part I.

Table 2
Labels mapping.

Scenario		Traffic-labels	PHB-labels [a]
A2/B	NonVPN	Mail	AF12
		Chat	
		FT	
		BRW	AF22
		STR	AF32
		P2P	
		VoIP	EF
	VPN	VPN-Mail	AF11
		VPN-Chat	
		VPN-FT	
		VPN-BRW	AF21
		VPN-STR	AF31
		VPN-P2P	
		VPN-VoIP	EF

[a] Drop probabilities: VPN Traffic has lower drop probability than NonVPN Traffic. (i.e AFx1 has lower drop probability than AFx2)

VPN and NonVPN Traffic). B dataset is a mixture of all PHB-labels ranging from AF11 to AF32 and its priorities numbers (Table 2). As suggest Habibi et al. in [8], different flow timeout (ftm) values were selected for testing (i.e. 15s, 30s, 60s, 120s).

Table 3 presents the setup for the experiment. Weka experimenter for testing each scenario was used. Both scenarios have the same cross validation settings and machine learning algorithms for modeling but different datasets. The A2 scenario has VPN and NonVPN for PHB-labels (i.e. AF11 to AF31 and AF12 to AF32, respectively). B scenario has all-in one PHB-Labels. The experiment was replicated in original dataset as well (i.e. Traffic-Labels). Finally, a T-Tester was performed in the algorithms for each dataset in order to get a trustworthy statistical result.

3.1. Results and analysis of scenario A2

The accuracy (i.e. Percent correct) in the Lashkari et al. work [8] using a single run for cross-validation and not T-Tester for Traffic-Labels indicated that C4.5 algorithm has a little better behavior than kNN algorithm. A little decrease in the precision for 15s to 120s was obtained. However, the numerical difference is not strong enough criteria for getting a statistical significance. To solve that, we have applied a T-Tester and trained other machine learning algorithms (See Fig. 5 parts a, b). As result, C4.5 is better than kNN and significantly better for all ftm values. This result initially validates what was obtained by work aforementioned. However, C4.5 has not the best behavior. Indeed, Bagging and Booting algorithms are significantly better than C4.5 but not significantly different between them (except for 60s ftm value in Non-VPN traffic wherein boosting is worse). Table 4 shows all results obtained by applying a statistical test based on Paired T-Tester (characters '*' and 'v' represent a statistically significant difference; lower scores and higher scores, respectively. Character '-' does not represent a statistically significant difference. C4.5 was selected as test base).

Our approach has the same behavior (i.e. PHB-Labels). C4.5 is significantly better than kNN for all ftm values, but worse in comparison with Bagging and Boosting (Fig. 5 parts c, d). Both Bagging and Boosting are significantly better for all ftm values but not significantly different between them (none exception is presented). Moreover, PHB-Labels has higher scores for all time out values than Traffic-Labels approach (See Fig. 6 parts a, b. 15s and 30s ftm are presented as higher results). Table 5 shows all results obtained by applying a statistical test based on Paired T-Tester (characters '*' and 'v' represent statistically significant difference; lower scores and higher scores, respectively. Character '-' does not represent statistically significant difference. C4.5 was selected as test base).

Finally, our approach has better behavior than Traffic-Label approach and continuing with the same trend wherein accuracy for shorter ftm values are better than accuracy for larger values (i.e. 15s for PHB-Labels in Non-VPN and 30s for PHB-Labels in VPN. Fig. 6 parts a, b). Moreover, the PHB-Labels have the best behavior for EF class both VPN and Non-VPN (Fig. 6 parts c, d. 15s for Rc in Non-VPN and 15s for Pr in VPN presented higher scores).

3.2. Results and analysis of scenario B

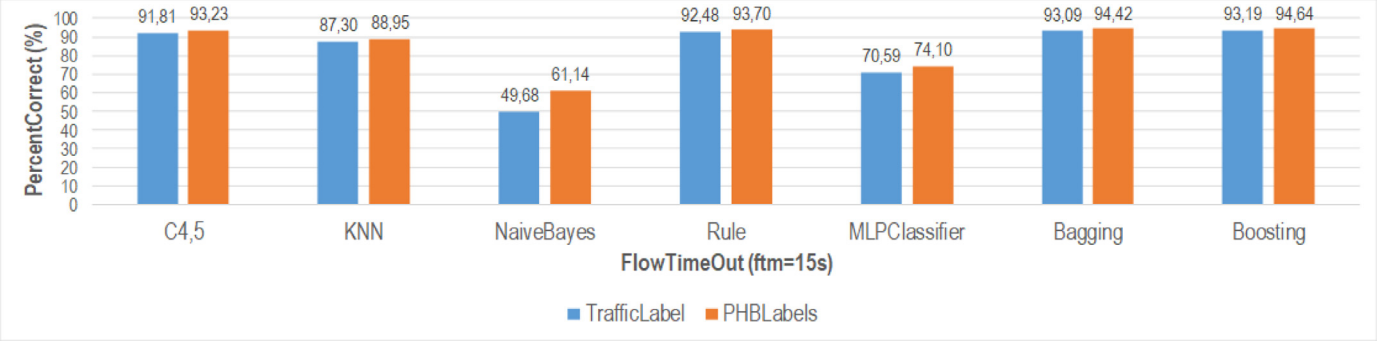
In this scenario, all PHB-Labels were mixed together in one dataset as in Traffic-Labels approach. As a result, 7 different classes were obtained unlike 14 classes in Traffic-Labels. PHB-Labels are more suitable for QoS Managing than Traffic-Labels. Labels as AF and EF are well-Known in a Diffserv architecture and allow routing all traffic in an effective manner.

The Accuracy achieved in PHB-Labels is 86.94% for Bagging in 15s ftm (See Fig. 7 part b). In the Traffic-Labels approach, the Bagging algorithm has the same behavior but its score is lower than PHB-Labels (See Fig. 7 part a). kNN, Naive Bayes, and Multilayer perceptron have worse behavior than C4.5, rules, Bagging, and Boosting for all ftm values. C4.5 and Rules are not significantly different but are significantly worse than Bagging and Boosting. Indeed, Bagging is significantly better than boosting and the other ones for all ftm values (See Tables 4 and 5). Hence, Bagging is more suitable for Traffic-Labels and PHB-Labels approaches. Again, the best results are presented for shorter ftm value in the overall accuracy and PHB-Labels approach has better scores than Traffic-Labels approach (Fig. 8, part c).

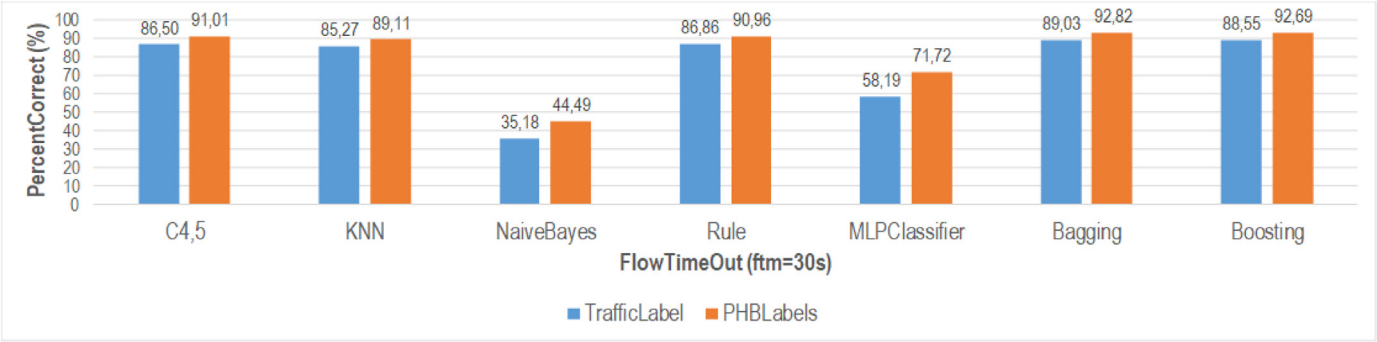
Precision and recall by class have a behavior a little different in each ftm for PHB-Labels. Precision in AF12 class has high scores for 15s and 120s (See Fig. 8 part a); bagging is significantly better than others algorithms for all ftm values, excepting for 30s and 60s wherein is not significantly different from boosting (for recall none change was detected). Recall in AF22 class has high scores for 60s and 120s (See Fig. 8 part b); bagging is significantly better than others algorithms for all ftm values, excepting for 15s and 30s wherein is not significantly different from boosting. EF and AF31 classes have the same behavior for bagging and boosting in all ftm values, having high scores for both precision and recall (but not

Table 3
Experimental setup.

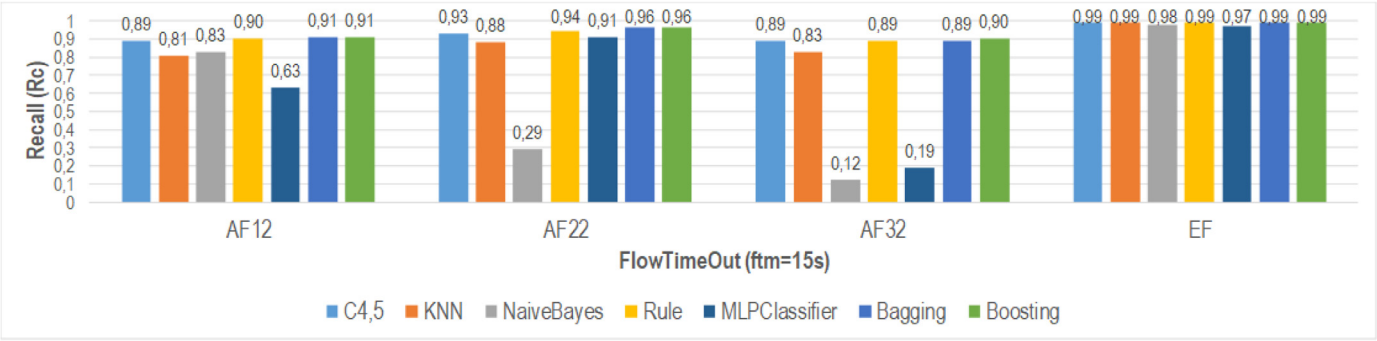
Approach	Machine learning algorithms/Weka	Scenario	Cross-Validation(Folds)	Run
Traffic/PHB-Labels	C4.5/J48 KNN (K=1)]/IBK Naive Bayes Rule(Part) Neuronal Networks/MLPClassifier Bagging (C4.5) Boosting (C4.5)/AdBoostM1	A2 and B	10	10



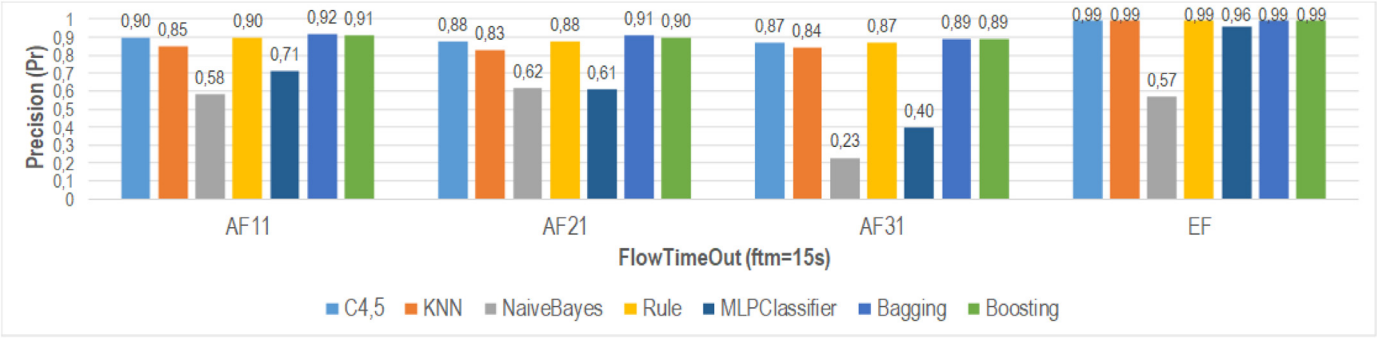
(a) NonVPN Percent Correct Traffic/PHBLabels



(b) VPN Percent Correct Traffic/PHBLabels



(c) NonVPN Recall PHB Labels



(d) VPN Precision PHB Labels

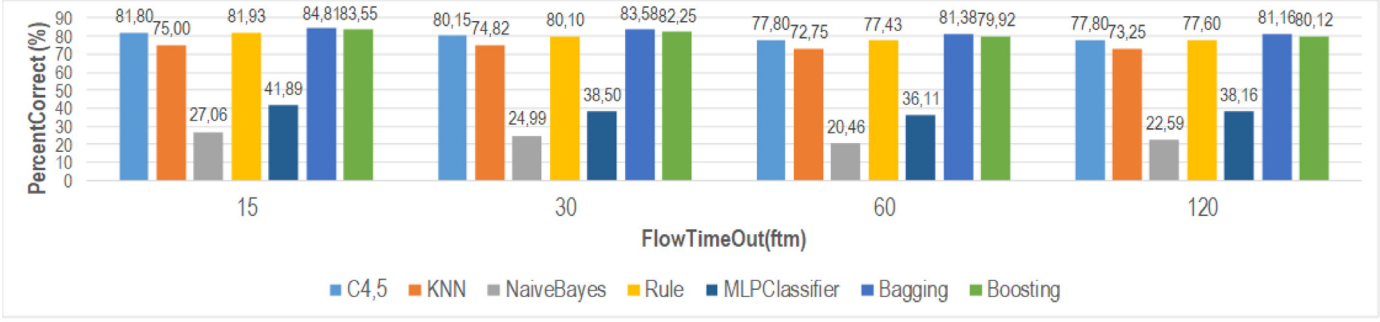
Fig. 6. Percent correct for Traffic-labels and PHB-labels approaches in Scenario A2 - Part II.

Table 4
Statistical significance by algorithms in Traffic-labels approach - C4.5 as test base.

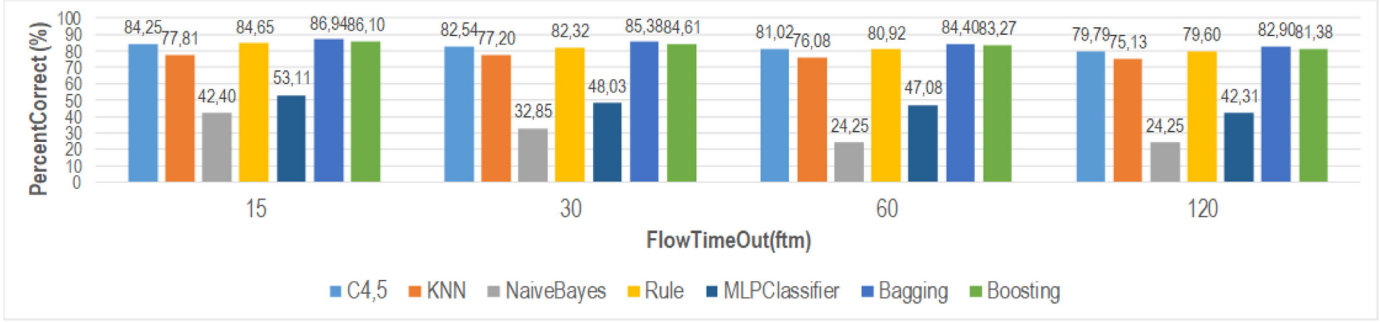
Scenario		ftm (s)	C4.5	KNN	Naive Bayes	Rule	Neuronal networks	Bagging	Boosting
A2	NonVPN	15	91.81	87.30*	49.68*	92.48 v	70.59*	93.09 v	93.19 v
		30	89.89	86.48*	44.87*	90.36-	67.66*	91.75 v	91.45 v
		60	88.06	84.05*	31.74*	88.58-	57.31*	90.63 v	89.33 v
		120	89.74	87.20*	39.53*	90.18-	69.17*	91.74 v	91.42 v
	VPN	15	85.91	82.52*	22.49*	86.09-	60.93*	88.28 v	87.68 v
		30	86.50	85.27*	35.18*	86.86-	58.19*	89.03 v	88.55 v
		60	84.15	82.21*	34.35*	84.22-	58.11*	86.66 v	85.97 v
		120	83.62	82.43*	38.86*	84.48-	54.90*	86.43 v	85.82 v
B	VPN-NonVPN	15	81.80	75.00*	27.06*	81.93-	41.89*	84.81 v	83.55 v
		30	80.15	74.82*	24.99*	80.10-	38.50*	83.58 v	82.25 v
		60	77.80	72.75*	20.46*	77.43-	36.11*	81.38 v	79.92 v
		120	77.80	73.25*	22.59*	77.60-	38.16*	81.16 v	80.12 v

Table 5
Statistical significance by algorithms in PHB-Labels approach - C4.5 as test base.

Scenario		ftm (s)	C4.5	KNN	Naive Bayes	Rule	Neuronal Networks	Bagging	Boosting
A2	NonVPN	15	9323	88.95*	61.14*	93.70-	74.10*	94.42v	94.64v
		30	91.35	88.18*	59.13*	91.60-	73.11*	92.89 v	92.95 v
		60	91.56	88.26	44.97*	91.89-	67.00*	93.45 v	92.98 v
		120	91.06	89.40*	47.73*	91.03-	72.62*	92.94 v	92.59 v
	VPN	15	90.89	87.69*	37.88*	91.11-	73.91*	92.80 v	92.36 v
		30	91.01	89.11*	44.49*	90.96-	71.72*	92.82 v	92.69 v
		60	87.88	85.87*	48.03*	87.64-	66.72*	90.34 v	90.00 v
		120	87.78	85.02*	48.10*	88.01-	64.75*	89.42 v	89.26 v
B	VPN-NonVPN	15	84.25	77.81*	42.40*	84.65-	53.11*	86.94 v	86.10 v
		30	82.54	77.20*	32.85*	82.32-	48.03*	85.38 v	84.61 v
		60	81.02	76.08*	24.25*	80.92-	47.08*	84.40 v	83.27 v
		120	79.79	75.13*	29.66*	79.60-	42.31*	82.90 v	81.38 v



(a) NonVPN/VPN Percent Correct Traffic Labels



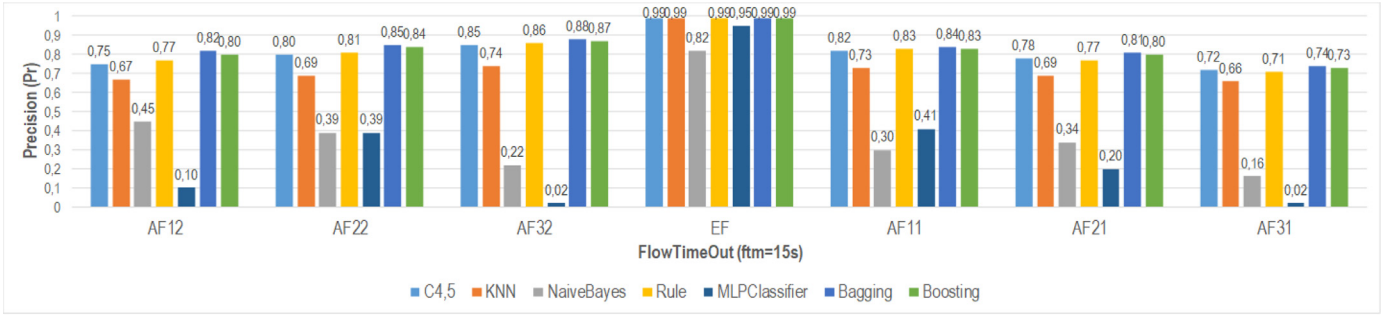
(b) NonVPN/VPN Percent Correct PHB Labels

Fig. 7. Percent correct for Traffic-labels and PHB-labels approaches in Scenario B - Part I.

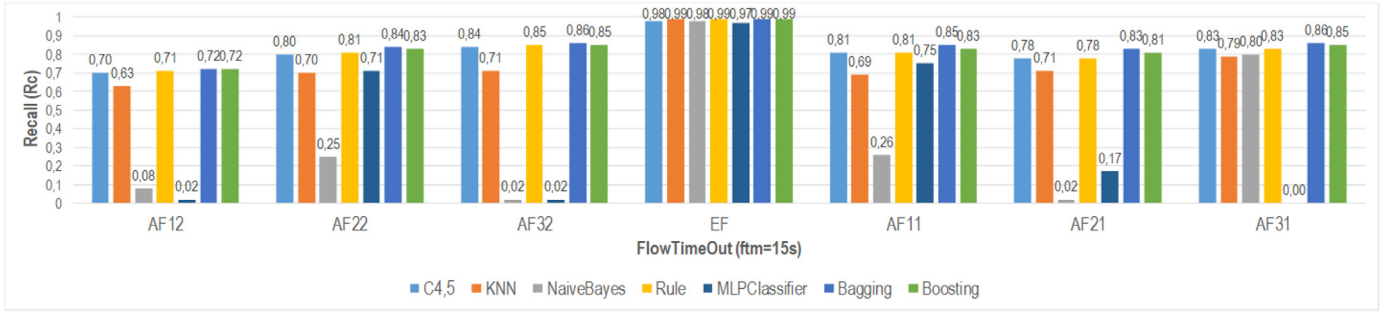
significantly different between them). Besides, higher scores are presented for EF class (Fig. 8 parts a, b). For the other classes bagging is significantly better than boosting in at least one ftm value both precision and recall.

4. Discussion

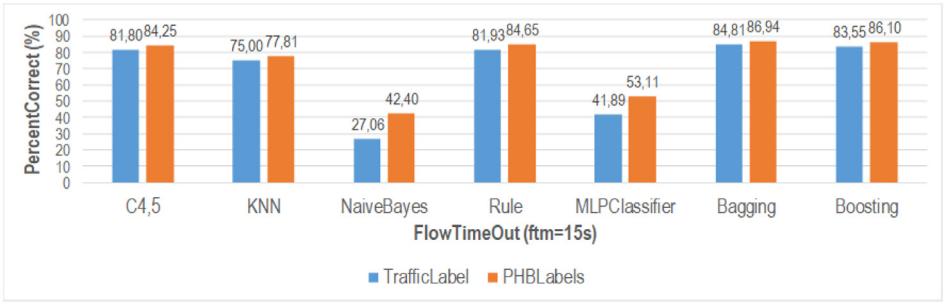
A QoS-marked traffic was obtained as from characterized VPN and Non-VPN traffic, wherein PHB-Labels are more suitable for



(a) VPN/NonVPN Precision PHBLabels



(b) VPN/NonVPN Recall PHBLabels



(c) NonVPN Percent Correct Traffic/PHBLabels

Fig. 8. Percent correct for traffic-labels and PHB-labels approaches in Scenario B - Part II.

managing the network traffic than Traffic-Labels in [8]. Indeed, PHB-Labels are well-known in Diffserv networks for routing traffic in a proper manner. Moreover, PHB-Labels can be mapped to EXP field as explained in [15] for a MPLS Network.

A statistical significance analysis suggests that Bagging and Boosting have the best accuracy in the A2 and B scenarios both Traffic-Labels and PHB-Labels approaches. Bagging is significantly better than boosting in B scenario but not significantly different in A2 scenario, except for 60s-NonVPN wherein bagging is better in Traffic-Labels approach. Hence, bagging is the best classifier for Traffic-Labels and PHB-Labels in two scenarios. Besides, PHB-Labels approach have better results than Traffic-Labels approach for all ftm values in each scenario and for all algorithms analyzed. On the other hand, MLPClassifier has to be tuning in order to achieve higher scores or any domain specific implementation has to be done for improving the accuracy.

QoS-Classifier has been proposed as a mechanism for controlling traffic in a proper manner and QoS managing through PHB-Labels. To this end, two tasks were defined and implemented. First one, an analyzing policies task takes into account the network policies for marking incoming traffic as a well-known network label. However, this task is network administrator knowledge dependent and domain-specific. To solve this, a domain ontology would be a good option for having parametrized policies in each domain wherein marking process can be more effective. Second one, a labeling task modifies one traffic label to another one for traffic control in the network. Again, labeling task is policing dependent and manual execution. A recommender system based on traffic behavior can suggest a well-known label for any specific domain policy.

Finally, the results suggest that QoS-Classifier implementation be bagging-based learning model in order to achieve the best behavior for Traffic Classification and Traffic Marking processes in DiffServ architecture.

5. Conclusion and future works

In this paper, the effectiveness of time-related features for VPN and Non-VPN traffic classification was confirmed. Moreover, we found that bagging and boosting are significantly better than C4.5 in the Traffic-Labels approach, which only two machine learning algorithms were performed (i.e. kNN and C4.5).

A QoS-Classifier with two tasks for traffic treatment on the network has been proposed. This proposal took into account the PHB-DiffServ concept and specific domain policies for traffic marking, as the well-known AF and EF labels. Hence, the QoS-Classifier output (i.e. QoS-Maked) allows routing of all traffic in an effective man-

ner. Finally, bagging-based learning model for QoS-Classifier has achieved better results for all ftm values in PHB-Label approach, besides our approach has better behavior in overall accuracy than Traffic-Labels approach for all machine learning algorithms in each ftm values both A2 and B scenarios. Higher scores with a statistically significant difference were obtained in scenario A2 with 94,42% and 92,82% of accuracy for Non-VPN and VPN traffic, respectively.

As future work, a domain ontology can be defined for recommender systems proposals. The ontology will allow to having a labeling recommender system for an incoming network traffic with time-related features. Besides, a mechanism based on incremental learning is suitable for implementing the policing stage in QoS managing process, mainly for being the network traffic so dynamic and requiring differenced treatment for each new telco services. Finally, due to the use of default parameters in each algorithm for testing, a tuning or implementing specific one is necessary for reaching higher scores and better behavior (e.g a specific Neuronal Network).

Acknowledgments

The authors are grateful with the researches groups: Telematics Engineering Group (GIT) of the University of Cauca and Systems Control, Learing and Optimization (CAOS) of the Carlos III University of Madrid for the technical support. In addition, the authors are grateful to the Administrative Department of Science, Technology, and Innovation -COLCIENCIAS- for funding the Ph.D. program in which this work was developed (Scholarship Program No. 727-2015). This work has been also supported by the Spanish Ministry of Economy, Industry and Competitiveness (Projects TRA2015-63708-R and TRA2016-78886-C3-1-R).

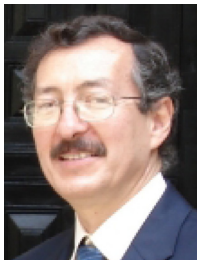
References

[1] ITU-T E.800, Definition of Terms Related to Quality of Service, Tech. report, 2008. URL https://www.itu.int/rec/T-REC-E.800/en.
[2] F. Ertam, E. Avc, A new approach for internet traffic classification: GA-WK-ELM, Measurement 95 (2017) 135–142, doi:10.1016/j.measurement.2016.10.001.
[3] N.F. Huang, G.Y. Jai, H.C. Chao, Y.J. Tzang, H.Y. Chang, Application traffic classification at the early stage by characterizing application rounds, Inf. Sci. (NY.) 232 (22) (2013) 130–142, doi:10.1016/j.ins.2012.12.039.
[4] R.-y. Wang, Z. Liu, L. Zhang, Method of data cleaning for network traffic classification, J. China Univ. Posts Telecommun. 21 (3) (2014) 35–45, doi:10.1016/S1005-8885(14)60299-5.
[5] C.J. Liu, Tracking DSCP marking of packets in a QoS enabled triple-play IP network, in: 3rd International Conference on Networking and Services, ICNS 2007, 2007, doi:10.1109/ICNS.2007.121.
[6] M. Peresse, J.M. Bonnin, Enabling QoS using packet marking and scheduling in a multi-homed NEMO mobile router with GNU/Linux, in: 2009 9th International Conference on Intelligent Transport Systems Telecommunications, ITST 2009, 2009, pp. 58–62, doi:10.1109/ITST.2009.5399382.
[7] E. Frank, M.A. Hall, I.H. Witten, The Weka Workbench. Online Appendix for "Data Mining: Practical Machine Learning Tools and Techniques", Morgan Kaufmann, fourth ed., 2016. URL https://www.cs.waikato.ac.nz/ml/weka/index.html.
[8] A.H. Lashkari, G. Draper-gil, M. Saiful, A.A. Ghorbani, Characterization of encrypted and VPN traffic using time-related features, in: O. Camp, S. Furnell, P. Mori (Eds.), Proceedings of the 2nd International Conference on Information Systems Security and Privacy (ICISSP 2016), Rome, 2016, pp. 407–414, doi:10.5220/0005740704070414. URL http://www.icissp.org/.
[9] Cisco Systems, Quality of Service Configuration Guide , Release 4 . 0 (4) SV1 (1), Tech. Rep. 4, Cisco Systems, 2009.
[10] Cisco Systems, WAN and Application Optimization Solution Guide Cisco Validated Design, Tech. Rep. August, 2008. San Jose-USA.
[11] M. Finsterbusch, C. Richter, E. Rocha, J.A. Müller, K. Hänßgen, A survey of payload-based traffic classification approaches, IEEE Commun. Surv. Tutorials 16 (2) (2014) 1135–1156, doi:10.1109/SURV.2013.100613.00161.
[12] IANA, Service Name and Transport Protocol Port Number Registry, 2016. URL http://www.iana.org/.
[13] Cisco Systems, DiffServ - The Scalable End-to-End QoS Model (August) (2005) 1–19. URL http://www.cisco.com.
[14] K. Nichols, S. Blake, Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers, Tech. rep., IETF-RFC2474, 1998. URL https://www.rfc-editor.org/info/rfc2474.
[15] L. Wu, B. Davie, S. Davari, P. Vaananen, R. Krishnan, P. Cheval, J. Heinanen, Multi-Protocol Label Switching (MPLS) Support of Differentiated Services Status, Tech. rep., IETF-RFC3270, 2002. URL https://www.rfc-editor.org/info/rfc3270.
[16] J. Erman, A. Mahanti, M. Arlitt, Internet traffic identification using machine learning, in: Global Telecommunications Conference, 2006. GLOBECOM '06. IEEE, 2006, pp. 1–6, doi:10.1109/GLOCOM.2006.443.
[17] Internet.org, State of Connectivity-A Report on Global Internet Access, Tech. rep., Internet.org, 2015, doi:10.1017/CBO9781107415324.004.
[18] A. Alshalan, S. Pisharody, D. Huang, A survey of mobile VPN technologies, IEEE Commun. Surv. Tutorials 18 (2) (2016) 1177–1196, doi:10.1109/COMST.2015.2496624.
[19] F. Palmieri, VPN scalability over high performance backbones evaluating MPLS VPN against traditional approaches, in: Proceedings - IEEE Symposium on Computers and Communications, 2003, pp. 975–981, doi:10.1109/ISCC.2003.1214243.
[20] Global Web Index, The missing billion: how analytics is wiping the emerging world off the map, 2014. URL https://www.globalwebindex.net.
[21] J. Babiarz, K. Chan, F. Baker, Configuration Guidelines for DiffServ Service Classes, Tech. rep., IETF-RFC4594, 2006. URL https://www.rfc-editor.org/info/rfc4594.
[22] D. Grossman, New Terminology and Clarifications for Diffserv, Tech. rep., IETF-RFC3260, 2002. URL https://www.rfc-editor.org/info/rfc3260.

Julian Andres Caicedo Muñoz received the degree in electronic and telecommunications engineering and master degree in telematics engineering at University of Cauca, Colombia, in 2010 and 2014 respectively. Currently he is a Ph.D. student in Telematics Engineering at the University of Cauca and Computer Science and Technology at Carlos III University of Madrid. His research interests focus on networking, quality of service, quality of experience and machine learning.

Agapito Ledezma is an associate professor in the Department of Computer Science at Carlos III of Madrid University. He received a B.S. degree from Universidad Latinoamericana de Ciencia y Tecnología in 1997. He received his Ph.D. degree in computer science from Carlos III University in 2004. His research interests center on machine learning, activity recognition, intelligent agents and advanced driving assistant systems. He has published over 80 journal and conference papers mainly in the field of artificial intelligence and machine learning.

Juan Carlos Corrales received the degree in electronic and telecommunications engineering and master's degree in telematics engineering from the University of Cauca, Colombia, in 1999 and 2004 respectively, and the Ph.D. in Computer Science from the University of Versailles Saint-Quentin-en-Yve-lines, France, in 2008. Presently, he is a full professor and leads the Telematics Engineering Group at the University of Cauca. His research interests focus on service composition and data analysis.



Alvaro Rendón received the degree in electronic engineering and master's degree in telematics engineering from the University of Cauca, Colombia, in 1979 and 1989 respectively, and the Ph.D. degree in telecommunications engineering from the Technical University of Madrid in 1997. At present, he is a full professor and director of Doctoral Program in Telematics Engineering at the University of Cauca. His research interests focus on telecommunications, e-Health, Real-Time applications and Tele-education