Efficiency of Supervised Machine Learning Algorithms in Regular and Encrypted VoIP Classification within NFV Environment

Gjorgji ILIEVSKI [1], Pero LATKOSKI [2]

[1] Makedonski Telekom AD Skopje, Kej 13-ti Noemvri 6, 1000 Skopje, RN Macedonia
[2] Faculty of Electrical Engineering and Information Technologies, Ss Cyril and Methodius University, Rugjer Boshkovic 18, 1000 Skopje, RN Macedonia

gjorgji.ilievski@telekom.mk, pero@feit.ukim.edu.mk

Submitted October 15, 2019 / Accepted December 18, 2019

Abstract. *Cloudification of all computing environments is an undergoing process. The process has overpassed the classical Virtual Machines (VM) and Software-Defined Networking (SDN) approach and has moved towards dockizing, microservices, app functions, network functions etc. 5G penetration is another trend, and it is built on such platforms. In this environment we are investigating the efficiency of supervised machine learning algorithms for classification of regular and encrypted Voice over IP (VoIP) traffic that 5G relies on, within a virtualized Network Functions Virtualization (NFV) environment and an east-west based network traffic. We are using statistical methods for classification of network packets without the need of inspecting the payload data and without the source, destination and port information of the packets. The efficiency is analyzed from a point of precision of the classification, but also from a point of time consumption, as adding delay to the original traffic may cause a problem, especially within 5G environments where packet delay is crucial.*

Keywords

VoIP, classification, supervised algorithms, machine learning, NFV, 5G

1. Introduction

VoIP communication is established as one of the most common services used by the general population. Multiple systems and applications are providing peer-to-peer VoIP services, both as an unencrypted and encrypted traffic. As the systems are migrating towards cloud virtualization, using public (e.g. Amazon's AWS, Microsoft's Azure, Google Cloud), mixed or private clouds, the classification and deep packet inspection (DPI) of traffic is a standard requirement for any organization. It allows implementation of high security within the network, application and data monitoring, as well as network optimization and provision of Quality of Service (QoS). The trends of machines and appliances virtualization have moved towards using docker instances, microservices, application functions and networking functions [1]. 5G technology is relying on these services and its data within the cloud is moving in the east-west direction, not leaving the virtual layer. East-west traffic means that the network data is moving inside the datacenter, usually managed by the cloud operator, between the servers and within the virtual layer. Applications such as Skype and Viber (and many more) are using encryption which makes DPI even more problematic. The latency of the network packets must be minimal [2].

Due to the previous, the paper is exploring the efficiency of the already established supervised machine learning algorithms within a scenario, where classification and speed of the algorithm are equally important. The challenge is to monitor and capture the "hidden" east-west traffic [3], where NFV elements are already operating and to classify it, but also minimize the expected latency added by the algorithm for classification. 5G networks are using VoIP as a standard and the 5G specification calls for user plane latency of just 1 ms for ultra-reliable low-latency communications (URLLC) [2]. The study that we have conducted provides a novel scenario that resembles the target architecture of the systems, which will be built upon 5G and NFV elements, as well as in terms of a variety of ML algorithms that we test and the DPI efficiency evaluation. It has an importance from functional, security, controlling, QoS and management aspects.

There are multiple works that are using machine learning approach for packet inspection [4], [5], [6], [7]. ML algorithms have been used in traditional networks DPI for a long time and their behavior is proven by many studies [8], [9], [10], as well as in practice. Best to our knowledge, there is no research that is focused on encrypted and unencrypted VoIP within NFV environment from the point of precision and speed of the supervised ML algorithms.

In particular, our paper focuses on 6 different ML algorithms: Bayes Net, Naïve Bayes, J48 (as an implementation of C4.5), K-Nearest Neighbors (K-NN), Decision Tree and AdaBoost. The classifications are made with Weka [10]. These algorithms are chosen due to the fact that they are the most common ML algorithms used in traditional networks, and are proven to be reliable in practice. Due to the encryption mechanisms used for VoIP network traffic and the specifics of the east-west traffic, using traditional DPI mechanisms is impossible. In our analysis we are not using the payload data of the network packets, the communication ports, IP addresses and MAC addresses of the source and destination entities. We are working with the statistical features of the packets and the packet flows in general, to create a training set for the ML algorithms. After the training, we are testing the precision, as well as the speed of the algorithms on the testing set. We have created a testing environment in which the traffic is sniffed inside an open vSwitch, directly listening on the east-west traffic without introducing an external probe or an SDN device that will collect the data. All network traffic is observed as a whole, including the communication among network elements, as well as management networking data, because this is a realistic scenario in practice. VoIP (both encrypted and unencrypted) is recognized successfully under these conditions. Because ML algorithms are consuming CPU within the virtualized environments and are adding to the packet latency, the speed of the ML algorithms is very important aspect to the overall efficiency.

The reminder of the paper is organized as follows: we briefly go through the related work on the subject in Sec. 2, after which the experimental setup and the dataset creation is explained in Sec. 3. The results analysis and conclusion follow in Sec. 4 and Sec. 5, respectively. The future work is at the end.

2. Related Work

Deep Packet Inspection is a service that is crucial in digital environments. As 5G ambition is to unify many services through one platform thus providing basis for further development of systems and applications, it is of high importance to have a valid DPI that will provide viable results and in the same time that will not increase the network latency. There are researches that are focused on DPI in SDN [11], [12], [13], while others are focused on the security aspects of DPI [14], [15], often proposing an introduction of probes or SDN appliances within the network. Network traffic classification in traditional networks is researched in the works of [16] and [17], but they do not consider the use of ML algorithms in NFV environment.

The authors of [18] propose a design of virtual network functions to flexibly select and apply the best suitable machine learning classifiers at run time. They analyze multiple ML algorithms, such as K-Nearest Neighbor, Support Vector Machine, Decision Tree, Ada-Boost, Naive Bayes and Multi-Layer perception. The experimental results show an improvement of 13% in the accuracy of the flow classification using the proposed NFV.

Vergara-Reyes et al. [4] introduced an NFV environment in which different types of TCP traffic are generated. Network packets are captured and analyzed using three different ML algorithms: J48, Naive Bayes and Bayes Net, to provide a benchmark on the performance of the algorithms. Statistical parameters of the individual packets are taken into consideration to prepare the training and testing sets for the ML algorithms. Three different datasets are created: traditional, virtual and combined to better characterize the traffic in the NFV based networks. On the other hand, in our research we are working with statistical parameters of packet flows within an NFV environment typical for cloud platforms, focusing on both encrypted and unencrypted VoIP traffic.

Alshammari et al. [5] covers the DPI of VoIP traffic within traditional networks using real data from existing network environments with different topology (with and without firewalls) and different access methods (WiFi or Ethernet), to evaluate the precision of three ML algorithms: C5.0, ADA Boost and GP Classifier. Subset sampling technique and statistical analysis test for precision and false positive rates is performed for evaluation of the ML algorithms. Their research has shown that C5.0 achieved the highest performance with the highest precision and the lowest false positive rate. Our work is focused on cloud-based environment with an accent on an NFV and unsupervised ML algorithms.

The work [19] proposes a runtime predictive analysis system that runs in parallel with the existing reactive monitoring systems within a network operator. Deep learning-based approach is used to identify anomaly events from NFV system logs, in order to identify faulty conditions and to take necessary pro-active actions within the network.

In [20] Machine Learning based classification of multi-service internet traffic is evaluated from the point of resource consumption in terms of CPU and memory consumption. Out paper is complementing this research as we are observing the time needed for various ML algorithms to perform the classification job.

The authors of [21] are researching the effect of NFV elements placement on the network traffic, especially on the increase or decrease of the volume of the processed traffic. An algorithm that determines the flow path and then proposes a Least-First-Greatest-Last routing is developed.

The work of Bonfiglio et al. [22] is on Skype and its generated traffic specifics regardless of the underlying architecture. It deals with two different approaches for revealing Skype encrypted traffic in real time, based on the statistical parameters of the generated packets. DPI and flow correlation are used to assess the effectiveness of the proposed approaches.

In [23], ML algorithms, big data analytics platforms, SDN and NFV elements are used to build a comprehensive

framework for developing future 5G Self-Organizing Network (SON) applications, as well as a framework for clustering, forecasting, and managing traffic behaviors for a huge number of base stations with different statistical traffic characteristics of different types of cells (GSM, 3G, 4G). Traffic flows are analyzed and SDN-based QoS control is implemented to enable bandwidth guarantees for each application. In this case study, 5 different ML algorithms are used to classify accurately mobile applications. Different types of encrypted traffic were used for classification, showing that Random Forrest algorithm has the best overall performance relative to the others tested algorithms. Compared to this work, we focus on VoIP and ML algorithm performance in terms of accuracy and speed in an environment, where NFV elements are deployed, a scenario feasible for future 5G development.

In general, our work is introducing similar testing setup as [4] and [23], adding new elements in the testing environment, like virtual hosts with internet access, from which VoIP is generated. Encrypted and unencrypted UDP based VoIP traffic, along with various random TCP and UDP traffic is generated and classified, using not only the packets themselves, but also the packet flows statistics. Skype and Viber are chosen as most widely used peer-to-peer VoIP clients that engage encryption. Furthermore, the paper proposes a novel testbed setup in the context of 5G that is built upon virtualized environment in which NFV elements are used. This is an expected setup for the systems that work in the virtualized plane and are using NFV and 5G communication. The data is analyzed directly into the network data flow between the NFV elements, without introduction of physical or SDN probes. As shown in the next section, both precision and speed of six different algorithms is evaluated in terms of which algorithm performs the best within the target scenario.

3. Experimental Environment and Dataset Creation

To simulate the east-west traffic within a virtualized NFV based network, we have created an experimental environment in which Oracle VirtualBox [24] is installed on Ubuntu 18.04 Server single physical host. Open vSwitch (OVS) [25], [26] is installed on the host for network communication, allowing to intercept and sniff all network traffic going through it. It allows to capture all transferring network packets. We are using Wireshark and tshark [27] for network capture.

Figure 1 shows the experimental environment that is used. Mininet [28] network simulator on 2 different VMs is used to create 2 networks with multiple hosts, switches and links among them. The networks have private IP addresses and are able to communicate with each other using GRE tunneling, within the OVS. Some of the simulated hosts are NAT-ed and have internet access. Simulated networks are controlled using Ryu Controller [29] in a dedicated controller VM.

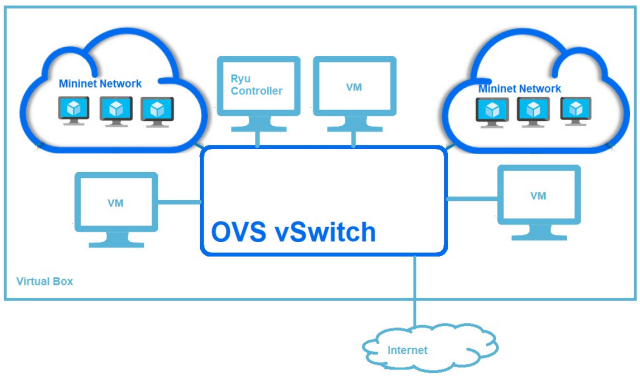

Fig. 1. NFV experimental environment.

To generate TCP and UDP traffic, a Distributed Internet Traffic Generator (D-ITG) [30] is used within the hosts created into the Mininet networks. D-ITG produces traffic at packet level, replicating appropriate stochastic processes for both IDT (Inter Departure Time) and PS (Packet Size) random variables.

Three additional VMs, also connected to the OVS, with Skype and Viber clients installed on them, simulate the encrypted UDP VoIP traffic in a peer-to-peer communication. Random audio calls are performed among clients with random duration.

The traffic goes inside the OVS in an east-west direction. Internet is needed for initial contact to Skype and Viber servers, after which the communication is entirely inside the OVS in the east-west direction.

We have simulated 50 different network traffic scenarios generating TCP and UDP streams using D-ITG. Encrypted VoIP was generated using Skype and Viber clients. To classify the network traffic, 3 labels were used: *VOIP* – for unencrypted VoIP, *EncVOIP* – for encrypted VoIP and *OTHER* – for all other network packets.

The experiments were conducted in an interval from 4 to 20 minutes. Every experiment produced one dataset. In every experiment, different network scenario was simulated, with different Mininet hosts used to generate and to receive the network traffic. The average length of an experiment was 625 sec. The average number of packets was 1.262.375 and the average number of flows was 4090. Skype and Viber calls were conducted randomly with a length form 10 seconds to 3 minutes.

When traffic is observed, it can be seen that there are multiple flows within the OVS, from the inter-virtual hosts traffic and from the management traffic generated by the hypervisor and the controller. The features that are not valid in NFV environment, as well as within encryption scenarios, are not used. Such features are the source and destination IP and MAC addresses, and the communication port which are not distinguishable when encryption is used. To generate the datasets, we have chosen creation of flow-based data sets. Similar to [5], we define a flow as a bi-directional connection between two hosts. TCP flows are ended either by flow time-out or by connection tear-down,

	Abbrevation	Feature
1	proto	transaction protocol
2	rate	packets per second
3	srate	source packets per second
4	drate	destination packets per second
5	sintpkt	source interpacket arrival time
6	dintpkt	destination interpacket arrival time
7	sjit	source jitter
8	djit	destination jitter
9	mdoffset	mean of the data offset values of the packets in the flow.
10	smeansz	mean of the flow packet size transmitted by the source
11	dmeansz	mean of the flow packet size transmitted by the destination
12	smaxsz	max packet size for source
13	dmaxsz	max packet size for destination
14	sminsz	min packet size for source
15	dminsz	min packet size for destination

Tab. 1. Flow features.

while UDP flows are ended by flow time-out. We used Argus [31] for generating network flows from the captured traffic.

Through observation of the traffic and from experience within traditional networks, the flow features explained in Tab. 1 were selected as attributes for the characterization of the flows. The main goal is to classify the network packet flows based on the statistical characteristic of these attributes, and to make the classification fast enough, so that minimal delay of the packet is introduced.

Weka [10], [32] was used for the processes of training and testing of the prepared datasets. We were using 2/3 vs 1/3 split method on each of the datasets for training vs testing set, accordingly. To select the relevant attributes within the dataset, we used the AttributeSelectedClassifier with Ranker as a search method and InfoGainAttributeEval as an evaluator that determines the gain of information that the features carry with the respect to our classification. In such a way, only attributes that carry more information are selected, which reduces the entropy in the dataset. With the prepared datasets, training and testing of the above-mentioned ML algorithms was performed.

The next section explains the experimental results and analyzes their meaning.

4. Results and Analysis

As explained in the previous section, we have prepared 50 experimental datasets and we have tested 6 different ML algorithms onto them. The final performance metrics are the mean value and the statistical standard deviation of the algorithms precision in the classification, but also the True Positive Rate (TP Rate) and False Positive Rate (FP Rate) of the classification, which combined give the classification performance of the algorithms.

True Positive *TP* is the number of instances that are truly identified of a class.

False Positive *FP* is the number of instances that are falsely identified of a class.

True Negative *TN* is the number of instances that are truly identified that are not of a class.

False Negative *FN* is the number of instances that are falsely identified that are not of a class.

Precision of the algorithm [32] is defined as the proportion of instances that are correctly identified in a class, divided by the total instances classified as that class.

$$Precision = TP/(TP + FP). \quad (1)$$

TP Rate is the proportion of the instances that are correctly classified and the total instances truly of that class.

$$TP\ Rate = TP/(FN + TP). \quad (2)$$

FP Rate is the proportion of the instances that are wrongly classified and the total instances truly of that class.

$$FP\ Rate = FP/(FP + TN). \quad (3)$$

The results are visually represented in Fig. 2. The best classification is performed by the algorithms that have higher Precision and TP Rate, and lower FP Rate. Figures 3, 4 and 5 show them individually for the ML algorithms in focus.

Table 2 shows the results for the 6 ML algorithms. Mean value and statistical standard deviation of the Precision, TP Rate and FP Rate for the three classes in the 50 datasets were calculated.

When comparing the results for Encrypted VoIP and VoIP traffic, one can see that Decision Tree and Bayes Net algorithms show the best results, with the highest precision, high TP rate and low FP rate, followed by J48 and K-Nearest Neighbor. Naive Bayes and AdaBoost performance are not so good especially in the False Positive Rate that shows us that those algorithms are classifying other traffic as VoIP and EncVoIP.

The comparison of the mean values of the precision in Tab. 2 shows that Bayes Net has the greatest overall precision, 1.35% higher than J48 and 1.65% higher than the Decision Tree ML Algorithm. But in the same time Decision Tree has 0.24% better TP Rate than Bayes Net and 0.73% better than J48. Even more important, Decision Tree has the lowest FP Rate, which is for 1.54% lower than both of them.

AdaBoost performs the worst in terms of the FP rate, with 87.7% higher value than Decision Tree. Naive Bayes has 51.2% higher FP rate than Decision Tree.

K-Nearest Neighbor algorithm is in the middle with 4.78% lower precision, 1.25% lower TP rate and 4.2% higher FP rate than Decision Tree.

The percentages have been calculated on the mean values of the precision, TP Rate and FP Rate of the three classes explored.

The second characteristic that is important for the overall efficiency is the time interval needed for the algorithms to perform the classification. The two metrics combined will give the whole picture needed to evaluate the algorithms. The time in our case is relative to our experimental environment, but the comparison is relevant due to the same experimental environment under which all measurements have been done and the same datasets used for

		OTHER	***VOIP***	***EncVOIP***
J48	Precision	0.996±0.002	0.975±0.051	0.962±0.011
	TP Rate	0.998±0.002	0.963±0.044	0.899±0.019
	FP Rate	0.061±0.031	0.001±0.003	0.001±0.001
BayesNet	Precision	0.997±0.002	1.000±0.000	0.976±0.033
	TP Rate	1.000±0.001	0.960±0.080	0.915±0.077
	FP Rate	0.062±0.075	0.000±0.000	0.001±0.001
Naive Bayes	Precision	0.996±0.003	1.000±0.000	0.481±0.504
	TP Rate	0.761±0.276	0.960±0.080	0.928±0.070
	FP Rate	0.054±0.034	0.000±0.000	0.235±0.271
Decision Tree	Precision	0.997±0.001	0.965±0.047	0.962±0.044
	TP Rate	0.997±0.003	0.984±0.020	0.900±0.061
	FP Rate	0.053±0.008	0.002±0.002	0.001±0.001
AdaBoost	Precision	0.973±0.019	0.172±0.344	1.000±0.000
	TP Rate	0.993±0.014	0.250±0.500	0.841±0.063
	FP Rate	0.448±0.287	0.007±0.014	0.000±0.000
K-NN	Precision	0.997±0.002	0.912±0.084	0.922±0.042
	TP Rate	0.996±0.002	0.943±0.064	0.906±0.064
	FP Rate	0.071±0.058	0.002±0.001	0.002±0.001

Tab. 2. Classification results.

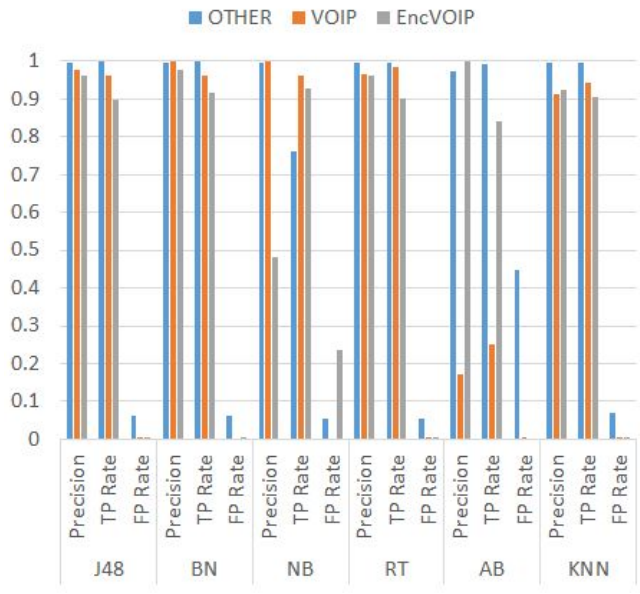

Fig. 2. Classification performance results.

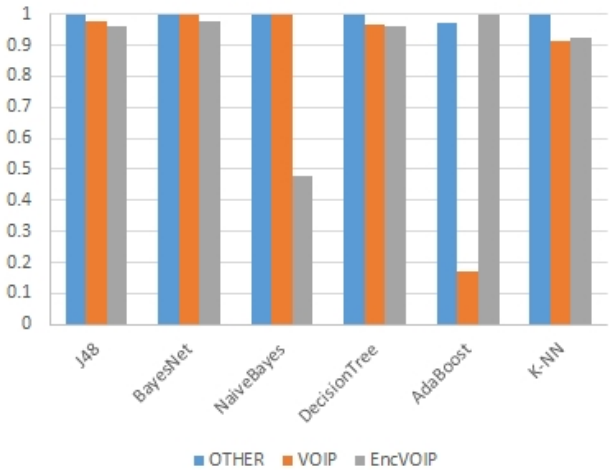

Fig. 3. Precision of ML algorithms.

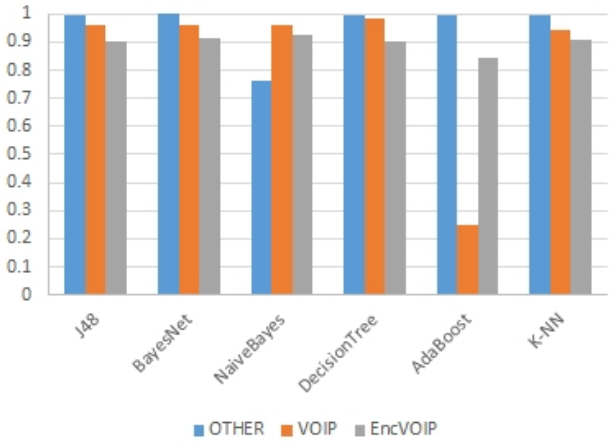

Fig. 4. TP Rate of ML algorithms.

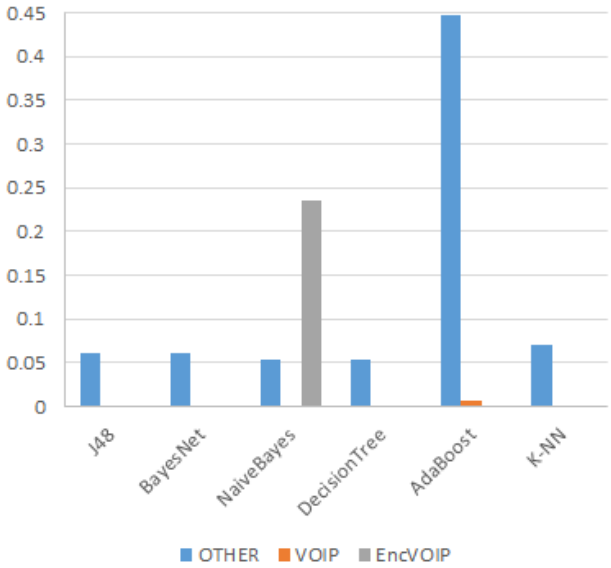

Fig. 5. FP Rate of ML algorithms.

every algorithm. Using faster or multiple machines for classification can significantly speed up the time needed for evaluation, but the ratio for the comparison of the algorithms is expected to stay the same. Due to this "resource

spending" of the algorithms, efficiency from the point of time needed for classification is very important.

Table 3 shows that AdaBoost algorithm is the fastest, taking only 0.8% of the time spent by the K-Nearest Neighbor. Due to the poor classification performance of AdaBoost, the second fastest algorithm – Decision Tree has the best overall efficiency for classifying encrypted and un-encrypted east-west based VoIP traffic within NFV based environment, taking only 1.66% of the time needed by K-NN. It is followed by Bayes Net, as well as J48 that also have good average time required to perform the classification. Because these two algorithms are also performing well within the classification, their overall performance is satisfactory. This is visually interpreted in Fig. 6.

K-Nearest Neighbor algorithm has good classification performance, but the time needed for classification is very high. We need to mention that the results shown here were using 1 nearest neighbor, but experiments using more (2 and 3) nearest neighbors have shown similar or worst performance in the experiments.

Naive Bayes has shown a poor classification performance and when compared relatively to the other algorithms, it requires a longer time for classification.

In context of 5G strict requirements for low latency, closer observation of the results shows that KNN and Naive Bayes need more time for classification than the rest of the ML algorithms. AdaBoost is performing badly in the terms of False Positive instances, while the others have satisfying classification performance, while the speed is the decisive element for their usage within a 5G scenario.

Algorithm	*AVERAGE TIME in seconds*
J48	0.036
BayesNet	0.026
Naive Bayes	0.069
Decision Tree	0.007
AdaBoost	0.004
K-NN	0.447

Tab. 3. Average time interval needed for classification.

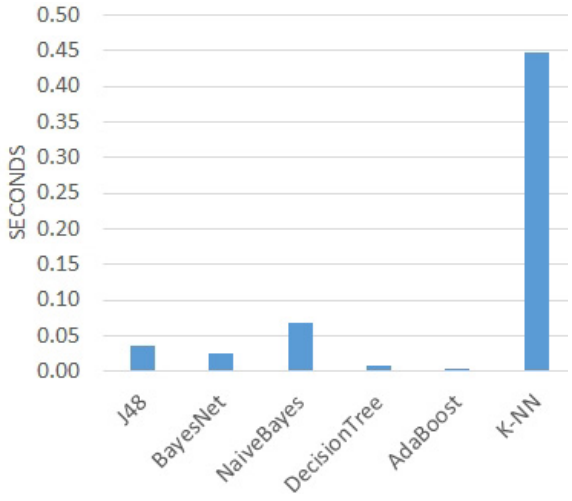

Fig. 6. Average time needed for classification.

5. Conclusion and Future Work

The paper aims to compare the efficiency of six different supervised machine learning algorithms in classifying VoIP and encrypted VoIP network traffic in a situation where the network traffic is flowing inside a virtualized environment where NFV elements are placed. This scenario has two main boundaries:

1. intercepting the network traffic inside the virtual layer without the need to introduce additional external network probes or SDN elements that would convert the east-west traffic into north-south traffic;
2. making the classification of the traffic with minimal consumption of resources that would increase the latency of the packets.

Due to this, we have defined the efficiency of each algorithm as an optimal balance between the classification performance and the time consumed by it. We have built an experimental environment and we have conducted multiple tests, generating various network traffic sets from which we have extracted the network data flows. The most relevant statistical features of the flows were selected as the attributes of the datasets. The source and destination IPs and MAC addresses, as well as the communication ports were not taken into consideration because they are not relevant in a virtualized scenario in which encryption is applied.

The results reveal that Decision Tree and Bayes Net algorithms have the best efficiency, with J48 following just behind them. K-Nearest Neighbor (with $k = 1$) has shown good classification results, but has spent more time for performing the classification. Naive Bayes and AdaBoost have good classification speeds, but have large False Positive classification performance for the VoIP traffic.

The benefits of this analysis are in its practical usage in systems that are highly built on cloud platform, where NFV elements are an integral part of the solution. 5G connectivity to such systems is likely to be used and even 5G own infrastructure is relying on cloud services. Efficient network traffic classification in order to establish the VoIP traffic is a necessity for enabling QoS, security of data, network and application management, monitoring and control.

For future work we are planning to expand our experimental environment to larger virtualized environments with multiple hosts, based on various platforms (Hyper-V, VMWare, XenServer), as well as validating the tests in a real TelCo environment.

References

[1] CHIOSI, M., et al. *Network Functions Virtualisation - Introductory White Paper*. 2012. [Online] Cited 2019-10-10 Available at: https://portal.etsi.org/nfv/nfv_white_paper.pdf

[2] EIMAN, M. *Minimum Technical Performance Requirements for IMT-2020 Radio Interface(s)*. Presentation. 2018. [Online] Cited 2019-10-10. Available at https://www.itu.int/en/ITU-R/study-groups/rsg5/rwp5d/imt-2020/Documents/S01-1_Requirements%20for%20IMT-2020_Rev.pdf

[3] SHANKARA, U. *Communication between Virtual Machines*. US Patent US20070220217A1, Mar. 16, 2007.

[4] VERGARA-REYES, J., MARTINEZ-ORDONEZ, M. C., ORDONEZ, A., et al. IP traffic classification in NFV: A benchmarking of supervised Machine Learning algorithms. In *IEEE Colombian Conference on Communications and Computing*. Cartagena (Colombia), 2017, p. 1–6. DOI: 10.1109/ColComCon.2017.8088199

[5] ALSHAMMARI, R., NUR ZINCIR-HEYWOOD, A. Identification of VoIP encrypted traffic using a machine learning approach. *Journal of King Saud University - Computer and Information Sciences Archive*, 2015, vol. 27, no. 1, p. 77–92. DOI: 10.1016/j.jksuci.2014.03.013

[6] MA, B., ZHANG, H., GUO, Y., et al. A summary of traffic identification method depended on machine learning. In *International Conference on Sensor Networks and Signal Processing (SNSP)*. Xian (China), 2018, p. 469–474. DOI: 10.1109/SNSP.2018.00094

[7] TRIVEDI, U., PATEL, M. A fully automated deep packet inspection verification system with machine learning. In *IEEE International Conference on Advanced Networks and Telecommunications Systems*. Bangalore (India), 2016, p. 1–6. DOI: 10.1109/ANTS.2016.7947802

[8] REZAEI, S., LIU, X. Deep learning for encrypted traffic classification: An overview. *IEEE Communication Magazine*, 2019, vol. 57, no. 5, p. 76–81. DOI: 10.1109/MCOM.2019.1800819

[9] SHAFIQ, M., YU, X., LAGHARI, A. A., et al. Network Traffic Classification techniques and comparative analysis using Machine Learning algorithms. In *The 2nd IEEE International Conference on Computer and Communications* (ICCC). Chengdu (China), 2016, p. 2451–2455. DOI: 10.1109/CompComm.2016.7925139

[10] FRANK, E., HALL, M. A., WITTEN, I. H. *Data Mining: Practical Machine Learning Tools and Techniques*. 4th ed. San Francisco (CA, USA): Morgan Kaufmann, 2016. ISBN: 9780128042915

[11] HUANG, U., LI, P., GUO, S. Traffic scheduling for deep packet inspection in software-defined networks. *Concurrency and Computation: Practice and Experience*. 2017, vol. 29, no. 16 (special issue), p. 1–8. DOI: 10.1002/cpe.3967

[12] MOUSA, M., BAHAA-ELDIN, A., SOBH, M. Software Defined Networking concepts and challenges. In *11th International Conference on Computer Engineering & Systems (ICCES)*. Cairo (Egypt), 2016, p. 79–90. DOI: 10.1109/ICCES.2016.7821979

[13] POLČÁK, L., CALDAROLA, L., CHOUKIR, A., et al. High level policies in SDN. In *International Conference on E-Business and Telecommunications*. 2016, p. 39–57. DOI: 10.1007/978-3-319-30222-5_2

[14] AREVALO HERRERA, J., CAMARGO, J. E. *A Survey on Machine Learning Applications for Software Defined Network Security*. In: Zhou J. et al. (eds) *Applied Cryptography and Network Security Workshops*. *Lecture Notes in Computer Science*, 2019, vol. 11605, Springer, p. 70–93. DOI: 10.1007/978-3-030-29729-9_4

[15] CHOWDHARY, A., HUANG, D., ALSHAMRANI, A., et al. *SDFW: SDN-based Stateful Distributed Firewall*. 2018. DOI: 10.13140/RG.2.2.11001.93281

[16] CHOUDHURY S., BHOWAL, A. Comparative analysis of machine learning algorithms along with classifiers for network intrusion detection. In *International Conference on Smart Technologies and Management for Computing, Communication, Controls, Energy and Materials (ICSTM*). Chennai (Tamil Nadu, India), 2015, p. 89–95. DOI: 10.1109/ICSTM.2015.7225395

[17] SHAFIQ, M. YU, X. LAGHARI, A. A., et al. WeChat text and picture messages service flow traffic classification using machine learning technique. In *IEEE 18th International Conference on High Performance Computing and Communications; IEEE 14th International Conference on Smart City; IEEE 2nd International Conference on Data Science and Systems (HPCC/SmartCity/DSS)*. Sydney (Australia), 2016, p. 58–62. DOI: 10.1109/HPCC-SmartCity-DSS.2016.0019

[18] HE, L., XU, C., LUO, Y. vTC: Machine Learning based traffic classification as a virtual network function. In *ACM International Workshop*, 2016, p. 53–56. DOI: 10.1145/2876019.2876029

[19] LI, Z., GE, Z., MAHIMKAR, A., et al. Predictive analysis in network function virtualization. In *Proceedings of the Internet Measurement Conference*. Boston (USA), 2018, p. 161–167. DOI: 10.1145/3278532.3278547

[20] ZANDER S., ARMITAGE, G. Practical machine learning based multimedia traffic classification for distributed QOS management. In *IEEE 36th Conference on Local Computer Networks* (*IEEE LCN)*. Bonn (Germany), 2011, p. 399–406. DOI: 10.1109/LCN.2011.6115322

[21] MA, W., MEDINA, C., PAN, D. Traffic-aware placement of NFV middleboxes. In *IEEE Global Communications Conference (GLOBECOM)*. San Diego (CA, USA), 2015, p. 1–6. DOI: 10.1109/GLOCOM.2015.7417851

[22] BONFIGLIO, D., MELLIA, M., MEO, M., et al. Revealing Skype traffic: When randomness plays with you. *ACM SIGCOMM Computer Communication Review,* 2007, vol. 37, no. 4, p. 37–48. DOI: 10.1145/1282427.1282386

[23] LE, L., LIN, B., DO, S. Applying big data, machine learning, and SDN/NFV for 5G early-stage traffic classification and network QoS control. *Transactions on Networks and Communications*, 2016, vol. 6, no. 2, p. 36–50. DOI: 10.14738/tnc.62.4446

[24] Oracle VirtualBox. 2019 [Online] Cited 2019-09-10. Available at: https://www.virtualbox.org

[25] BERNAL, M. V., CERRATO, I., RISSO, F., et al. Transparent optimization of inter-virtual network function communication in open vSwitch. In *IEEE International Conference on Cloud Networking (Cloudnet)*. Pisa (Italy), 2016, p. 76–82. DOI: 10.1109/CloudNet.2016.26

[26] Linux Foundatrion, Open vSwitch Project, 2016 [Online] Available at: http://www.openvswitch.org

[27] Wireshark, 2006 [Online] Cited 2019-09-10. Available at: https://www.wireshark.org/

[28] M. Team, 2017 Mininet: An instant virtual network on your laptop (or other pc) - mininet. [Online] Cited 2019-09-12. Available at: http://mininet.org

[29] Ryu Framework, 2019. [Online] Cited 2019-09-10. Available at: http://osrg.github.io/ryu/

[30] BOTTA, A., DAINOTTI, A., PESCAPÈ, A. A tool for the generation of realistic network workload for emerging networking scenarios. *Computer Networks: The International Journal of Computer and Telecommunications Networking,* 2012, vol. 56, no. 15, p. 3531–3547. DOI: 10.1016/j.comnet.2012.02.019

[31] Argus Quosient, 2015. [Online] Cited 2019-09-10. Available at: https://qosient.com/argus/

[32] HALL, M., FRANK, E., HOLMES, G., et al. The WEKA data mining software: An update. In *ACM SIGKDD Explorations Newsletter*, 2009, vol. 11, no. 1, p. 10–18. DOI: 10.1145/1656274.1656278

About the Authors ...

Gjorgji ILIEVSKI was born in Ohrid, Macedonia. He received his M.Sc. in Computer Engineering in the field of Data Mining Technologies in 2012. His research interests include statistical analysis, cloud computing, computer networking, virtualization, LTE and 5G. He works at telecommunication industry in Makedonski Telekom AD as a senior system engineer.

Pero LATKOSKI received his M.S. and Ph.D. degree at the Faculty of Electrical Engineering and Information Technologies, Ss Cyril and Methodius University in Skopje, in 2006 and 2010, respectively. He currently holds the position of a full Professor at the Institute of Telecommunications, at the same university. His research interests include communication protocol engineering, software defined networking, and information theory.