Recognition of traffic generated by WebRTC communication

Nadina Ajdinović[1], Semina Nurkić[1], Jasmina Baraković Husić[1], Sabina Baraković[2]

[1]*University of Sarajevo, Faculty of Electrical Engineering, Zmaja od Bosne bb, Sarajevo, 71000, Bosnia and Herzegovina*

[2]*University of Sarajevo, Faculty of Traffic and Communications, Zmaja od Bosne 8, Sarajevo, 71000, Bosnia and Herzegovina*

Abstract

Network traffic recognition serves as a basic condition for network operators to differentiate and prioritize traffic for a number of purposes, from guaranteeing the Quality of Service (QoS), to monitoring safety, as well as monitoring and detecting anomalies. Web Real-Time Communication (WebRTC) is an open-source project that enables real-time audio, video, and text communication among browsers. Since WebRTC does not include any characteristic pattern for semantically based traffic recognition, this paper proposes models for recognizing traffic generated during WebRTC audio and video communication based on statistical characteristics and usage of machine learning in Weka tool. Five classification algorithms have been used for model development, such as Naive Bayes, J48, Random Forest, REP tree, and Bayes Net. The results show that J48 and BayesNet have the best performances in this experimental case of WebRTC traffic recognition. Future work will be focused on comparison of a wide range of machine learning algorithms using a large enough dataset to improve the significance of the results.

Keywords: *Jitsi media server, machine learning algorithms, Wireshark, WebRTC communication, Weka*

1. Introduction

Web Real-Time Communication (WebRTC) is an open-source project that enables direct real-time communication in web browsers and mobile applications through a simple Application Programming Interface (API) [1]. It contains the basic building blocks for high-quality communication on the web, such as network, audio, and video components used in audio and video applications. WebRTC aims to create a secure media connection between two or more web browsers without the need to install plugins or download native applications. Using existing protocols and applied APIs, WebRTC enables audio and video communication between users via a peer-to-peer connection, supporting all modern web browsers. WebRTC currently supports Chrome, Mozilla Firefox, Safari, Opera, and other Chromium-based browsers [2] [3].

In recent years, real-time network traffic classification has been a major challenge and is an increasingly important area with applications from the Quality of Service (QoS) to security monitoring and anomaly detection. The main goal of the classification is to provide the possibility of automatic recognition of the application that generated a given packet flow by direct or passive observation of individual packets or packet flows flowing through a network [4]. In the past, traffic classification was largely based on well-known protocol ports. WebRTC-based applications use random or non-standard ports, which makes these approaches much less efficient than in the past, as this mode of communication does not include any characteristic pattern for semantically based recognition [5] [6].

Newer techniques classify traffic by recognizing statistical patterns in externally observable traffic attributes, which include the length and arrival time of the packet. The main goal of the statistical method is based on grouping or classifying network traffic flows into groups that have identical statistical properties. The need to classify or group large data sets is one of the reasons for the introduction of Machine Learning (ML) techniques [7]. Statistical methods for accurate and efficient traffic recognition can be divided based on the type of machine learning used, supervised or unsupervised. The aim of this paper is to propose a model for recognizing traffic generated during WebRTC audio and video communication based on statistical characteristics and the use of machine learning.

The rest of the paper is structured as follows. Section 2 provides a brief review of issues, methods, tools, and algorithms for network traffic recognition. Section 3 gives an insight into the methodology used to perform the experimental study. Section 4 provides the results of the study in the form of a model for recognizing traffic

Published: 30 April 2021
Corresponding author: Jasmina Baraković Husić *(jbarakovic@etf.unsa.ba)*

generated during WebRTC audio and video communication. Section 5 concludes the paper and proposes the direction for future work.

2. Related work

This section provides an insight into related work that suggests models for network traffic recognition. Table 1 presents a non-exhaustive review of related work, which considers types of traffic, methods, tools, and algorithms for network traffic recognition. Also, Table 1 shows the main conclusions of the considered works. We have chosen 12 papers according to their relevance to a given topic.

In recent literature, several ways have been introduced to recognize the traffic generated during WebRTC audio and video communication based on statistical characteristics and the usage of machine learning. A total of two papers have proposed models based on a decision theory that enable recognition of encrypted WebRTC traffic using machine learning techniques, using the Weka tool. In addition, a comparison of the most important classification algorithms, such as J48, Simple Cart, Naive Bayes, and Random Forest, has been presented in [5] and [8]. The evaluation shows that the J48, Simple Cart, and Random Forest algorithms achieve better and more comparable performance than the Naive Bayes algorithm. Also, the experiment suggests that the J48 offers best results in terms of False Positive Rate (FPR), whereas Random Forest performs better in terms of True Positive Rate (TPR) detection.

Bayesian analyses, implemented in the Weka environment, are discussed in [9], [10], and [11]. Compared to other network traffic classification algorithms, the obtained results show the efficiency of the Naive Bayes algorithm in terms of accuracy.

Models for recognizing Skype traffic based on statistical characteristics and the usage of machine learning have been proposed in the [12] and [13]. An assessment of classification algorithms, such as J48, Simple Cart, and Naive Bayes, has been proposed in [12]. The comparison of algorithms shows that the J48 and Simple Cart achieve the best results. The authors of reference [13] propose appropriate machine learning tools for recognizing Skype traffic, implement a system that separates Voice over Internet Protocol (VoIP) calls made to Skype, and define functions to eliminate repetitive and redundant information all by providing a way to filter out records based on Internet Protocol (IP) address.

A framework based on two complementary techniques to reveal Skype traffic in real time is presented in [14]. The first approach is based on statistical recognition of Skype traffic using Pearson's Chi-Square test. Contrariwise, the second approach is based on a stochastic recognition of Skype traffic in terms of packet arrival speed and packet length, which are used as characteristics of a decision process based on Naive Bayes algorithm. Experimental results obtained from measurements in different networks show that the combination of the above techniques is very effective in identifying Skype traffic.

The authors of reference [15] present a comparison of the performance of machine learning algorithms for network traffic recognition. In [15], a performance assessment was performed for five IP traffic classification algorithms, such as: Naive Bayes with Discretization (NBD), Naive Bayes Kernel Estimation (NBKE), J48, BayesNet, and Naive Bayes Tree (NBTree). Comparing the classification speed, the J48 algorithm was able to identify network flows faster than the remaining algorithms. Also, the experimental results show that the NBK algorithm has the slowest classification speed, followed by the algorithms: NBTree, Bayes Net, NBD, and J48. Time taken to build model shows that NBTree is the slowest by a considerable margin. The rest of algorithms were more uniform, where a classier is built the fastest by NBK, followed by NBD, Bayes Net, and J.48.

A method for recognizing peer-to-peer network traffic between BitTorrent, PPLive, Skype, and MSN Messenger, based on the Support Vector Machine (SVM) algorithm, has been proposed in [16]. The experiment shows that this method can carry on effective classification for peer-to-peer flows, even for protocol encryption of application layer and some network flows which are difficult to be classified.

Previous related works rely on the classification of network traffic using statistical characteristics expressed in full-flow. The authors of references [17] and [18] propose a novel approach to train the machine learning classifier using statistical features calculated over multiple short sub-flows extracted from full-flow generated by the target application, resulting in excellent performance.

Most related work from Table 1 is based on statistical methods (91.67%) and machine learning algorithms using Weka tool (75%). Also, a number of classification algorithms were tested in related works, among which the most common are: Naive Bayes (75%), J48 (50%), and Simple Cart (25%). The overall analysis, which considers methods, tools, and algorithms for the classification of network traffic served to define the research methodology to be used in this paper.

As stated in Table 1, only two references (16.67%) study the classification of traffic generated during WebRTC communication, which compare only four classification algorithms. Therefore, the aim is to examine a number of algorithms for classifying WebRTC traffic and propose a model for recognizing such traffic based on statistical characteristics and usage of machine learning in Weka tool.

Table 1. Related work on traffic classification

Ref.	Year	Type of traffic	Method	Tool	Algorithm	Conclusion
[5]	2015	WebRTC	Statistical	Weka	J48, Simple Cart, Naive Bayes, Random Forest	The Random Forest algorithm offers the best results in terms of TPR, whereas J48 performs better in terms of FPR detection.
[8]	2015	WebRTC	Statistical	Weka	J48, Simple Cart, Naive Bayes	The J48 and Simple Cart algorithms achieve better and comparable performances than the Naive Bayes algorithm.
[9]	2005	Web	Statistical	Weka	Naive Bayes	The Naive Bayes algorithm is able to provide 65% accuracy for data from the same period and can achieve over 95% accuracy when combined with a number of simple refinements.
[10]	2012	VoIP	Statistical	Weka	Naive Bayes	Comparisons of classifiers in terms of accuracy and computational time show the efficiency of Bayesian classifiers.
[11]	2005	Web	Statistical	-	Naive Bayes	Classification of traffic using the Naive Bayes algorithm is capable of 67% accuracy per-flow or better than 83% accuracy both per-byte and per-packet.
[12]	2014	Skype	Statistical	Weka	J48, Simple Cart, Naive Bayes	Beside Skype traffic detection, the algorithms are scalable and flexible enough to be applicable to the detection of other types of network traffic.
[13]	2009	Skype	Statistical	NetAl NetMate Weka	J48, Naive Bayes	In terms of accuracy, the J48 algorithm has better results than the Naive Bayes algorithm for classifying Skype traffic.
[14]	2007	Skype	Statistical/ stochastic	-	Naive Bayes	Although the Bayesian classifiers follow traditional design, the Chi-Square test can be successfully extended to the more general traffic classification problem.
[15]	2006	IP	Statistical	Weka	NBTree, J48, BayesNet, NBD, NBKE	Comparing the classification speed, the J48 algorithm is able to identify network flows faster than the remaining algorithms, while the NBK algorithm has the slowest classification speed followed by NBTree, Bayes Net, NBD and J48. The highest accuracy was achieved by the J48 algorithm.
[16]	2008	Web	Statistical	-	SVM	A method based on the SVM algorithm to realize the P2P network traffic classification can carry on effective classification for P2P flow, even for protocol encryption of application layer and some network flows which are difficult to classified
[17]	2006	IP	Statistical	Weka	Naive Bayes	Using the Naive Bayes algorithm, this approach showed results in excellent performance even when classification is initiated mid-way through a flow.
[18]	2006	IP	Statistical	Weka	Naive Bayes, J48	The decision of J48 algorithm is based on a particularly possible value of the attribute, while the decision of Naive Bayes algorithm is based on the distribution of the attribute values that might match better with possible attribute values of the interference traffic.

Legend: *BayesNet – Bayesian Network; FPR – False Positive Rate; IP – Internet Protocol; NBD – Naive Bayes with Discretization; NBKE – Naive Bayes Kernel Estimation; NBTree – Naive Bayes Tree; REPTree – Reduced Error Pruning Tree; SVM – Support Vector Machine; TPR – True Positive Rate; VoIP – Voice over Internet Protocol; WebRTC – Web Real-Time Communication.*

3. Experiment design and procedure

A. Experiment Design

In order to achieve the aforementioned aim, an experiment environment was setup and configured on HP laptop with Ubuntu virtual machines. One virtual machine was used to install Jitsi Meet [19], as WebRTC open-source media server, which achieved the best performance for relatively small number of participants [20], and another one was used to install Ostinato [21], as a tool to generate additional network traffic. This laptop was connected to wireless router Innbox F60 FTTH which provided the access to a Wireless Fidelity (Wi-Fi) network. To conduct the WebRTC audio and video call over Wi-Fi networks and collect whole generated traffic, two laptops were used with installed Google Chrome browser version 85.0.4183.121.

Two users, who were in different rooms, participated in this experiment. A free conversation task was performed between participants knowing each other and being located in different rooms as recommended in ITU-T P.805 [22].

B. Experiment Procedure

Both participants, after connecting to the server, had to perform eight steps, i.e., (i) launch Google Chrome browser, (ii) enter the domain jitsitest.mms.com, (iii) enter the name of pre-arranged common room and start audio and video call over Jitsi Meet, (iv) start the network traffic generator Ostinato, (v) start recording network traffic via Wireshark, (vi) participate in an audio and video call lasting 3 minutes, (vii) after the expiration of the defined time, and before the communication is interrupted, stop recording and save recorded network traffic via Wireshark [23], (viii) stop audio and video call and close the web browser.

The recorded network traffic was processed using the Weka tool [24], in which 10-fold cross-validation was used for evaluation. From recorded traffic, features such as *time*, *source address*, *source port*, *destination address*, *destination port*, *protocol, length,* and *ID* can be extracted. The resulting .csv file contains 720 samples and the 8 previously mentioned attributes on the basis of which the packages were classified on *WebRTC* (89 samples) and *Normal* (631 samples). Different sizes of these classes result in probabilities p_{Normal} = 0.8764 and p_{WebRTC} = 0.1236, that are the values required by different algorithms for attribute selection and decision making.

4. Results and discussion

Since the aim of this paper is to propose a model for recognizing traffic generated by WebRTC communication, models have been created using classification algorithms based on machine learning. The following algorithms have been used: Naive Bayes, J48, Random Forest, REPTree, and BayesNet.

Table 2 shows a comparison of these algorithms based on the time taken to build the model and its accuracy. The J48 algorithm has the largest number of correctly classified instances (accuracy: 93.8889%), and the Naive Bayes algorithm has the largest number of incorrectly classified instances (accuracy: 79.8611%). The J48 algorithm requires the least time to build the model (0s), and the Random Forest algorithm requires the most time (0.19s).

A comparative analysis of classification algorithms on the same dataset is presented in Table 3 in terms of six quality metrics: (i) True Positive Rate (TPR) indicating the number of correctly classified positive samples in relation to the total number of positive samples, (ii) False Positive Rate (FPR) indicating the number of incorrectly classified negative samples in relation to the total number of negative samples, (iii) Precision indicating the number of correctly classified positive samples in relation to the total number of samples classified as positive, (iv) Recall, (v) F-Measure, a measure of test’s accuracy, (vi) Receiver Operating Characteristic (ROC) Area. The J48 algorithm offers the best results in terms of True Positive Rate detection, whereas BayesNet performs better in terms of False Positive Rate detection. The Naive Bayes algorithm has the lowest percentage in terms of True Positive Rate detection, and the REPTree algorithm has the lowest percentage in terms of False Positive Rate detection. The best precision has the BayesNet algorithm, and the REPTree algorithm has a low percentage of precision. The recall metric is often presented as a TPR metric, so the results, shown in Table 3, are the same for these two metrics. The F1-measure, as a combination of precision and response metrics, has the highest percentage for the J48 algorithm and the lowest for the REPTree algorithm.

Table 2. Time taken to build model and accuracy of classification algorithms

	Naive Bayes	**J48**	**Random Forest**	**REPTree**	**Bayes Net**
Time taken to build model [s]	0.02	0	0.19	0.03	0.03
Number of Correctly Classified Instances	575	676	634	630	666
Number of Incorrectly Classified Instances	145	44	86	90	54
Total Number of Instances	720	720	720	720	720

Legend: *BayesNet – Bayesian Network; REPTree – Reduced Error Pruning Tree.*

Table 3. Comparison of quality metrics of different classification algorithms

	Naive Bayes	J48	Random Forest	REPTree	Bayes Net
TPR	0.799	0.939	0.881	0.875	0.925
FPR	0.337	0.356	0.828	0.867	0.040
Precision	0.867	0.936	0.861	0.810	0.949
Recall	0.799	0.939	0.881	0.875	0.925
F-Measure	0.823	0.934	0.833	0.821	0.931
ROC Area	0.821	0.852	0.977	0.515	0.990

Legend: *BayesNet – Bayesian Network; FPR – False Positive Rate; REPTree – Reduced Error Pruning Tree; ROC – Receiver Operating Characteristic; TPR – True Positive Rate.*

Figure 1 shows the comparison of all metrics for the five considered classification algorithms. Comparing the quality metrics of different classification algorithms, the best results were obtained by the J48 and BayesNet algorithms with an accuracy of 93.8889% and 92.5%, respectively. Time taken to build model for the J48 algorithm is 0s, and for the NaiveBayes algorithm is 0.03s.

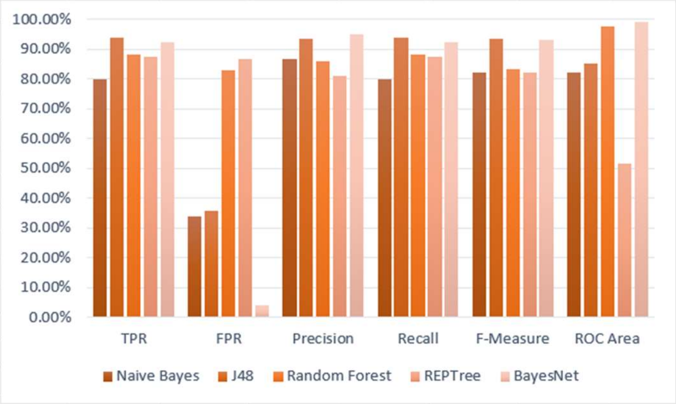

Figure 1. Algorithms comparison

Finally, a model for recognizing traffic generated during WebRTC communication can be implemented using two algorithms, J48, and BayesNet. The existing models for traffic recognition, presented in Table 1, have a comparable or lower rate of classification accuracy compared to the models proposed in this paper. In [5] and [8], four models for WebRTC traffic recognition were proposed, based on classification algorithms, such as: J48, Simple Cart, Naive Bayes, and Random Forest. The four created models were analyzed based on the quality metrics used in this paper as well. Evaluation of quality metrics has shown that the J48, Simple Cart and Random Forest algorithms perform better than the Naive Bayes algorithm. Therefore, as in this paper, the J48 algorithm has better performance with an accuracy of 95.0652%, and the Naive Bayes algorithm has worse performance with an accuracy of 85.9404%.

5. Conclusion

Applications based on WebRTC technology, which provides real-time audio and video communication via a web browser, represent a significant innovation in web telephony. Communication based on WebRTC technology is difficult to detect because it can use dynamic port allocation and does not include any characteristic pattern that allows a semantic-based recognition. The focus of this paper was on statistically based methods for recognizing traffic generated during WebRTC communication. Based on the related work, the tool and machine learning algorithms were selected by which the model for recognizing WebRTC traffic has been created.

Therefore, the main contribution of this paper are the new models for recognizing traffic generated by WebRTC communication, based on classification algorithms such as J48 and BayesNet. These proposed models provide the ability to recognize WebRTC traffic with greater accuracy than previously proposed models.

The results in this paper are a good starting point for future research activities, which will include a comparison of a wide range of machine learning algorithms using a large enough dataset to improve the relevance of the results. Furthermore, future work will include consideration of additional features used for classification purposes, such as *flags*, *headerChecksum*, *timeToLive*.

References

[1] Real-time communication with WebRTC. 2021. [Online]. Available: https://codelabs.developers.google.com/codelabs/webrtc-web#0 .

[2] Who supports WebRTC?. 2021. [Online]. Available: https://www.3cx.com/voip/which-browsers-support-webrtc/

[3] C. Vogt, M. J. Werner and T. C. Schmidt, "Leveraging webrtc for p2p content distribution in web browsers," *2013 21st IEEE International Conference on Network Protocols (ICNP)*, 2013.

[4] S. Valenti, D. Rossi, A. Dainotti, A. Pescape, A. Finamore and M. Mellia, "Reviewing Traffic Classification," *Traffic Monitoring and Analysis*, vol. 7754, 2013.

[5] M. Di Mauro and M. Longo, "Revealing Encrypted WebRTC Traffic via Machine Learning Tools," in *12th International Joint Conference on e-Business and Telecommunications (ICETE)*, 2015, pp. 259–266.

[6] R. N. Jesudasan, P. Branch and J. But, “Generic Attributes for Skype Identification Using Machine Learning,” *Technical Report 100820A*, 2010.

[7] R. C. Jaiswal and S. D. Lokhande, “Machine learning based internet traffic recognition with statistical approach,” *2013 Annual IEEE India Conference (INDICON)*, 2013, pp. 1-6.

[8] M. Di Mauro and M. Longo, “A Decision Theory Based Tool for Detection of Encrypted WebRTC Traffic,” in *18th International Conference on Intelligence in Next Generation Networks*, 2015, pp. 89–94.

[9] A. W. Moore and D. Zuev, “Internet traffic classification using Bayesian analysis Techniques,” in *Proceedings of the 2005 ACM SIGMETRICS international conference on Measurement and modeling of computer systems (SIGMETRICS ’05). Association for Computing Machinery*, 2005, pp. 50–60.

[10] F. Rahdari and M. Eftekhari, “Using Bayesian classifiers for estimating quality of VoIP,” *The 16th CSI International Symposium on Artificial Intelligence and Signal Processing (AISP 2012)*, 2012, pp. 348–353.

[11] D. Zuev and A. W. Moore, “Traffic Classification Using a Statistical Approach,” in *Proceedings of the 6th international conference on Passive and Active Network Measurement (PAM’05)*, 2005, pp. 321–324.

[12] M. Di Mauro and M. Longo, “Skype traffic detection: a decision theory based tool,” *2014 International Carnahan Conference on Security Technology (ICCST)*, 2014, pp. 1–6.

[13] A. O. Calchland, Van T. Dinh, P. Branch and J. But, “Skype Traffic Detector,” *Technical Report 090128A*, 2009.

[14] D. Bonfiglio, M. Mellia, M. Meo, D. Rossi and P. Tofanelli, “Revealing skype traffic: When randomness plays with you,” in *Proceedings of the 2007 conference on Applications, technologies, architectures, and protocols for computer communications,* 2007, pp. 37–48.

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

[16] Z. Xusheng, “A P2P Traffic Classification Method Based on SVM,” *2008 International Symposium on Computer Science and Computational Technology*, 2008, pp. 53–57.

[17] T.T.T. Nguyen and G. Armitage, “Training on multiple sub-flows to optimise the use of Machine Learning classifiers in real-world IP networks,” in *Proceedings 2006 31st IEEE Conference on Local Computer Networks*, 2006, pp. 369–376

[18] T.T.T. Nguyen and G. Armitage, “Synthetic sub-flow pairs for timely and stable IP traffic identification,” in *Proceedings Australian Telecommunication Networks and Application Conference*, 2006.

[19] Jitsi Meet. 2021. [Online]. Available: https://jitsi.org/jitsi-meet/

[20] E. Andre, N. Le Breton, A. Lemesle, L. Roux and A. Gouaillard, “Comparative study of WebRTC SFUs for Video Conferencing,” in *Proceedings of the Principles, Systems and Applications of IP Telecommunications (IPTComm)*, 2018, pp. 1–8.

[21] Ostinato. 2021. [Online]. Available: https://ostinato.org/ .

[22] Subjective evaluation of conversational quality, ITU-T P.805, 2007.

[23] Wireshark. 2021. [Online]. Available: https://www.wireshark.org/ .

[24] Weka. 2021. [Online]. Available: https://www.cs.waikato.ac.nz/ml/weka/ .