A Look Behind the Curtain: Traffic Classification in an Increasingly Encrypted Web

Iman Akbari[1], Mohammad A. Salahuddin[1], Leni Ven[1], Noura Limam[1], Raouf Boutaba[1]
Bertrand Mathieu[2], Stephanie Moteau[2], and Stephane Tuffin[2]
{iakbaria,mohammad.salahuddin,shwen,noura.limam,rboutaba}@uwaterloo.ca
{bertrand2.mathieu,stephanie.moteau,stephane.tuffin}@orange.com
[1]University of Waterloo, Waterloo, Ontario, Canada, [2]Orange Labs, Lannion, France

ABSTRACT

Traffic classification is essential in network management for operations ranging from capacity planning, performance monitoring, volumetry, and resource provisioning, to anomaly detection and security. Recently, it has become increasingly challenging with the widespread adoption of encryption in the Internet, e.g., as a de-facto in HTTP/2 and QUIC protocols. In the current state of encrypted traffic classification using Deep Learning (DL), we identify fundamental issues in the way it is typically approached. For instance, although complex DL models with millions of parameters are being used, these models implement a relatively simple logic based on certain header fields of the TLS handshake, limiting model robustness to future versions of encrypted protocols. Furthermore, encrypted traffic is often treated as any other raw input for DL, while crucial domain-specific considerations are commonly ignored. In this paper, we design a novel feature engineering approach that generalizes well for encrypted web protocols, and develop a neural network architecture based on Stacked Long Short-Term Memory (LSTM) layers and Convolutional Neural Networks (CNN). We evaluate our approach on a real-world web traffic dataset from a major Internet service provider and Mobile Network Operator. We achieve an accuracy of 95% in service classification with less raw traffic and smaller number of parameters, out-performing a state-of-the-art method by nearly 50% fewer false classifications. We show that our DL model generalizes for different classification objectives and encrypted web protocols. We also evaluate our approach on a public QUIC dataset with finer application-level granularity in labeling, achieving an overall accuracy of 99%.

KEYWORDS

Encrypted traffic classification; HTTP/2; QUIC; TLS; deep learning

ACM Reference Format:
Iman Akbari[1], Mohammad A. Salahuddin[1], Leni Ven[1], Noura Limam[1], Raouf Boutaba[1], Bertrand Mathieu[2], Stephanie Moteau[2], and Stephane Tuffin[2]. 2021. A Look Behind the Curtain: Traffic Classification in an Increasingly Encrypted Web. In *Abstract Proceedings of the 2021 ACM SIGMETRICS / International Conference on Measurement and Modeling of Computer Systems (SIGMETRICS '21 Abstracts), June 14–18, 2021, Virtual Event, China.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3410220.3453921

SIGMETRICS '21 Abstracts, June 14–18, 2021, Virtual Event, China

ACM ISBN 978-1-4503-8072-0/21/06.
https://doi.org/10.1145/3410220.3453921

1 INTRODUCTION

Traffic classification is quintessential for network operators to perform a wide range of network operation and management activities. This includes capacity planning, security and intrusion detection, quality of service (QoS), performance monitoring, volumetry, and resource provisioning, to name a few. For example, an enterprise network administrator or Internet service provider (ISP) may want to prioritize traffic for business critical services, identify unknown traffic for anomaly detection, or perform workload characterization for designing efficient resource management schemes to satisfy performance and resource requirements of diverse applications. Depending on the context, misclassification on a large scale may result in failure to deliver QoS guarantees, incur operational expenses, security breaches or even disruption in services.

Today, encrypted communication between clients and servers has become the norm. Most prominent web-based services are now running over HTTPS. On the other hand, to improve security and quality of experience (QoE) for end-users, new web protocols (e.g., HTTP/2 and QUIC) have emerged, which overcome various limitations of HTTP/1.1. We estimate that around 32% of all HTTPS sessions already use HTTP/2 as their underlying protocol. However, HTTP/2 features, such as payload encryption, multiplexing and concurrency, resource prioritization, and server push, add to the complexity of traffic classification. While a large body of literature harnesses the power of Machine Learning (ML) for different traffic classification objectives (e.g., service- and application-level, QoE prediction, security, etc.), there exist various limitations that must be addressed for its real-world, practical usage.

For instance, the particular nature of encrypted traffic is not taken into account in many state-of-the-art approaches, which affects their performance and efficiency when applied to encrypted protocols. Due to a lack of standard framework for traffic classification, numerous works in traffic classification (e.g., [2, 7]) pick their labels somehow arbitrarily, which are often inconsistent in granularity. Furthermore, many approaches (e.g., [3, 4, 6, 8]) use datasets with a mixed set of protocols that are often easily distinguishable using header signatures, making it unrealistic to justify the use of computationally expensive ML models. In some cases (e.g., [1]), traffic classification approaches rely on clever techniques to guide the models based on expert domain-specific knowledge that can be jeopardized by small variations in the protocol. Another important issue is that some protocol extensions, such as the Server

	Weighted Average Precision (%)	Weighted Average Recall (%)	Weighted Average F1-score (%)	Accuracy (%)	Epoch Time (s)
Full Model (SLSTM)	95.62	95.56	95.57	95.56	2584
Full Model (CNN)	94.54	94.42	94.37	94.43	232
Flow-only Model (SLSTM)	86.71	86.51	86.56	86.51	1814
Flow-only Model (CNN)	76.77	73.17	73.76	73.17	211
UCDavis CNN [4]	91.09	91.06	91.04	91.05	168
UCDavis CNN-LSTM [4]	89.74	89.72	89.73	89.72	245
Traditional Baseline (C4.5)	81.56	81.39	81.41	81.39	18*

Table 1: Performance comparison of TLS flow classification models (*C4.5 time is reported for entire training)

Name Indication (SNI) in Transport Layer Security (TLS), can essentially reveal the server's identity, allowing for trivial classification of many traffic flows based on the server name. In this case, it can be argued that expensive and complex models are being used to learn a relatively simple logic, similar to that of a server name to label look-up table, which can be implemented deterministically.

These issues call for a more comprehensive study of how deep traffic classification models behave on encrypted traffic, especially popular emerging web protocols due to their ubiquity. They also underline the importance of developing general frameworks and guidelines for how encrypted web traffic should be treated as a data type for future research in traffic classification.

2 CONTRIBUTIONS

In this paper, we leverage Deep Learning (DL) for service classification (e.g., video streaming, social media, web mail) with a focus on new encrypted web protocols, i.e., HTTP/2 and QUIC, and overcome the above limitations. Unlike many works in this area, we focus exclusively on encrypted web traffic, and explore the challenges of unleashing the full potential of DL to find complex patterns that are innate to each traffic class. We occlude parts of the input that the DL model can use to learn a lazy and unsophisticated logic, and instigate how encrypted traffic should be treated differently from general raw ML input, e.g., images. We also place emphasis on a feature set that generalizes the applicability of the model for varied encrypted web traffic classification objectives.

We propose a novel feature engineering approach for encrypted traffic classification that focuses on protocol-agnostic aspects of the encrypted web traffic. In our approach, we make use of standard flow statistics, the traffic shape with respect to packet sizes, inter-arrival times, and direction, along with raw bytes from the TLS handshake packets. This is in contrast to most DL approaches for traffic classification, where the full raw traffic is fed to the DL model. We justify the proposed feature set to be a better fit for the classification of encrypted traffic. We also develop a neural network architecture based on Convolutional Neural Network (CNN) and Stacked Long Short-Term Memory (LSTM) layers that is highly effective in leveraging the extracted features for distinguishing between different traffic classes. Our DL model identifies and correlates useful traffic traits, while being lighter in the number of trainable parameters and less likely to overfit, compared to the existing methods.

We use a real-world mobile traffic dataset from an ISP, and demonstrate that our approach has an edge over the state-of-the-art in service classification over encrypted web traffic. Table 1 compares the performance of our model with [4] and a traditional baseline. Using our model based on Stacked LSTM layers, we achieve an accuracy of over 95% for classification exclusively over HTTPS (i.e., HTTP/1.1 and HTTP/2 over TLS), outperforming [4] by a significant margin of nearly 50% fewer false classifications. It is also shown that our approach generally achieves higher accuracies as it is less prone to over-fitting. Furthermore, the variation of our model that uses CNN layers instead of Stacked LSTM, requires lower training time while still achieving a higher accuracy compared to the state-of-the-art. We have made the corresponding pre-processed dataset available to the public.[1]

We also showcase that our DL model generalizes for a finer classification granularity, i.e., application-level classification. Furthermore, we show that our model adapts to a different encrypted web protocol, i.e., QUIC, by simply changing the training data. We achieve an accuracy of 97% in application-level classification and an accuracy of 99% on a public QUIC dataset [5].

ACKNOWLEDGMENTS

We thank our shepherd Athina Markopoulou and the anonymous reviewers for their valuable feedback, Yann Meyer for his help in the preparation of the dataset, and Ezzeldin Tahoun for his help in the preliminary stages of the project.

REFERENCES

[1] Pierre-Olivier Brissaud, Jérôme Francçis, Isabelle Chrisment, Thibault Cholez, and Olivier Bettan. 2019. Transparent and Service-Agnostic Monitoring of Encrypted Web Traffic. *IEEE Transactions on Network and Service Management* 16, 3 (2019), 842–856.

[2] Manuel Lopez-Martin, Belen Carro, Antonio Sanchez-Esguevillas, and Jaime Lloret. 2017. Network traffic classifier with convolutional and recurrent neural networks for Internet of Things. *IEEE Access* 5 (2017), 18042–18050.

[3] Mohammad Lotfollahi, Mahdi Jafari Siavoshani, Ramin Shirali Hossein Zade, and Mohammdsadegh Saberian. 2020. Deep packet: A novel approach for encrypted traffic classification using deep learning. *Springer Soft Computing* 24, 3 (2020), 1999–2012.

[4] Shahbaz Rezaei, Bryce Kroencke, and Xin Liu. 2019. Large-scale mobile app identification using deep learning. *IEEE Access* 8 (2019), 348–362.

[5] Shahbaz Rezaei and Xin Liu. 2018. How to achieve high classification accuracy with just a few labels: semi-supervised approach using sampled packets. *arXiv preprint arXiv:1812.09761* (2018).

[6] Wei Wang, Ming Zhu, Jinlin Wang, Xuewen Zeng, and Zhongzhen Yang. 2017. End-to-end encrypted traffic classification with one-dimensional convolution neural networks. In *IEEE International Conference on Intelligence and Security Informatics*. 43–48.

[7] Haipeng Yao, Pengcheng Gao, Jingjing Wang, Peiying Zhang, Chunxiao Jiang, and Zhu Han. 2019. Capsule network assisted IoT traffic classification mechanism for smart cities. *IEEE Internet of Things Journal* 6, 5 (2019), 7515–7525.

[8] Zhuang Zou, Jingguo Ge, Hongbo Zheng, Yulei Wu, Chunjing Han, and Zhongjiang Yao. 2018. Encrypted traffic classification with a convolutional long short-term memory neural network. In *IEEE International Conference on High Performance Computing and Communications; IEEE International Conference on Smart City; IEEE International Conference on Data Science and Systems*. 329–334.

[1] The dataset is available for download at http://bit.ly/UW-Orange-2020