Generalized Classification of DNS over HTTPS Traffic with Deep Learning

Lionel F. Gonzalez Casanova
Department of Electrical Engineering
Yuan Ze University
Taoyuan, Taiwan, R.O.C.
Email: s1018504@mail.yzu.edu.tw

Po-Chiang Lin
Department of Electrical Engineering
Yuan Ze University
Taoyuan, Taiwan, R.O.C.
Email: pclin@saturn.yzu.edu.tw

***Abstract*—Network anomaly detection has been a challenge for both industry and academia. The alarming situation of network attacks is a worrisome problem for many Internet services. Machine learning techniques are widely investigated to detect suspicious events from network traffic flow. In this paper, we investigate the DNS over HTTPS traffic classification. The majority of related works use a variety of features from datasets. However, some of the adopted features are specific to some networking environments, and those features make the trained models not generalized to other network environments. The generalization of a machine learning model is of critical importance, since it would affect the effectiveness when the model is applied to other network environments. We design an appropriate data processing pipeline to process the CIRA-CIC-DoHBrw-2020 time series dataset, including feature selection and data imbalance handling, in order to facilitate the generalization of deep learning models. We develop truly generalized deep learning models, including the LSTM model and the BiLSTM model, to classify DoH traffic with high accuracy and low latency. While both models achieve good performance, the BiLSTM model performs better than the LSTM model does for both the accuracy and the computation time.**

***Index Terms*—Deep learning, network attack, anomaly detection, machine learning, neural network, DNS over Hypertext Transfer Protocol Secure, DoH.**

I. Introduction

The Internet has become intrinsic in our daily lives, so information security on the Internet is of paramount importance [1]. Among all Internet protocols, the Domain Name System (DNS) is one of the most important and widely-used protocols. The DNS is a critical subsystem of the Internet infrastructure, on which most Internet-applications depend [2]. The DNS is a hierarchical and decentralized system used to name computers, services, or other resources on the Internet. Human and machines rely on the DNS requests and responses to find their communication targets over the Internet. Just in the first quarter of 2019, more than five trillion DNS messages were exchanged among users per month [3]. However, the DNS is one of the vulnerable network protocols that have been exploited by network attackers repeatedly over the years. Providing secure DNS requests and responses is a challenging task.

This work was supported in part by Ministry of Science and Technology (MOST), Taipei, Taiwan, R.O.C. under grant number MOST 108-2221-E-155-023-MY2 and MOST 110-2221-E-155-002.

The DNS over Hypertext Transfer Protocol Secure (HTTPS), abbreviated as DoH, is already defined in the Internet Engineering Task Force (IETF) Request for Comments (RFC) 8484. The purpose of DoH is to send DNS queries and getting DNS responses over HTTPS. General web browsers, such as Google Chrome and Mozilla Firefox, already support DoH. However, there exist several DNS tunneling tools, such as dns2tcp, DNSCat2, and Iodine, that can be used by network attackers to generate malicious DoH traffic. The detection of malicious DoH traffic is thus important.

In this paper we investigate the traffic classification of DoH by using machine learning and the CIRA-CIC-DoHBrw-2020 dataset [4]. To date, Machine learning is taking many businesses and industries by storm. Huge amount of data is produced and processed to train machine learning models. According to [5], over the past decade there has been an increased interest in time series classification. Time series data is everywhere existing in many areas of research. Look no further than the CIRA-CIC-DoHBrw-2020 dataset which is a time series dataset. Moreover, the CIRA-CIC-DoHBrw-2020 dataset is the first of its kind and it needs to be processed meticulously in order to get good results from any machine learning model. Hence, we have studied the dataset used in this research whereby we want to contribute significantly to the area of network security by using machine learning techniques. Deep learning has been successfully implemented in various applications that require time series data. The majority of related works use a variety of features from datasets. For example, Banadaki and Robert uses 34 features in their work, including the SourceIP, DestinationIP, SourcePort, and DestinationPort features [6]. However, some of the adopted features are specific to some networking environments, and those features make the trained models not generalized to other network environments. We argue that the SourceIP is apparent and can be easily changed by network attackers. In other words, using the SourceIP feature to guarantee network security does not represent a real world scenario. Moreover, the timestamp feature just means the time stamp when the data instance is collected. The timestamp feature would be useless for future inference. To summarize, the generalization of a machine learning model is of critical importance, since it would affect the effectiveness when a model is applied

to other network environments. In this paper, we show that the CIRA-CIC-DoHBrw-2020 dataset could be used to train machine learning models by using fewer features compared to the previous papers in the literature. We drop several specific features to imitate a real intrusion detection scenario; that is, the fewer features we use to train the machine learning models, the more realistic it can be.

Another crucial aspect of data processing for machine learning is the data imbalance problem. In a binary classification problem with data samples from two groups, class imbalance occurs when one class, the minority group, contains significantly fewer samples than the other class, the majority class. It is important to stress that in many problems, the minority group is the class of interest. Johnson and Khoshgoftaar stated that highly imbalanced data poses added difficulty, as most learners will exhibit bias towards the majority class, and in extreme cases, may ignore the minority class altogether [7]. In this paper, we process the data to make it balanced in order to improve the performance. We use the processed dataset to train and evaluate two deep learning models, including the Long Short-Term Memory (LSTM) and the Bi-directional Long Short-Term Memory (BiLSTM) model.

The major contributions of this paper are twofold:

1) We design an appropriate data processing pipeline to process the CIRA-CIC-DoHBrw-2020 dataset, including feature selection and data imbalance handling, in order to facilitate the generalization of deep learning models.
2) We develop truly generalized deep learning models to classify DoH traffic with high accuracy and low latency.

The rest of this paper is organized as follows. In Section II we describe the related work in the literature. The dataset description and the data processing pipeline are provided in Section III. The deep learning-based classification models for DoH traffic are described in Section IV, followed by the numerical results and discussions in Section V. Finally, conclusions are presented in Section VI.

II. Related Work

Banadaki and Robert studied a systematic two-layer approach for detecting DNS over HTTPs (DoH) traffic and distinguishing Benign-DoH traffic from Malicious-DoH traffic using a number of machine learning algorithms [6]. The author evaluated the CIRA-CIC-DoHBrw-2020 dataset using the LGBM and XGBoost algorithms considering their accuracy, precision, recall, F-score, confusion matrices, ROC curves, and feature importance. These two algorithms outperformed the other four machine learning algorithms. LGBM and XGBoost algorithms show the maximum accuracy of 100% in the classification tasks of layers 1 and 2. The author explains that LGBM algorithms misclassified one DoH traffic test as non-DoH out of 4000 test datasets. In addition, out of 34 features extracted from the CIRA-CIC-DoHBrw-2020 dataset, Source IP is the critical feature for classifying DoH traffic from non-DoH traffic in layer one followed by the DestinationIP feature. Interestingly, the feature DestinationIP is an important feature for LGBM and gradient boosting when classifying Benign-DoH from Malicious-DoH traffic in layer 2. Singh and Roy also use machine learning techniques to detect malicious DoH traffic and got similar results [8].

MontazeriShatoori et al. states that Domain Name System (DNS) is the internet backbone for providing a mapping between human-readable hostnames and computer understandable Internet Protocol (IP) addresses [9]. The DNS protocol follows a decentralized hierarchical approach. The mechanism is when a DNS client generates a DNS query requesting for an IP address, the local DNS server responds after looking into its cache. Technically, it does not find the answer within cache memory, it forwards the DNS query to recursive DNS resolver which tracks down the DNS record with repetitive DNS queries to root name server, Top Level Domain (TLD) name server and authoritative name servers until it gets the target authoritative answer. They did research on identifying the tunneling activities that utilize DNS communications over HTTPs by presenting a two-layered approach to detect and characterize DoH traffic using time-series classifiers. Their paper presents a novel two-layered approach to classify DoH traffic from non-DoH at layer 1 and characterize DoH traffic at layer 2. They generated a labeled dataset by capturing Benign-DoH, Malicious-DoH and non-DoH encrypted traffic in the network premises. They also proposed a new feature-set based on time-series representation of traffic flows by introducing the concept of packet clumps and demonstrating the effectiveness of this feature set in encrypted traffic characterization.

Lotfollahi et al. focused on feature extraction and classification to handle network traffic characterization and identify end-user applications [10]. Authors used stacked AutoEncoder and Convolutional Neural Network for network classification. Leroux et al. also worked on machine learning models based on size and timing features to fingerprint VPN and ToR encrypted traffic [11]. In addition, a Quick UDP Internet Connection (QUIC) protocol based CNN classifier is developed by Tong et al. which used flow-based and packet-based features to identify some QUIC protocol based on Google services with an accuracy of approximately 99% [12]. Wang et al. proposed an end-to-end encrypted traffic classification method with ID-CNN which integrated feature extraction, feature selection, and classifier into one end-to-end network [13]. Buczak et al. used traffic captured at each device and analyzed DNS tunneled data [14]. Authors used random forest with features selected from the previous works on this area to show that the method is sufficiently effective even when the classifier had not seen the tunneling technique in the training set. They also determined the features that worked better with random forest classifier.

III. Data Processing

A. Dataset Description

In this paper, We use the CIRA-CIC-DoHBrw-2020 dataset [4]. The authors of this dataset captured benign and malicious DoH traffic along with non-DoH traffic, and used a two-layered approach to classify them. The first layer is to classify data instances as DoH or non-DoH, and to pass the data

TABLE I
COUNT OF EACH LABEL.

Label Name	Count
DoH	269,643
Non-DoH	897,493
Benign	19,807
Malicious	249,836

TABLE II
FEATURES OF THE CIRA-CIC-DOHBRW-2020 DATASET.

#	Feature
1	SourceIP
2	DestinationIP
3	SourcePort
4	DestinationPort
5	TimeStamp
6	Duration
7	FlowBytesSent
8	FlowSentRate
9	FlowBytesReceived
10	FlowReceivedRate
11	PacketLengthVariance
12	PacketLengthStandardDeviation
13	PacketLengthMean
14	PacketLengthMedian
15	PacketLengthMode
16	PacketLengthSkewFromMedian
17	PacketLengthSkewFromMode
18	PacketLengthCoefficientofVariation
19	PacketTimeVariance
20	PacketTimeStandardDeviation
21	PacketTimeMean
22	PacketTimeMedian
23	PacketTimeMode
24	PacketTimeSkewFromMedian
25	PacketTimeSkewFromMode
26	PacketTimeCoefficientVariation
27	ResponseTimeTimeVariance
28	ResponseTimeTimeStandardDeviation
29	ResponseTimeTimeMean
30	ResponseTimeTimeMedian
31	ResponseTimeTimeMode
32	ResponseTimeTimeSkewFromMedian
33	ResponseTimeTimeSkewFromMode
34	ResponseTimeTimeCoefficientofVariation

instances classified as DoH to the second layer. The second layer is to classify data instances as benign-DoH or malicious-DoH. To give this dataset some more context and on how it was produced, DoH traffic, including benign DoH and malicious-DoH, was generated by accessing top 10k Alexa websites, and used browsers and DNS tunneling tools which support DoH protocol, respectively. TABLE I shows the count of each label in the CIRA-CIC-DoHBrw-2020 dataset. The count of DoH traffic is the sum of the counts of benign and malicious DoH traffic. The count values in the table show that the CIRA-CIC-DoHBrw-2020 dataset is highly imbalanced. TABLE II shows the original features of the CIRA-CIC-DoHBrw-2020 dataset.

B. Data Processing Pipeline

The data processing pipeline consists of the following parts:

1) Feature Selection: to select appropriate features without fitting to some specific environments or time period.
2) Missing Data Handling: to drop the data instances with missing values in one or more features.
3) Train-Test Split: to split the original dataset into two parts, including the training set and the test set. In this paper, we split 20% of the dataset as the test set, and take the other 80% as the training set.
4) Data Imbalance Handling: to balance the numbers of data instances of different labels.
5) One Hot Encoding: to convert categorical features to numerical features in order to facilitate machine learning methods.
6) Feature Scaling: to scale and shift the feature values to some ranges that are suitable for machine learning methods. In this paper, we use the min-max scaling to transform the features by scaling each feature to [0, 1].

Detailed descriptions of the critical parts of the data processing pipeline, including the feature selection and the data imbalance handling, are provided in the following subsections.

C. Feature Selection

Among all features in the CIRA-CIC-DoHBrw-2020 dataset shown in TABLE II, we argue that the first four features are environment-specific. Different network environments would certainly have different IP addresses and port numbers. Moreover, it is very easy for networks attackers to modify IP addresses in network packets. On the other hand, the time stamp feature and the duration feature are time-specific. A model trained by using all the 34 features would not be generalized to other network environments. Therefore, we drop the first six features in this work. Note that the dropping of the first six features would lead to performance degradation, since the test set to evaluate the model performance also comes from the original dataset. More accurate model is required to compensate the performance degradation caused be the feature dropping.

D. Data Imbalance Handling

As TABLE I shows, the CIRA-CIC-DoHBrw-2020 is highly imbalanced. We use the resampling technique to deal with the data imbalance problem. The data imbalance handling is performed after the train-test split mentioned above. In the second layer, the DoH set and *Non-DoH* set are resampled to 161,796 data instances. In the second layer, the *benign* set and the *malicious* set are both resampled to 3,269 data instances.

IV. DEEP LEARNING-BASED CLASSIFICATION MODELS FOR DOH TRAFFIC

In this paper, we design the following two deep learning models:

1) Long Short-Term Memory (LSTM)
2) Bi-directional Long Short-Term Memory (BiLSTM)

We use the grid search method to optimize their corresponding hyperparameter combinations. TABLE III and IV show the optimization results for both layers of the LSTM model and the BiLSTM model, respectively. We use TimeSeriesSplit in the scikit-learn as the cross validator.

TABLE III
HYPERPARAMETER COMBINATION FOR THE LSTM MODEL.

Layer 1		Layer 2	
Batch Size	20	Batch Size	10
Epochs	100	Epochs	100
Optimizer	Adam	Optimizer	Adam
Learning Rate	0.001	Learning Rate	0.001
Activation	ReLU	Activation	Softsign
Dropout	0.1	Dropout	0.9
Hidden Neurons	20	Hidden Neurons	30

TABLE IV
HYPERPARAMETER COMBINATION FOR THE BILSTM MODEL.

Layer 1		Layer 2	
Batch Size	80	Batch Size	80
Epochs	100	Epochs	100
Optimizer	Nadam	Optimizer	Adam
Learning Rate	0.001	Learning Rate	0.01
Activation	tanh	Activation	tanh
Dropout	0.2	Dropout	None
Hidden Neurons	30	Hidden Neurons	30

V. NUMERICAL RESULTS AND DISCUSSIONS

Fig. 1 and 2 show the confusion matrices for layer 1 and layer 2 of the LSTM model, respectively. In layer 1, 0 means DoH and 1 means Non-DoH. In layer 2, 0 means benign and 1 means malicious. Fig. 1 shows that the LSTM model provides 91.1% recall for DoH traffic and 96.6% recall for Non-DoH traffic in layer 1, and Fig. 2 shows that the LSTM model provides 78.4% recall for benign traffic and 99.3% recall for malicious traffic in layer 2.

Fig. 3 and 4 show the confusion matrices for layer 1 and layer 2 of the BiLSTM model, respectively. Fig. 3 shows that the BiLSTM model provides 98.0% recall for DoH traffic and 99.8% recall for Non-DoH traffic in layer 1, and Fig. 4 shows that the BiLSTM model provides 98.0% recall for benign traffic and 99.9% recall for malicious traffic in layer 2. These results show that the BiLSTM model performs better than the LSTM model does, while both models perform well.

TABLE V
COMPARISONS OF THE ACCURACY AND TIME TAKEN FOR LAYER 1.

Train			
Model	Mean Accuracy	Standard Deviation	Time Taken (mm:ss)
LSTM	0.944	0.011	45:22
BiLSTM	0.990	0.000	31:42
Test			
Model	Mean Accuracy	Standard Deviation	Time Taken (mm:ss)
LSTM	0.944	0.011	00:03
BiLSTM	0.990	0.000	00:14

TABLE V and VI show the comparisons of the accuracy and time taken for layer 1 and layer 2, respectively. The results show that both the LSTM and the BiLSTM models perform well for training and test accuracy. The BiLSTM model performs better than the LSTM model does for both the accuracy and the time taken.

Note that the time taken for testing considers the calculation of all data instances in the test set. Therefore, the time taken to classify each data instance is in the order of milliseconds. This validates the efficiency of the proposed method from the time complexity perspective.

VI. CONCLUSION

In this paper, we investigate the DoH traffic classification by using the LSTM and BiLSTM models. We design an appropriate data processing pipeline to process the CIRA-CIC-DoHBrw-2020 dataset, including feature selection and data imbalance handling, in order to facilitate the generalization of deep learning models. We develop truly generalized deep

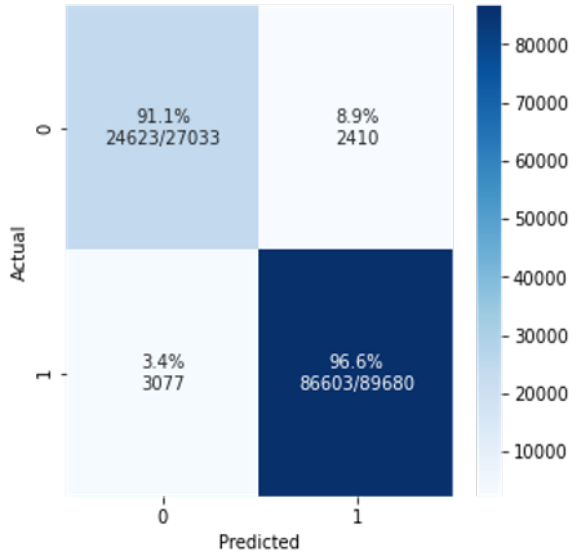

Fig. 1. Confusion matrix for layer 1 of the LSTM model.

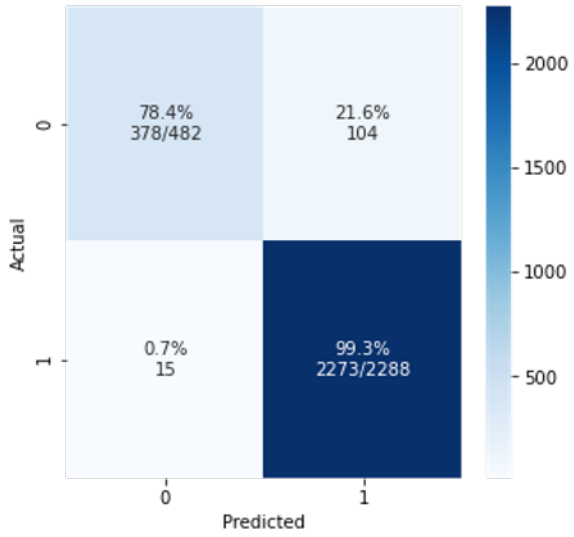

Fig. 2. Confusion matrix for layer 2 of the LSTM model.

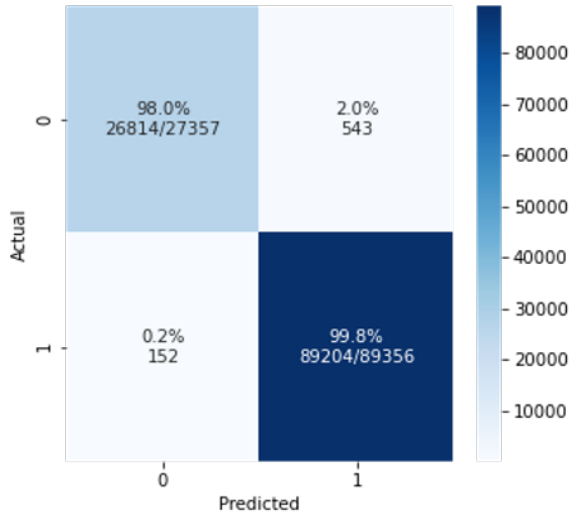

Fig. 3. Confusion matrix for layer 1 of the BiLSTM model.

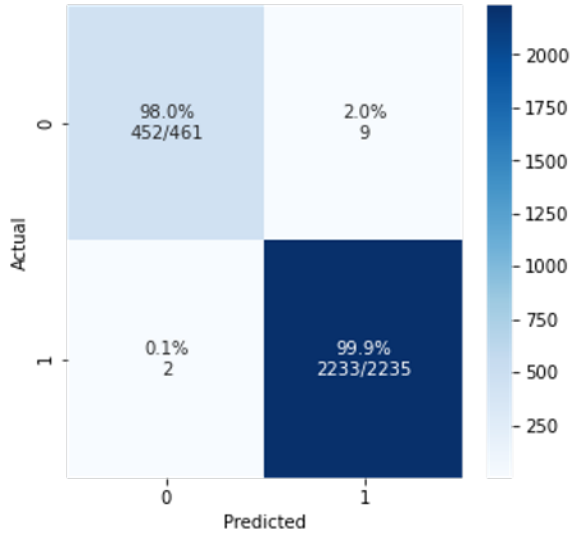

Fig. 4. Confusion matrix for layer 2 of the BiLSTM model.

TABLE VI
COMPARISONS OF THE ACCURACY AND TIME TAKEN FOR LAYER 2.

Train			
Model	Mean Accuracy	Standard Deviation	Time Taken (mm:ss)
LSTM	0.950	0.007	01:52
BiLSTM	0.998	0.004	00:51
Test			
Model	Mean Accuracy	Standard Deviation	Time Taken (mm:ss)
LSTM	0.952	0.004	00:02
BiLSTM	0.994	0.005	00:01

learning models to classify DoH traffic with high accuracy and low latency. The BiLSTM model performs better than the LSTM model does for both the accuracy and the time taken.

In the future, we will investigate the feasibility to apply the proposed models to embedded systems with limited resources.

REFERENCES

[1] S. Rezaei and X. Liu, "Deep Learning for Encrypted Traffic Classification: An Overview," *IEEE Communications Magazine*, vol. 57, no. 5, pp. 76–81, 2019.

[2] F. Laghrissi, S. Douzi, K. Douzi, and B. Hssina, "Intrusion Detection Systems using Long Short-Term Memory (LSTM)," *Journal of Big Data*, vol. 8, no. 1, May 2021.

[3] E. Vasilomanolakis, S. Karuppayah, M. Mühlhäuser, and M. Fischer, "Taxonomy and Survey of Collaborative Intrusion Detection," *ACM Comput. Surv.*, vol. 47, no. 4, May 2015.

[4] "CIRA-CIC-DoHBrw-2020." [Online]. Available: https://www.unb.ca/cic/datasets/dohbrw-2020.html

[5] F. Karim, S. Majumdar, H. Darabi, and S. Chen, "LSTM Fully Convolutional Networks for Time Series Classification," arXiv preprint arXiv:1709.05206, 2017.

[6] Y. Banadaki and S. Robert, "Detecting Malicious DNS over HTTPS Traffic in Domain Name System using Machine Learning Classifiers," *Journal of Computer Sciences and Applications*, vol. 8, pp. 46–55, Aug. 2020.

[7] J. M. Johnson and T. M. Khoshgoftaar, "Survey on Deep Learning with Class Imbalance," *Journal of Big Data*, vol. 6, no. 1, Mar. 2019.

[8] S. K. Singh and P. K. Roy, "Detecting Malicious DNS over HTTPS Traffic Using Machine Learning," in *2020 International Conference on Innovation and Intelligence for Informatics, Computing and Technologies (3ICT)*, 2020.

[9] M. MontazeriShatoori, L. Davidson, G. Kaur, and A. Habibi Lashkari, "Detection of DoH Tunnels using Time-series Classification of Encrypted Traffic," in *2020 IEEE Intl Conf on Dependable, Autonomic and Secure Computing, Intl Conf on Pervasive Intelligence and Computing, Intl Conf on Cloud and Big Data Computing, Intl Conf on Cyber Science and Technology Congress (DASC/PiCom/CBDCom/CyberSciTech)*, 2020, pp. 63–70.

[10] M. Lotfollahi, M. Jafari Siavoshani, R. Shirali Hossein Zade, and M. Saberian, "Deep Packet: A Novel Approach for Encrypted Traffic Classification Using Deep Learning," *Soft Computing*, vol. 24, no. 3, pp. 1999–2012, Feb 2020.

[11] S. Leroux, S. Bohez, P.-J. Maenhaut, N. Meheus, P. Simoens, and B. Dhoedt, "Fingerprinting Encrypted Network Traffic Types Using Machine Learning," in *NOMS 2018 - 2018 IEEE/IFIP Network Operations and Management Symposium*, 2018.

[12] V. Tong, H. A. Tran, S. Souihi, and A. Mellouk, "A Novel QUIC Traffic Classifier Based on Convolutional Neural Networks," in *2018 IEEE Global Communications Conference (GLOBECOM)*, 2018.

[13] W. Wang, M. Zhu, J. Wang, X. Zeng, and Z. Yang, "End-to-End Encrypted Traffic Classification with One-Dimensional Convolution Neural Networks," in *2017 IEEE International Conference on Intelligence and Security Informatics (ISI)*, 2017, pp. 43–48.

[14] A. L. Buczak, P. A. Hanke, G. J. Cancro, M. K. Toma, L. A. Watkins, and J. S. Chavis, "Detection of Tunnels in PCAP Data by Random Forests," in *Proceedings of the 11th Annual Cyber and Information Security Research Conference*, ser. CISRC '16. New York, NY, USA: Association for Computing Machinery, 2016.