Zero-Day Traffic Identification Using One-Dimension Convolutional Neural Networks And Auto Encoder Machine

Dong Jin*, Jinsen Xie*, Shuangwu Chen*, Jian Yang *, Xinmin Liu †, Wei Wang †
*University of Science and Technology of China, Hefei, China
{kingdon, jasenxie}@mail.ustc.edu.cn, {chensw, jianyang}@ustc.edu.cn
†New H3C Technologies Co., Beijing, China
{liuxinmin, david_wang}@h3c.com

***Abstract*—Network traffic identification plays an important role in traffic engineering, anomaly detection and traffic billing. Recently, the machine learning and deep learning based algorithms have made a great success in identifying the known applications, where the training set and the test set are supposed to contain the same traffic classes. However, in a realistic scenario, the network traffic classifier may suffer from a low identification accuracy due to the substantial zero-day (unknown) traffic. The essential to solve this problem is to find the boundary between the known and zero-day traffic, which has not been well studied before. In this paper, based on the fact that for an Auto Encoder (AE) machine, the reconstruction error of a zero-day class is generally larger than that of a known class, we propose a zero-day traffic identification method using one-Dimension Convolutional Neural Networks (1D-CNN) and AE machine. In order to further improve the identification accuracy, we propose an algorithm to estimate the confidence possibility of the identification results based on the Extreme Value Theory. The experiments conducted on realistic traffic datasets demonstrate that our method has a great improvement in identification accuracy than the benchmarks.**

I. Introduction

As an essential part of network management, traffic identification plays an important role in: 1) traffic engineering: optimize the bandwidth allocation and routing strategy among applications for differentiated quality of service requirements; 2) anomaly detection: prevent malware and avoid network intrusion; 3) traffic billing: charge separately by the class of applications. Recently, motivated by the advents in machine learning (ML) and deep learning (DL), lots of works have been done to classify the network traffic using a well trained model. These methods have been proved to be effective when the training set and the test set contain the same traffic classes. However, in a realistic scenario, it's intractable for a pre-trained traffic classifier to deal with unknown traffic, and thus may suffer from a low identification accuracy. The underlying problem is that the realistic network traffic may contain substantial zero-day traffic classes, which are unknown for the training model. This problem is made even worse, especially when new classes of applications keep popping up. According to the report from China Internet Network Information Center (CNNIC) [1], the number of network applications in China is up to 4.49 million by December 2018, which has a 10.3% increase compared to 2017.

Annex to ISBN 978-3-903176-28-7 © 2020 IFIP

Recent advents in traffic identification focus on ML or DL [2]. ML based methods leverage the statistical features of network flows, such as time series, protocol types and dynamics of packet size. Jonas et al. [3] adopted an unsupervised traffic flow classification using time interval based features. Furat et al. [4] classified encrypted traffic using different ML algorithms including Support Vector Machine (SVM), Naive Bayesian, C4.5 and Multilayer Perceptron (MLP). However, the accuracy of ML based methods is affected by the features which are empirically selected. To address this problem, the DL based methods can extract latent features from raw data packets using a neural network [5]. Zhou et al. [6] applied an improved CNN suitable for indefinite length dataset to traffic identification. Lotfollahi et al. [7] used Stacked Auto Encoder machine (SAE) and CNN to classify network traffic. Wang et al. [8] utilized a 1D-CNN model to identify encrypted traffic. Although these methods achieve a high accuracy, most of them are applied to a closed test scenario. As a result, the zero-day traffic is misclassified as known categories.

The key to identify zero-day traffic is to find the boundary between zero-day and known classes. Este et al. [9] modified a one-class SVM to identify zero-day traffic, which established different boundaries for every known class in the training set. If a test flow was not within any boundary of known classes, it would be identified as a zero-day class. However, this method was very complex and had to establish boundaries between each known class with substantial labeled samples. In [10], in order to label the unknown traffics in the training set, the features of the traffic were clustered. Then, with the labeled data, a supervised identification model was well trained for testing. However, the traffic class used for training should be the same as that for testing. Without any prior knowledge, it is intractable to identify the zero-day traffic for an open-set scenario, where the zero-day traffic do not appear in the training set, which has not been well studied yet.

To address this problem, in this paper, we propose a novel zero-day traffic identification (ZTI) method using 1D-CNN

and AE machine. The convolutional layer of 1D-CNN is employed to extract the discriminative latent features for different traffic classes. Due to the fact that for an AE machine, reconstruction errors of zero-day classes are generally larger than those of known classes. Then, an AE machine is used to reconstruct these latent features. To further improve the identification accuracy, we propose an algorithm to estimate the confidence possibility of the identification results based on the Extreme Value Theory. The experiments conducted on realistic traffic datasets demonstrate that our method has a great improvement in identification accuracy than the benchmarks.

II. Zero-day traffic identification using one-Dimension Convolutional Neural Networks and Auto Encoder machine

For the classifier in a realistic network traffic identification scenario, suppose that the training set S_R contains K known traffic classes $\{c_1,c_2,\cdots,c_K\}$, the test set S_T also contains several zero-day traffic classes other than K known classes. Given a flow in the test set, our purpose is to identify if it belongs to a zero-day class. If not, we want to identify which known class it belongs to.

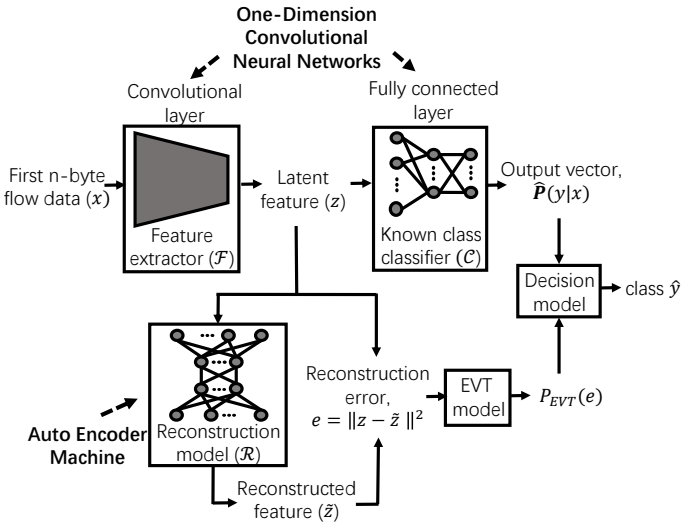

Fig. 1. Framework of the zero-day traffic identification

We propose a ZTI method using 1D-CNN and AE machine. As shown in Fig. 1, it contains five models, which are the feature extractor, the known class classifier, the reconstruction model, the EVT model and the decision model. The feature extractor and the known class classifier are composed of the convolutional layer and the fully connected layer of a 1D-CNN, the reconstruction model is composed of an AE machine. We use the first n bytes of each flow data packets as input. The feature extractor extracts more discriminative latent features from raw flow data for an accurate identification. The known class classifier utilizes the latent features to predict which known class it belongs to. And the reconstruction model produces reconstructed features using latent features. Due to the fact that the reconstruction errors of the zero-day classes are larger than those of known classes, the EVT model obtains the confidence possibility reflecting the likelihood of the input belonging to a zero-day class. Finally, the decision model will output the final prediction class. In the following parts, we will explain each model and how to obtain its parameters.

A. Feature extractor and known class classifier

An accurate identification for known classes is an important basis for ZTI. 1D-CNNs perform very well in feature extraction for network traffic identification [7], since 1D-CNNs can capture latent features between adjacent bytes in raw flow packets through training, which aims to find more discriminative patterns for different traffic classes, and finally achieves an accurate identification. Therefore, we utilize the convolutional layer and the fully connected layer of a 1D-CNN as the feature extractor and the known class classifier.

Then we introduce how to learn the parameters of the feature extractor and the known class classifier. The training set S_R can be denoted as pairs $(\boldsymbol{x}, \boldsymbol{y}) = \{(x_1, y_1), (x_2, y_2), \cdots, (x_N, y_N)\}$, where x_i denotes the i^{th} raw input data and $y_i \in \{1, 2, \cdots, K\}$ is its label. The feature extractor maps the input x_i to the corresponding latent feature z_i, which is used to produce the normalized output vector $\hat{\boldsymbol{P}}(y|x_i) = (\hat{P}(1|x_i), \hat{P}(2|x_i), \cdots, \hat{P}(K|x_i))$ by the known class classifier. Where $\hat{P}(j|x_i)$ reflects the possibility of x_i belonging to known class j. So the feature extractor and the known class classifier can be described as the function $\mathcal{F} : x \to z$ and $\mathcal{C}: z \to \hat{\boldsymbol{P}}(y|x)$. And we denote the parameters of $\mathcal{F}$ and $\mathcal{C}$ as $\theta_{\mathcal{F}}$ and $\theta_{\mathcal{C}}$. Obviously, $\hat{\boldsymbol{P}}(y|x_i)$ can be obtained from $\mathcal{C}(\mathcal{F}(x_i))$, the accuracy of the known classifier is also affected by the feature extractor. We jointly train parameters $\theta_{\mathcal{F}}$ and $\theta_{\mathcal{C}}$ with the cross-entropy loss function $\mathcal{L}_{\mathcal{C}}$, which can be defined as follows:

$$\mathcal{L}_{\mathcal{C}} = -\frac{1}{M}\sum_{i=1}^{M}\sum_{j=1}^{K}\mathbb{I}_{y_i}(j)\log\left[\hat{P}(y=j|x_i)\right] \tag{1}$$

where M denotes the batch size and $\mathbb{I}_{y_i}$ is the one-hot label vector of y_i.

B. Reconstruction model

As mentioned before, the key for ZTI is to find the boundary between the known and zero-day traffic. However, we don't have access to zero-day classes in the training set. Hence, the boundary must rely on the known classes. AE machine as an unsupervised deep learning method is used to reconstruct its input so that the network learns the latent pattern of the input. Since zero-day classes don't appear in the training procedure, the AE machine has poor performance in reconstructing zero-day classes compared to the known classes. Reconstruction errors of zero-day classes are generally larger than those of known classes, which can be a good boundary between zero-day and known classes. Thus, we utilize AE machine composed of a fully connected network as the reconstruction model.

After training the feature extractor, we can get a corresponding latent feature z_i for each raw flow x_i with learned $\mathcal{F}$. The

reconstruction model produces the reconstructed features $\widetilde{z}_i$ based on the latent features z_i. Then we train the reconstruction model $\mathcal{R}$ with the L^2-norm loss function $\mathcal{L}_{\mathcal{R}}$, which is defined as follows:

$$\mathcal{L}_{\mathcal{R}} = \frac{1}{M}\sum_{i=1}^{M} \|z_i - \widetilde{z}_i\|_2^2 \tag{2}$$

C. EVT model

Although reconstruction errors of known classes are larger than those of zero-day classes, we only have a knowledge of reconstruction errors of known class samples. So the decision threshold between zero-day classes and known classes must rely on the known class samples with large reconstruction errors.

EVT [11] is a statistical theory that deals with situations far away from the median of the probability distribution, and is often used to analyze situations that rarely occur. EVT has been successfully applied to the field of open-set recognition [12], which mainly investigates how to identify the images that don't belong to any class of the training set. Inspired by these studies, we apply EVT to identify zero-day traffic. We calculate the reconstruction errors for all training samples with trained reconstruction model using the following equation.

$$e_i = \|z_i - \widehat{z}_i\|_2^2, i = 1, 2, \cdots, N \tag{3}$$

Then we sort all reconstruction errors in an ascending order, we utilize EVT to model the last η ($\eta << N$) reconstruction errors, where η is a hyper parameter. Weibull distribution is a statistical distribution that follows EVT, its cumulative distribution function is defined as follows:

$$P_{EVT}(x) = 1 - e^{-\left(\frac{x-\tau}{\sigma}\right)^m} \tag{4}$$

where τ ($\tau < x$) is the location parameter, m ($m > 0$) and σ ($\sigma > 0$) are the shape parameter and the scale parameter. After modeling with a maximum likelihood method, we get our EVT model's Weibull distribution parameters (m, τ, σ). Then for a given reconstruction error e, The value of $P_{EVT}(e)$ reflects the confidence possibility that the corresponding raw flow belongs to a zero-day class. And as you can see from Eq. (4), the larger value of the reconstruction error e is, the larger value of $P_{EVT}(e)$ is.

D. Decision model

In this part, we will explain how our method classifies a test flow. As is seen from Fig. 1, the feature extractor maps the test flow x to the latent feature z. On the one hand, the known class classifier uses this latent feature z as input to produce the normalized output vector $\hat{\boldsymbol{P}}(y|x)$, where $y \in \{1, 2, \cdots, K\}$. Then we obtain the known class which the test flow may belong to as follows:

$$y^* = \arg\max_j \hat{P}(y = j|x), j = 1, 2, \cdots, K \tag{5}$$

$$P_{y^*} = \hat{P}\left(y = y^*|x\right) \tag{6}$$

where y^* is the predicted known class, and P_{y^*} reflects the confidence for this prediction. On the other hand, the reconstruction model produces the reconstructed feature $\widetilde{z}$ based on the latent feature z. The EVT model uses the reconstruction error e to obtain $P_{EVT}(e)$. According to $P_{EVT}(e)$ and P_{y^*}, then we obtain the final possibility P_z as Eq. (7), which reflects the likelihood of the test flow belonging to a zero-day class.

$$P_z = (1 - P_{y^*}) + \alpha P_{EVT}(e) \tag{7}$$

where α is a hyper parameter and its value can be designed from experiments. Finally, according to P_{y^*} and P_z, we have the identified traffic class as:

$$\hat{y} = \begin{cases} y^*, & \text{if } P_{y^*} > P_z \\ 0 \text{ (Zero-day)}, & \text{otherwise,} \end{cases} \tag{8}$$

where $\hat{y} = 0$ denotes a zero-day class.

III. EXPERIMENT

A. Experiment implementation

1) Dataset: We select three public datasets containing raw flow data. They are ISCX VPN dataset [13], USTC Malware data and USTC Benign data [14]. The traffic classes of three datasets are shown as TABLE I.

TABLE I
THE TRAFFIC CLASSES OF DATASETS

USTC Malware	USTC Benign	ISCX VPN
Htbot	Facetime	File
Geodo	Gmail	
Cridex	FTP	Email
Shifu	BitTorrent	
Neris	Outlook	Chat
Zeus	MySQL	
Miuref	Skype	P2P
Tinba	SMB	
Virut	Weibo	Streaming
Nsis-ay	Warcraft	VoIP

Every dataset is divided into known classes, validation classes and zero-day classes. 85% of known class samples are used to train our network models (1D-CNN and AE machine). The remaining 15% of known class samples form the validation set with validation classes to determine the value of the hyper parameter η and α, and finally form the test set with zero-day classes. To obtain enough features of each flow, we utilize the first 784 bytes of each flow packets as input. Hence, truncation and zero-padding are required inevitably.

2) Evaluation metrics: To evaluate our proposed method, we use accuracy (ACC), precision (PR), recall (RC) and f1 score (F1) as evaluation metrics. ACC is defined as the ratio of the samples classified correctly and all test samples. PR, RC and F1 are evaluation metrics for a single class.

B. Experiment Results

1) 1D-CNN's performance in feature extraction for traffic identification task: Our feature extractor is designed to map the raw input data to the 1100-dimension (1100D) latent

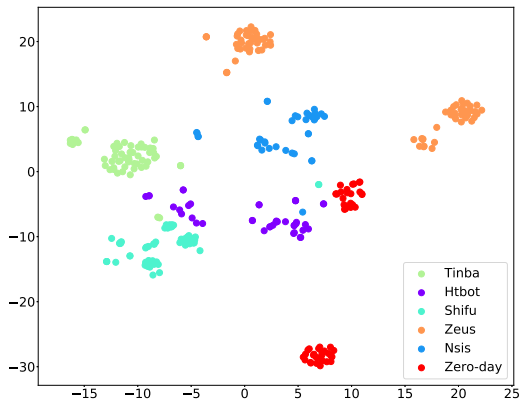

Fig. 2. Latent features in 2D space

Fig. 3. The distribution of reconstruction errors of known and zero-day classes in USTC Malware dataset

features. we set Htbot, Nsis, Shifu, Tinba, and Zeus as the known classes, and set Cridex as the zero-day class in the USTC Malware dataset. As is shown in Fig. 2, we present the 2D latent features for visualization using the t-SNE dimension reduction method. We can find that latent features of different classes hardly overlap even in 2D space, which leads to an accurate identification.

2) Reconstruction errors of known and zero-day classes: As you can see from Fig. 3, we compute reconstruction errors of test flows in USTC Malware dataset containing 1240 known samples and 1246 zero-day samples. Then we obtain the distribution of reconstruction errors of known and zero-day classes. We can see all reconstruction errors from 0 to 0.06 are generated from known classes, and all reconstruction errors of zero-day classes are from 0.08 to 0.18. In addition, reconstruction errors of the known classes from 0.08 to 0.18 account for less than 3% of all test flows. So we can get the conclusion that reconstruction errors of the zero-day classes are generally larger than those of the known classes.

3) Identification performance on different dataset: We compute the average F1 score and the accuracy in three datasets with three different methods. We set 5 known classes and 1 zero-day class in each dataset, and zero-day samples account for 38%, 24% and 37% in three test sets. As you can see from Fig. 4 and Fig. 5, Since 1D-CNN misclassifies zero-day samples into known classes, it has a poor identification performance in ZTI task. Our proposed method (CNNAE) achieves the average F1 scores of 0.886, 0.922 and 0.830 for USTC Malware, USTC Benign and ISCX VPN datasets, and they are obviously higher than the average F1 scores of Oneclass svm [9]. Similarly, the accuracies of Oneclass svm are all less than 0.8 in three datasets. The accuracies of our method are 0.921, 0.920 and 0.848.

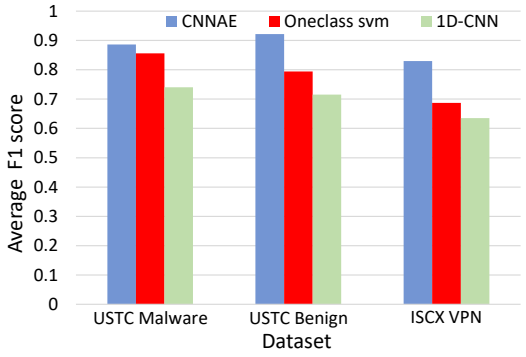

Fig. 4. Average F1 score of three datasets

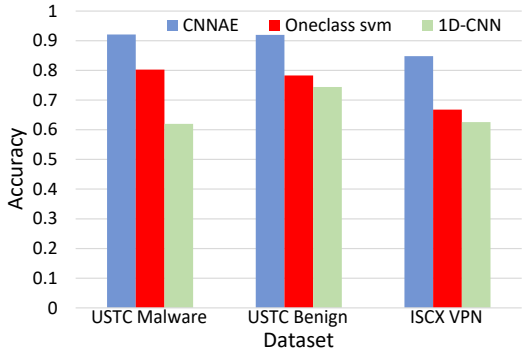

Fig. 5. Accuracy of three datasets

4) The effect of the number of zero-day classes: To evaluate the effect of the number of zero-day classes on our proposed method's performance, as is seen from TABLE II, we set Outlook, Skype, SMB, Weibo and Warcraft as the known classes in the USTC Benign dataset, and set 1 to 4 other classes (Bittorrent, Facetime, FTP and Gmail) as zero-day classes. We only change the number of the zero-day classes in this experiment. We can find that all the average values of three evaluation metrics (PC,RC and F1) are over 0.85, and they are all over 0.90 when the number of zero-day classes is 1. On the other hand, the F1 scores of zero-day class generally decrease as the number of zero-day classes increases. However, it is over 0.80 even there are 4 zero-day classes. When the number of zero-day classes increases from 1 to 2, the F1 of zero-day classes has a serious decline. The possible reason is that the reconstruction errors of newly added zero-day class (Facetime) are relatively small, which make our proposed method misclassify some zero-day samples into a known class.

For 1 zero-day class, we make a normalized confusion matrix to have a further study for our proposed method's performance. As is seen from Fig. 6, we can find about 20% of Skype and SMB samples are misclassified to the zero-day

TABLE II
THE EFFECT OF THE NUMBER OF ZERO-DAY CLASSES ON CNNAE'S PERFORMANCE

Number of zero-day classes	1			2			3			4		
Traffic class	**PR**	**PC**	**F1**	**PR**	**PC**	**F1**	**PR**	**PC**	**F1**	**PR**	**PC**	**F1**
Outlook	0.998	0.917	0.956	0.785	0.891	0.835	0.695	0.908	0.788	0.579	0.905	0.707
Skype	1.00	0.818	0.900	1.000	0.820	0.901	1.000	0.962	0.981	1.000	0.962	0.981
SMB	0.944	0.781	0.855	0.822	0.788	0.805	0.806	0.880	0.842	0.794	0.889	0.839
Weibo	0.930	0.990	0.959	0.888	0.990	0.936	0.853	0.990	0.916	0.800	0.990	0.885
Warcraft	1.000	0.970	0.997	1.000	0.995	0.997	1.000	0.996	0.998	1.000	0.996	0.998
Zero-day	0.784	0.970	**0.867**	0.816	0.806	**0.811**	0.923	0.775	**0.842**	0.935	0.729	**0.819**
Average	**0.943**	**0.912**	**0.922**	**0.885**	**0.882**	**0.881**	**0.879**	**0.919**	**0.894**	**0.851**	**0.912**	**0.871**

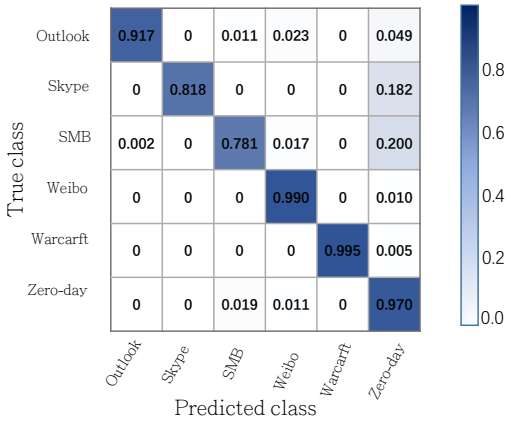

Fig. 6. Normalized confusion matrix of USTC Benign dataset.

class due to the relatively large reconstruction errors from Skype and SMB. On the other hand, almost few (less than 3%) zero-day class samples are misclassified to the known classes. So the zero-day class has a small PR value and a large PC value in TABLE II for 1 zero-day class. It is worth mentioning that this result can be seen from Fig. 3, a few known class samples have large reconstruction errors, but few zero-day class samples have small reconstruction errors.

IV. CONCLUSION

We proposed a novel zero-day traffic identification method based on two deep learning algorithms (1D-CNN and AE machine). We utilized the convolutional layer and the fully connected layer of 1D-CNN as the feature extractor and the known classifier. And due to the fact that for an AE machine, the reconstruction errors of zero-day classes were generally larger than those of known classes, the zero-day traffic were identified. We proposed an algorithm to estimate the confidence possibility of the identification results based on the Extreme Value Theory. The experiments conducted on realistic traffic datasets demonstrated that our method had a great improvement in identification accuracy than the benchmarks.

V. ACKNOWLEDGMENT

This work was supported by National Key R&D Program of China (No.2018YFF01012200) and Anhui Provincial Natural Science Foundation (No.1908085QF266). This work was also supported by the Fundamental Research Funds for the Central Universities (No. WK2100000009). The corresponding author is Shuangwu Chen.

REFERENCES

[1] "The 43rd china statical report on internet development," http://www.cnnic.cn/hlwfzyj/hlwxzbg/hlwtjbg/201902/P020190318523029756345.pdf, february, 2019.

[2] B. Ma, H. Zhang, Y. Guo, Z. Liu, and Y. Zeng, "A summary of traffic identification method depended on machine learning," in *2018 International Conference on Sensor Networks and Signal Processing (SNSP)*, Oct 2018, pp. 469–474.

[3] J. Höchst, L. Baumgärtner, M. Hollick, and B. Freisleben, "Unsupervised traffic flow classification using a neural autoencoder," in *2017 IEEE 42nd Conference on Local Computer Networks (LCN)*, 2017, pp. 523–526.

[4] F. Al-Obaidy, S. Momtahen, M. F. Hossain, and F. Mohammadi, "Encrypted traffic classification based ml for identifying different social media applications," in *2019 IEEE Canadian Conference of Electrical and Computer Engineering (CCECE)*, May 2019, pp. 1–5.

[5] S. Rezaei and X. Liu, "Deep learning for encrypted traffic classification: An overview," *IEEE Communications Magazine*, vol. 57, no. 5, pp. 76–81, May 2019.

[6] H. Zhou, Y. Wang, and M. Ye, "A method of cnn traffic classification based on sppnet," in *2018 14th International Conference on Computational Intelligence and Security (CIS)*, Nov 2018, pp. 390–394.

[7] M. Lotfollahi, R. S. H. Zade, M. J. Siavoshani, and M. Saberian, "Deep packet: a novel approach for encrypted traffic classification using deep learning," *Soft Computing*, pp. 1–14, 2017.

[8] 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)*, July 2017, pp. 43–48.

[9] A. Este, F. Gringoli, and L. Salgarelli, "Support vector machines for tcp traffic classification," *Computer Networks*, vol. 53, no. 14, pp. 2476 – 2490, 2009.

[10] J. Zhang, X. Chen, Y. Xiang, W. Zhou, and J. Wu, "Robust network traffic classification," *IEEE/ACM Trans. Netw.*, vol. 23, no. 4, pp. 1257–1270, Aug. 2015.

[11] W. J. Scheirer, L. P. Jain, and T. E. Boult, "Probability models for open set recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 11, pp. 2317–2324, Nov 2014.

[12] P. Oza and V. M. Patel, "Deep cnn-based multi-task learning for open-set recognition," *CoRR*, vol. abs/1903.03161, 2019.

[13] G. Draper-Gil., A. H. Lashkari., M. S. I. Mamun., and A. A. Ghorbani., "Characterization of encrypted and vpn traffic using time-related features," in *Proceedings of the 2nd International Conference on Information Systems Security and Privacy - Volume 1: ICISSP*, INSTICC. SciTePress, 2016, pp. 407–414.

[14] Wei Wang, Ming Zhu, Xuewen Zeng, Xiaozhou Ye, and Yiqiang Sheng, "Malware traffic classification using convolutional neural network for representation learning," in *2017 International Conference on Information Networking (ICOIN)*, Jan 2017, pp. 712–717.