FLOWGAN: DEEP LEARNING BASED ENCRYPTED NETWORK TRAFFIC CLASSIFICATION WITH CLASS IMBALANCE

Pan Wang[1] , ZiXuan Wang [1] , ShuHang Li [1] , Chen Huang [1] ,Feng Xiang [2]

[1] School of Modern Posts, University of Nanjing University
of Posts and Telecommunications, Nanjing,CO 210003, China

[2] YTO Express Co., Ltd., Shanghai,CO 201705,China

Corresponding author: Pan Wang (e-mail: wangpan@njupt.edu.cn)

Abstract

It is crucial to accurately identify the type of traffic and application so that it can enable various policy-driven network managements and security monitoring. However, with more and more adoptions of encryption over internet applications, it brings a lot of challenges about traffic classification. Although classical machine learning approaches and recent proposed deep learning methods can solve many issues that port and payload based methods cannot solve, it still has some limitations, one of which is imbalanced property of network traffic data. In this paper, we proposed a deep learning method called FlowGAN to tackle with the problem of class imbalance for traffic classification. As an instance of Generative Adversarial Network (GAN), FlowGAN leverages the superority of GAN's data augmentation to generate synthesized traffic data samples for minor classes. Furthermore, we trained a classical deep learning model, Multilayer perceptron (MLP) based network traffic classifier to evaluate the performance of FlowGAN. Based on public dataset 'ISCX', our experimental results show that our proposed FlowGAN can outperforms unbalanced dataset and balancing dataset by oversampling method in terms of data augmentation.

Key Words

traffic classification, encrypted traffic, deep learning, Generative Adversarial Network, class imbalance

1. Introduction

Network traffic identification and classification is an important research topic in the field of network management and security. Efficient, accurate and real-time traffic classification is of great practical significance to provide service quality assurance, dynamic access control and abnormal network behaviors detection. However, with the widespread adoptions of encryption techniques in internet applications, encrypted traffic has dramatically become a great challenge for network management and security monitoring.

Traditionally, the evolution of encrypted traffic classification technology has gone through three stages: port-based, payload-based and flow-based statistical characteristics. Port-based classification method infers application's type by assuming that most applications consistently use 'well known' TCP or UDP port numbers, however, the emergence of port camouflage, random port and tunneling technology makes these methods lose efficacy quickly. Payload-based methods, namely, DPI (Deep Packet Inspection) technology cannot deal with encrypted traffic because they need to match packet content and have high computational overhead [1, 2]. As a result, in order to attempt to solve the problem of encrypted traffic identification , flow-based methods emerged, which usually rely on statistical or time series features and employ Machine Learning (ML) algorithms, such as naive bayes(NB), support vector machine(SVM), decision tree, Random Forest(RF), k-nearest neighbor(KNN) [3]. Although classical machine learning approach can solve many issues that port and payload based methods cannot solve, it still has some limitations, such as handcrafted traffic features driven by domain-expert, time-consuming, unsuited to automation, rapidly outdated when compared to the evolution. Unlike most traditional ML algorithms, Deep Learning performs automatic feature extraction without human intervention, which undoubtedly makes it a highly desirable approach for traffic classification, especially encrypted traffic. Recent research work has demonstrated the superiority of DL methods in traffic classification [4], such as MLP, CNN, SAE, LSTM.

However, imbalanced class distribution of a dataset has posed a serious difficulty to most classifiers based on machine learning algorithms which assume a relatively balanced distribution [5]. Imbalanced class distribution is characterized as that there are many more instances of some classes than others. Network traffic classification is no exception due to the imbalanced property of network traffic data [6]. Therefore, how to overcome the problem of imbalanced class distribution in traffic dataset plays a very crucial role in the network traffic classification. However, very few research has been conducted focusing on examining the performance of traffic classification when facing the problem of imbalanced traffic data, especially deep learning methods.

In this paper, we present the design and development of Generative Adversarial Network (GAN) based traffic data augmenting method called FlowGAN to generate synthesized samples for encrypted traffic classification. The synthesized data is then combined with the original (viz. real) data to construct the new traffic training dataset. As a proof of concept, we adopted three state-of-the-art deep learning based encrypted traffic classification methods, MLP on our new augmented training dataset synthesized from our FlowGAN. The experimental results demonstrate that our proposed FlowGAN can achieve better performance compared with other traditional data augmenting methods like over sampling [7].

The rest of this paper is organized as follows. Section II introduces the preliminaries and related works of traffic classification, some current methods for tackling with the problem of imbalanced class data and GAN. Section III illustrates the algorithm of GAN and application of generating traffic samples. Section IV describes the methodology of FlowGAN. The experimental results are provided and discussed in Section V. Section VI concludes our work and presents some future works.

2. Related works

2.1. Deep Learning based Traffic Classification

Traffic classification has play a crucial role in the network management domain, especially QoS. In summary, there are three approaches to identify network traffic: port-based, payload-based and machine learning based. Unlike most traditional ML algorithms, Deep Learning performs automatic feature extraction without human intervention, which undoubtedly makes it a highly desirable approach for traffic classification, especially mobile services encrypted traffic. Recent research work has demonstrated the superiority of DL methods in traffic classification [4, 8–11]. The application of DL techniques involves three steps. First, model inputs are defined and designed according to some principles, such as packets, PCAP files, flow statistics vectors. Second, models and algorithms are deliberately chosen based on models' characteristics and aim of the classifier. Finally, the DL classifier is trained to automatically extract the features of traffic and associate the inputs with corresponding class labels.

2.2. Methods for handling imbalanced data

Generally, there are three approaches for tackling with imbalanced data: ***Modifying the objective cost function, Sampling and Generating artificial data*** [12]. The approach of ***modifying objective cost function*** alleviates the problem of class imbalance by means of weighting the data samples in minor and major classes differently, which gives higher score on the minor samples to penalize more intensely on miss-classifiying of the sample in the minor class. Sampling methods include two different ways of ***under-sampling*** and ***over-sampling***, which to reduce the size of major class by removing some major data samples and raising the samples in the minor class, respectively. Random under sampling (RUS) and Random over sampling (ROS) are two main methods of under-sampling and over-sampling [7]. RUS randomly removes some instances in major class, accordingly, ROS generates some copies of samples of the minor class. However, overfitting problem is always the main drawback of ROS due to generating same copies from the minor class. A classical method for generating artificial data is Synthetic Minority Over-sampling Techinique (SMOTE) in which minority samples are generated by synthetic samples rather than copies [13].

3. GAN

3.1. Overview of GAN

Generative adversarial network(GAN) has been considered as a promising technique since proposed by Goodfellow in 2014 [14], which is a framework to train the generative models. The main idea of GAN is that two networks, the generator network and discriminator network, play a minimax game in order to converge to an optimal solution. GAN has shown its state-of-the-art advance

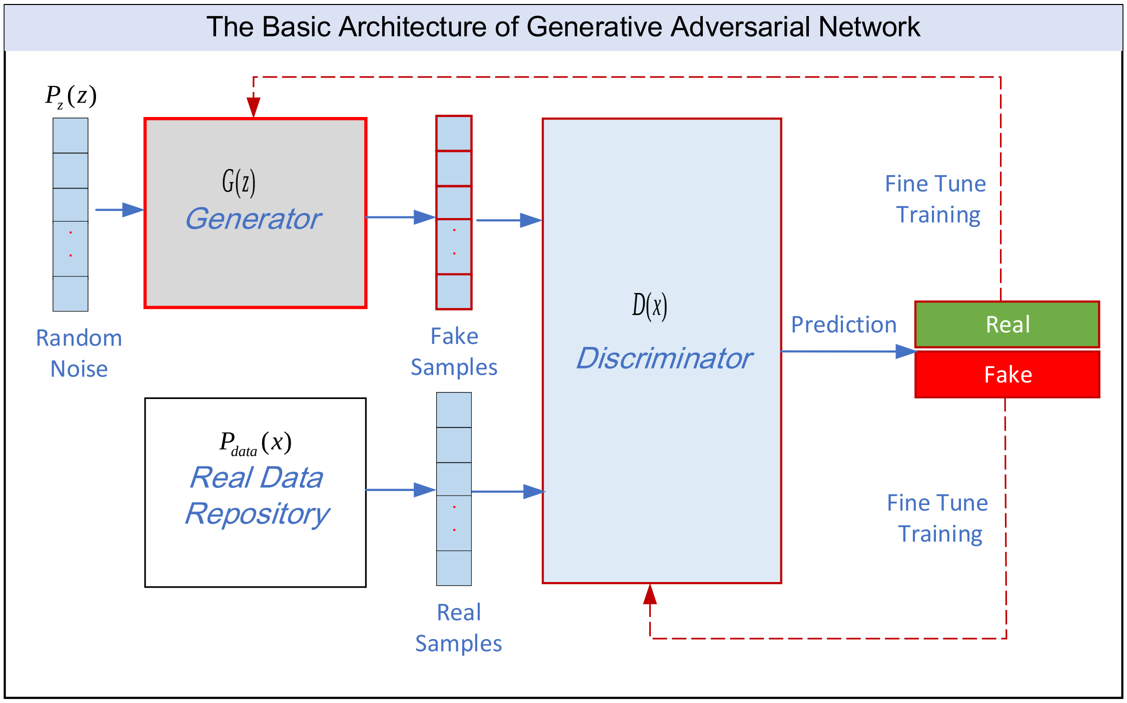


Figure 1: The basic architecture of GAN

in the generation of images, sound and texts. A basic architecture of GAN is shown in Fig. 1. To learn the generator's distribution $G(z)$, a prior on input noise variables $p_z(z)$ is defined. Meanwhile, discriminator's distribution $D(x)$ is defined, which represents the probability that x comes from the real data repository represented by $p_{data}(x)$ rather than $G(z)$. One can train D to maximize the probability of assigning the correct label to both training examples and samples from G by fine tuning, which is trained to minimize $log(1 - D(G(z)))$. In summary, D and G play the following two-player minimax game with value function $V(D, G)$, which is shown as following (1):

$$\min_{\mathrm{G}} \max_{D} V(D, G) = \mathbb{E}_{x \sim p_{data}(x)}[\log D(x)] + \mathbb{E}_{z \sim p_z(z)}[\log(1 - D(G(z)))] \tag{1}$$

The generator G takes random noise as input and generates fake samples. The discriminator D is fed with samples from both the generator and the real training data and attempt to distinguish between the two sources. These two networks play a competitive game, where the generator G is learning to generate more and more realistic samples, and the discriminator D is learning to get better and better at distinguishing the generated data from the real data. These two networks are trained simultaneously, and hope that the competition will drive the generated samples to be indistinguishable from the real data. The output of the generator G is a synthesized sample $X_{fake} = G(z)$. Discriminator network D takes the input of a real data sample or a synthesized sample from the generator and the output is a probability distribution $P(F|X) = D(X)$ over possible sources. Discriminator D is trained to maximize the log-likelihood to assigns the correct label as shown in equation (2) while Generator G is trained to minimize the second term in this equation.

$$L{=}E[\log P(F = real|X_{real})] + E[\log P(F = fake|X_{fake}) \tag{2}$$

3.2. The application of GAN in generating traffic data samples

Similar with texts or sentences, GAN can also be applied to the traffic data generation. Current researches have shown that GAN can improve the malware detection or IDS [15]. As for the application of GAN in the traffic classification, recent research work has proposed some ideas using GAN to generate the traffic samples to overcome the imbalaced property of network data. In [16], the authors adopted an unsupervised learning method called auxiliary classifier GANs(AC-GAN) to generate synthesized traffic samples for balancing between the minor and major classes over a well-known traffic dataset NIMS which only included SSH and non-SSH two classes. The AC-GAN took both a random noise and a class label as input in order to generate the samples of the input class lable accordingly. The experimental results have shown that their proposed method achieved better performance compared to other methods like SMOTE.

4. FlowGAN - GAN based traffic data generating method

4.1. The Framework of FlowGAN Based Encrypted Traffic Classification

This section will discuss the framework of FlowGAN based encrypted traffic classification shown in Fig. 2. The same as general architecture of deep learning based traffic classification, there are six steps for traffic classification, which is classification task definition, data preparation, data pre-processing, model input design, pre-training design, model architecture design. The detail has been introduced in our previous work [17]. Apparently, the only difference between FlowGAN based framework and general architecture is data augmentation in data preparation phase. Flow-

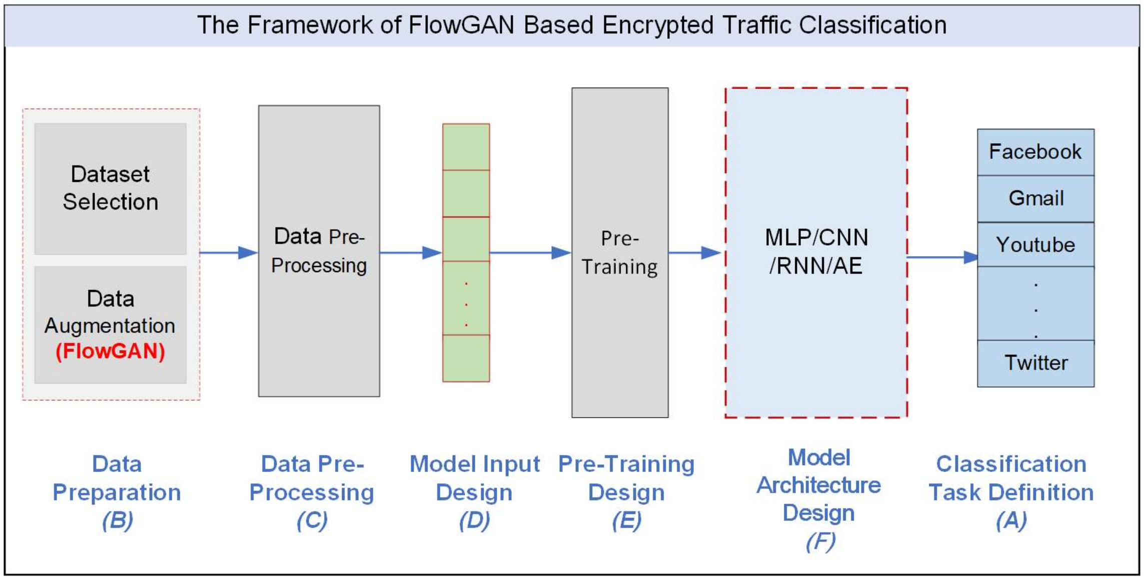


Figure 2: The framework of FlowGAN based encrypted traffic classification

GAN proposed by us is the method of data augmentation to alleviate the problem of imbalanced class data.

4.2. The Methodology of FlowGAN

The methodology of FlowGAN is shown in Fig. 3. There are three phases during the traffic data generating by FlowGAN, which are raw PCAP files pre-processing, GAN model training and data balancing. The detail of each phase will be illustrated as follows.

4.2.1. Raw PCAP Files Pre-processing

A raw data packet is always captured in PCAP or PCAPNG format, which has to be pre-processed for the input of subsequent GAN model training. In general, pcap files pre-processing has three steps, ***filtering***, ***truncating/zero padding*** and ***normalization***. An overview of the pre-processing procedure (viz. *phase 1*) is shown in Fig. 3. ***filtering*** is to remove the Ethernet header of a raw data packet. Data-link layer information such as MAC address, type of frame, etc., is not useful in packet classification. The ***filtering*** process reduces the input size of a packet. Moreover, some noise is filtered during the process for better performance. ***truncating and zero-padding*** is to fix the size of each data packet input to the GAN model. An equal size of all inputs is required for

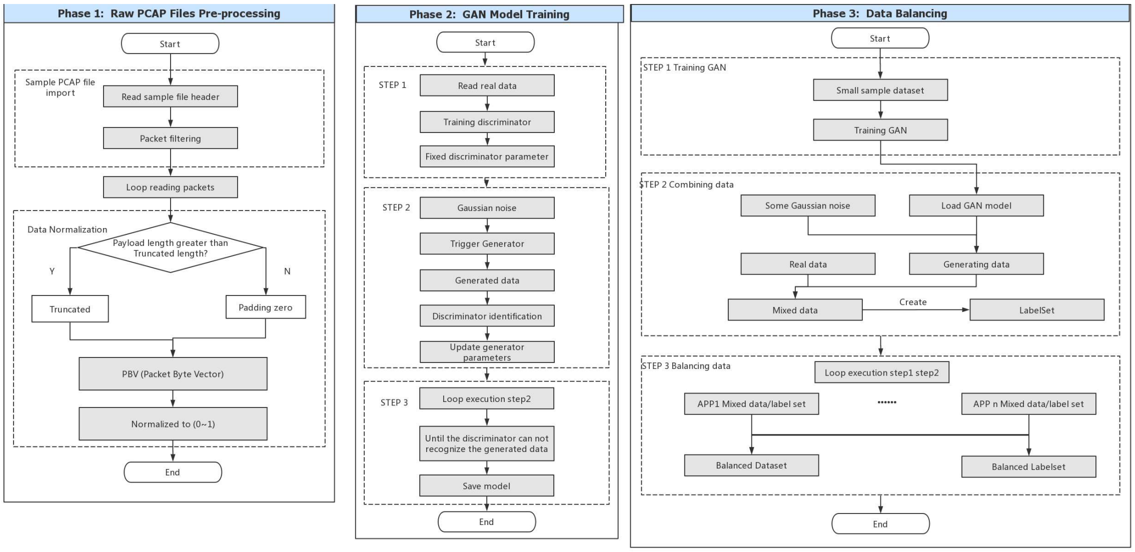


Figure 3: The methodology of FlowGAN

the proposed FlowGAN model. An input after truncating and zero-padding is defined as a Packet Byte Vector (PBV). For example, the i-th PBV is described as follows:

$$X_i = \{x_{i1}, x_{i2}, x_{i3}, ...x_{in}\} \tag{3}$$

where x_{ij} denotes the j-th byte X_i. Each PBV is then *normalized* to $[0, 1]$ for faster convergence. For simplicity, we assume X_i is the normalized result of the i-th PBV. Generation of a data packet is processed using the normalized PBV.

4.2.2. GAN Model Training

There are three steps during GAN model training. In step 1, the discriminator D is trained with the input of real traffic data from 4.2.1 till D is near convergence as shown in Algorithm. 1. And then step 2 starts while fixing the parameters of D, the generator G takes random Gaussian noise as input and generate samples. D receives traffic samples from both the generator and the real traffic data and attempt to distinguish between the two sources as shown in Algorithm. 2. These two networks play a competitive game, where the generator is learning to generate more and more realistic traffic samples, and the discriminator is learning to get better and better at distinguishing the generated data from the real data in step 3. These two networks are trained simultaneously, and hope that the competition will drive the generated samples to be indistinguishable from the real data.

To start the training process, training parameters are set as $\{N_e, M, \eta\}$, where N_e is the maximum number of Epoch, M is the size of mini_batch used in the stochastic gradient method, η is the learning rate. The complete process for the training process is summarized in Algorithm. 1 and 2. Without loss of generality, the algorithm only summarizes the basic structure of the process. Stopping criteria such as validation is not given in the description.

Algorithm 1 Discriminator of FlowGAN training

```
Input: real data from Section 4.2.1
Output: Discriminator network $D$ of FlowGAN.
for $t = 1$ to $N_e$ do
    for each batch of $M$ input data do
        For each training samples $X_i \in \mathbb{X}$:
        Compute the output using Equation. (2);
        Process with activation function;
        Output distinguishing results according to Equation. (2);
        Compute the training error;
        Update weights and bias;
    end for
end for
```

4.2.3. Data Balancing

The phase 3 of data balancing is the combination of the training of FlowGAN, mixing the synthesized/real samples and balancing for each class, especially minority class. In our paper, we reduce the size of majority class randomly and augment the size of minority class by FlowGAN.

5. Evaluation and Experimental Results

In this section, we present the experimental results to evaluate the accuracy of the proposed FlowGAN. Moreover, we have to evaluate the performance of FlowGAN by applying the dataset

Algorithm 2 Generator of FlowGAN training

Input: random Guassian noise, fixed discriminator's parameters

Output: Generator network G of FlowGAN.

```
for $t = 1$ to $N_e$ do
    for each batch of $M$ input data do
        For each random Guassian noise input:
        Generate the fake synthesized samples as output:
        Feed the output samples into discriminator $D$;
        Process with activation function;
        Output distinguishing results with the real data according to Equation. (2);
        Compute the training error;
        Update weights and bias;
    end for
end for
```

augmented by FlowGAN to the well-known deep learning based network traffic classifier (such as MLP) because there are still no more better quantified performance metrics for GAN.

5.1. Experiment Settings

5.1.1. Dataset for Evaluation

The dataset for evaluation is selected from the "ISCX VPN-nonVPN traffic dataset" [18]. As shown in Table 1, the total dataset for evaluation is composed of 15 applications, e.g., Facebook, Youtube, Netflix, etc. The chosen applications are encrypted with various security protocols, including HTTPS, SSL, SSH, and proprietary protocols. A total of $206,688$ data packets are included in the selected dataset. Apparently, there are some majority classes like Netflix, which accounts for 25.126% of the total dataset. Accordingly, there are some minority classes like AIM, Email-Client, Facebook, ICQ. To tackle with the imbalance class problem, we further create two datasets with more balanced data samples for each application, which are augmented by the methods of oversampling and FlowGAN respectively.

Table 1: Description of the chosen datasets.

Application	Security Protocol	Unbalanced dataset		Oversampling dataset		FlowGAN augmenting dataset	
		Quantity	Percentage	Quantity	Percentage	Quantity	Percentage
AIM	HTTPS	4869	2.356%	10000	6.67%	10067	6.706%
Email-Client	SSL	4417	2.137%	10000	6.67%	10015	6.671%
Facebook	HTTPS	5527	2.674%	10000	6.67%	10025	6.678%
Gmail	HTTPS	7329	3.546%	10000	6.67%	10007	6.666%
Hangout	HTTPS	7587	3.671%	10000	6.67%	10005	6.664%
ICQ	HTTPS	4243	2.053%	10000	6.67%	10001	6.662%
Netflix	HTTPS	51932	25.126%	10000	6.67%	9999	6.660%
SCP	SSH	15390	7.446%	10000	6.67%	9999	6.660%
SFTP	SSH	4729	2.287%	10000	6.67%	10007	6.666%
Skype	proprietary	4607	2.229%	10000	6.67%	10005	6.664%
Spotify	proprietary	14442	6.987%	10000	6.67%	9999	6.660%
torTwitter	proprietary	14654	7.089%	10000	6.67%	9999	6.660%
Vimeo	HTTPS	18755	9.074%	10000	6.67%	9999	6.660%
voipbuster	proprietary	35469	17.161%	10000	6.67%	9999	6.660%
Youtube	HTTPS	12738	6.163%	10000	6.67%	9999	6.660%
TOTAL		**206688**	**100%**	**150000**	**100%**	**150125**	**100%**

5.1.2. Configurations of the Computing Platform

The performance evaluations are conducted using a Dell R730 server with an Intel I7-7600U CPU 2.8 GHz, 8 GB RAM and an external GPU (Nvidia GeForce GTX 1050TI). The software platform for deep learning is built on Keras library with Tensorflow (GPU-based version 1.9.0) as the back-end support.

5.1.3. Description of deep learning based network traffic classifier

In this paper, we select a well-known deep learning model, Multi-perceptron (MLP) based network traffic classifier to evaluate the performance of our proposed FlowGAN. The MLP network traffic classifier consists of one input layer, two hidden layers and one output layer. Using the full size of the data packet as an example, the input layer has 1480 inputs. The two hidden layers are composed of 6 and 6 neurons respectively. The output layer is composed of 15 neurons with Softmax as classifier. The MLP model has been trained with Adam optimizer and Cross-entropy loss function.

5.1.4. Performance Metrics

The performance metrics used for evaluations of network traffic classifiers are *Precision*, *Recall* and F_1 *score*.

- ***Precision***: precision r_p is the ratio of *true positives* n_T^P over the sum of n_T^P and *false positives* n_F^P. In the proposed classification methods, precision is the percentage of packets that are properly attributed to the targeted application.

$$r_p = \frac{n_T^P}{n_T^P + n_F^P}. \tag{4}$$

- ***Recall***: recall r_c is the ratio of n_T^P over the sum of n_T^P and *false negatives* n_F^N or the percentage of packets in an application class that are correctly identified.

$$r_c = \frac{n_T^P}{n_T^P + n_F^N}. \tag{5}$$

- F_1***-score***: the F_1 score r_f is a widely-used metric in information retrieval and classification that considers both precision and recall as follows:

$$r_f = \frac{2r_p \cdot r_c}{r_p + r_c}. \tag{6}$$

5.2. The Performance of FlowGAN

5.2.1. FlowGAN Model Architecture

In our experiments, we employ MLP architecture to design Generator and Discriminator network as shown in Table. 2 and 3. The input of Generator network is a vector of 100 scalars generated from random Gaussian noise. The following subsequent 3 hidden layers have 128, 256, 512 neurons, respectively and the output layer has 1024 neurons. Accordingly, the input of Discriminator network is the vector of 1024 generated from either real traffic data or output of generator network. Three hidden layers and output layer are all with LeakyReLU as activation function. During the process of training for FlowGAN, we take Adam as optimizer and cross-entropy as loss function with 1000 epoches and 256 of mini_batch.

Table 2: Generator model description of FlowGAN.

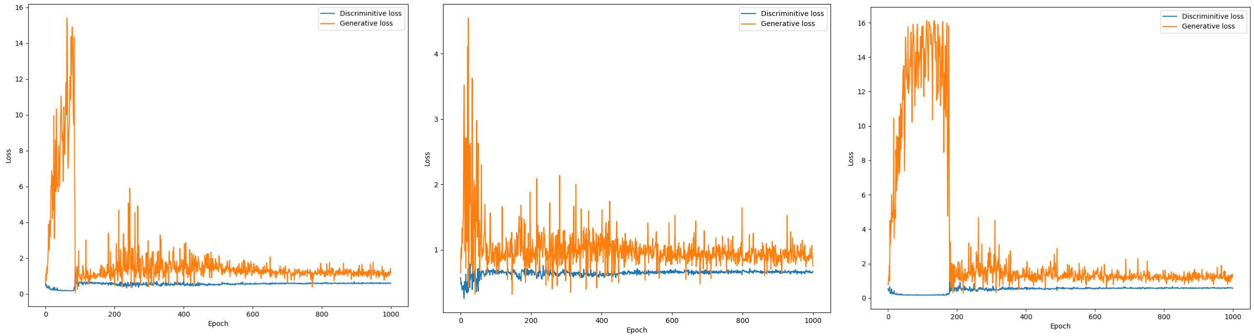
Input Layer		Hidden Layer1		Hidden Layer 2		Hidden Layer 3		Output Layer	
Input	**Activation**	**Output**	**Activation**	**Output**	**Activation**	**Output**	**Activation**	**Output**	**Activation**
100	LeakyReLU	128	LeakyReLU	256	LeakyReLU	512	LeakyReLU	1024	LeakyReLU

Table 3: Discriminator model description of FlowGAN.

Input Layer		Hidden Layer1		Hidden Layer 2		Hidden Layer 3		Output Layer	
Input	**Activation**	**Output**	**Activation**	**Output**	**Activation**	**Output**	**Activation**	**Output**	**Activation**
1024	LeakyReLU	512	LeakyReLU	256	LeakyReLU	128	LeakyReLU	1	Softmax

5.2.2. Loss of Generator and Discriminator Network of FlowGAN

As shown in Fig. 4, training loss of FlowGAN over three minority application classes are demonstrated, which are aim_chat, ICQ_chat, hangout_chat. Apparently, discriminator is fitted stably after 100 epoches, on the contrary, generator after 600 epoches.



(a) Loss of aim_chat.

(b) Loss of ICQ_chat.

(c) Loss of hangout_chat.

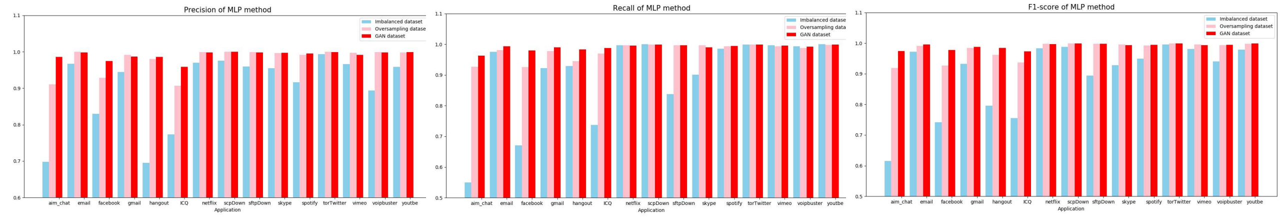
Figure 4: Loss of FlowGAN (three minority classes from 15 applications).

5.3. The performance of MLP-based Network Traffic Classifier

From the Table. 4, we can see that MLP classifier on FlowGAN dataset outperforms the other two methods about Accuracy, Precision, Recall and F1-Score. Furthermore, we can verify that FlowGAN can properly improve the problem of class imbalance and get better performance than oversampling method.

Table 4: Performance of MLP-based traffic classifier.

data augmenting methods	Accuracy	Precision	Recall	F1-Score
MLP classifier on unbalanced dataset	0.8995	0.9	0.8995	0.8968
MLP classifier on oversampling dataset	0.9794	0.9799	0.9794	0.9796
MLP classifier on FlowGAN dataset	0.991	0.9911	0.991	0.991



(a) Precision of MLP based classifier.

(b) Recall of MLP based classifier.

(c) F1-Score of MLP based classifier.

Figure 5: Performance comparison between three methods.

6. Conclusion and Future work

In this paper, we proposed the design and development of Generative Adversarial Network (GAN) based traffic data augmenting method called FlowGAN to generate synthesized samples for encrypted traffic classification. The synthesized data is then combined with the original (viz. real) data to construct the new traffic training dataset. As a proof of concept, we adopted three state-of-the-art deep learning based encrypted traffic classification methods, MLP on our new augmented training dataset synthesized from our FlowGAN. The experimental results demonstrate that our proposed FlowGAN can achieve better performance compared with other traditional data augmenting methods like over sampling. In the future, we will further study the other genre of GAN, like CGAN, WGAN for traffic classification to improve the performance of data augmentation methods.

Acknowledgement

The authors would like to thank the hardworking effort of my postgraduates for the experiments of FlowGAN.

Reference

[1] P. Wang, F. Ye, and X. Chen, "A smart home gateway platform for data collection and awareness," *IEEE Communications Magazine*, vol. 56, no. 9, pp. 87–93, Sep. 2018.

[2] P. Wang, X. Chen, F. Ye, and Z. Sun, "A smart automated signature extraction scheme for mobile phone number in human-centered smart home systems," *IEEE Access*, vol. 6, pp. 30 483–30 490, 2018.

[3] A. Dainotti, A. Pescape, and K. C. Claffy, "Issues and future directions in traffic classification," *IEEE Network*, vol. 26, no. 1, pp. 35–40, January 2012.

[4] G. Aceto, D. Ciuonzo, A. Montieri, and A. Pescap, "Mobile encrypted traffic classification using deep learning," in *2018 Network Traffic Measurement and Analysis Conference (TMA)*, June 2018, pp. 1–8.

[5] N. Japkowicz and S. Stephen, "The class imbalance problem: A systematic study," *Intell. Data Anal.*, vol. 6, no. 5, pp. 429–449, Oct. 2002. [Online]. Available: http://dl.acm.org/citation.cfm?id=1293951.1293954

[6] L. Vu, C. T. Bui, and Q. U. Nguyen, "A deep learning based method for handling imbalanced problem in network traffic classification," in *Proceedings of the Eighth International Symposium on Information and Communication Technology*, ser. SoICT 2017. New York, NY, USA: ACM, 2017, pp. 333–339. [Online]. Available: http://doi.acm.org/10.1145/3155133.3155175

[7] H. Guo and H. L. Viktor, "Learning from imbalanced data sets with boosting and data generation: The databoost-im approach," *SIGKDD Explor. Newsl.*, vol. 6, no. 1, pp. 30–39, Jun. 2004. [Online]. Available: http://doi.acm.org/10.1145/1007730.1007736

[8] P. Wang, F. Ye, X. Chen, and Y. Qian, "Datanet: Deep learning based encrypted network traffic classification in sdn home gateway," *IEEE Access*, vol. 6, pp. 55 380–55 391, 2018.

[9] W. Wang, M. Zhu, J. Wang, X. Zeng, and Z. Yang, "End-to-end encrypted traffic classification with one-dimensional convolution neural networks," *2017 IEEE International Conference on Intelligence and Security Informatics (ISI)*, pp. 43–48, 2017.

[10] M. Lopez-Martin, B. Carro, A. Sanchez-Esguevillas, and J. Lloret, "Network traffic classifier with convolutional and recurrent neural networks for internet of things," *IEEE Access*, vol. 5, pp. 18 042–18 050, 2017.

[11] X. Chen, J. Yu, F. Ye, and P. Wang, "A hierarchical approach to encrypted data packet classification in smart home gateways," in *2018 IEEE 16th Intl Conf on Dependable, Autonomic and Secure Computing, 16th Intl Conf on Pervasive Intelligence and Computing, 4th Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress(DASC/PiCom/DataCom/CyberSciTech)*, Aug 2018, pp. 41–45.

[12] L. Vu, D. Van Tra, and Q. U. Nguyen, “Learning from imbalanced data for encrypted traffic identification problem,” in *Proceedings of the Seventh Symposium on Information and Communication Technology*, ser. SoICT ’16. New York, NY, USA: ACM, 2016, pp. 147–152. [Online]. Available: http://doi.acm.org/10.1145/3011077.3011132

[13] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, “Smote: Synthetic minority over-sampling technique,” *J. Artif. Int. Res.*, vol. 16, no. 1, pp. 321–357, Jun. 2002. [Online]. Available: http://dl.acm.org/citation.cfm?id=1622407.1622416

[14] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, “Generative adversarial nets,” in *Advances in Neural Information Processing Systems 27*, Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2014, pp. 2672–2680. [Online]. Available: http://papers.nips.cc/paper/5423-generative-adversarial-nets.pdf

[15] W. Hu and Y. Tan, “Generating adversarial malware examples for black-box attacks based on gan,” *CoRR*, vol. abs/1702.05983, 2017.

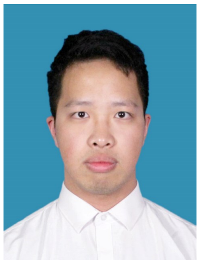
[16] L. Vu, C. Thanh Bui, and U. Nguyen, “A deep learning based method for handling imbalanced problem in network traffic classification,” 12 2017, pp. 333–339.

[17] P. Wang, X. Chen, F. Ye, and Z. Sun, “A survey of techniques for mobile service encrypted traffic classification using deep learning,” *IEEE Access*, vol. 7, pp. 54 024–54 033, 2019.

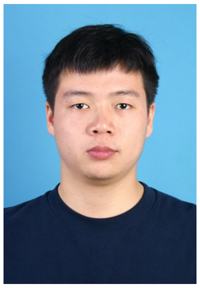
[18] A. Habibi Lashkari, G. Draper Gil, M. Mamun, and A. Ghorbani, “Characterization of encrypted and vpn traffic using time-related features,” 02 2016.

Biography

Pan Wang (M’18) received the BS degree from the Department of Communication Engineering, Nanjing University of Posts & Telecommunications, Nanjing, China, in 2001, and the PhD degree in Electrical & Computer Engineering from Nanjing University of Posts & Telecommunications, Nanjing, China, in 2013. He is currently an Associate Professor in the School of Modern Posts, Nanjing University of Posts & Telecommunications, Nanjing, China. His research interests include cyber security and communication network security, network measurements, Quality of Service, Deep Packet Inspection, SDN, big data analytics and applications. From 2017 to 2018, he was a visiting scholar of University of Dayton (UD) in the Department of Electrical and Computer Engineering.(email:wangpan@njupt.edu.cn)



ZiXuan Wang was born in Nanjing, Jiangsu, China ,in 1994 . He obtained a bachelor's degree from Tongda College of Nanjing University of Posts and Telecommunications in 2017, He is currently pursuing a master's degree in logistics engineering at Nanjing University of Posts and Telecommunications, under the direction of Professor Wang. His research interests include encrypted traffic identification and data balancing.(email:wangzx@runtrend.com.cn)



ShuHang Li was graduated from Jiangsu University of Science and Technology,Zhenjiang ,China, in 2013.He is currently pursuing a master's degree at Nanjing University of Posts & Telecommunications,Nanjing China.His research direction is encrypted traffic identification,and he also interested in Deep Packet Inspection and applications.(email:lish@runtrend.com.cn)

Chen Huang was born in Xi'an, Shaanxi, China, in 1995. She received the B.S. degree from School of Communication and Information Technology, Xi'an University of Posts & Telecommunications, in 2017. She is currently pursuing the master's degree in logistics engineering with the Nanjing University of Posts and Telecommunications, under the supervision of Prof. P. Wang. His research interests include data mining and recommendation systems.(email: ahchenen@163.com)

Feng Xiang received his B.A. degree from Nanjing University and Master of Public Administration from JFK School of Government, Harvard University. He is currently CEO of YTO Express Company Ltd and Director of National Engineering Laboratory for Logistics Information Technology. His research interests include logistics engineering and strategic management of business.(email: 13770610203@163.com)