IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

SPECIAL SECTION ON MOBILE SERVICE COMPUTING WITH INTERNET OF THINGS

Received April 8, 2019, accepted April 20, 2019, date of publication April 23, 2019, date of current version May 3, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2912896

A Survey of Techniques for Mobile Service Encrypted Traffic Classification Using Deep Learning

PAN WANG[1], (Member, IEEE), XUEJIAO CHEN[2], (Member, IEEE), FENG YE[3], (Member, IEEE), AND ZHIXIN SUN[1]

[1]Technology Research and Development Center of Postal Industry, Nanjing University of Posts and Telecommunications, Nanjing 210003, China
[2]School of Communication, Nanjing College of Information Technology, Nanjing 210023, China
[3]Department of Electrical and Computer Engineering, University of Dayton, Dayton, OH 45469, USA

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

This work was supported in part by the National Natural Science Foundation of China under Grant 61373135, Grant 61672299, Grant 61702281, and Grant 61602259, and in part by the Natural Science Foundation of Jiangsu Province under Grant BK20150866 and Grant BK20160913.

ABSTRACT The rapid adoption of mobile devices has dramatically changed the access to various networking services and led to the explosion of mobile service traffic. Mobile service traffic classification has been a crucial task that attracts strong interest in mobile network management and security as well as machine learning communities for past decades. However, with more and more adoptions of encryption over mobile services, it brings a lot of challenges about mobile traffic classification. Although classical machine learning approaches can solve many issues that port and payload-based methods cannot solve, it still has some limitations, such as time-consuming, costly handcrafted features, and frequent features update. With the excellent ability of automatic feature learning, Deep Learning (DL) undoubtedly becomes a highly desirable approach for mobile services traffic classification, especially encrypted traffic. This survey paper looks at emerging research into the application of DL methods to encrypted traffic classification of mobile services and presents a general framework of DL-based mobile encrypted traffic classification. Moreover, we review most of the recent existing work according to dataset selection, model input design, and model architecture. Furthermore, we propose some noteworthy issues and challenges about DL-based mobile services traffic classification.

INDEX TERMS Mobile services, encrypted traffic classification, traffic identification, deep Learning, CNN.

I. INTRODUCTION

Mobile services traffic identification and classification is an important research topic in the field of mobile network management and security. It is the premise and foundation of mobile network resource scheduling, content based billing, intrusion detection and other mobile network management and security monitoring tasks. Efficient, accurate and real-time mobile traffic classification is of great practical significance to provide mobile service quality assurance, dynamic access control and abnormal network behaviors detection. With the widespread adoption of encryption techniques in mobile services (including E-commerce, search engine, social networking, etc.), encrypted traffic has dramatically become a great challenge for mobile network management and security monitoring. Studies on mobile encrypted traffic classification not only help to improve the fine-grained mobile network resource allocation based on services or application, but also enhance security of mobile network and application.

The evolution of mobile encrypted traffic classification technology has gone through three stages: port-based, payload-based and flow-based statistical characteristics. Port-based classification method infers mobile services or 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

The associate editor coordinating the review of this manuscript and approving it for publication was Wenbing Zhao.

because it needs to match packet content and has high computational overhead [1]. As a result, in order to attempt to solve the problem of mobile services 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) [2]–[5]. In addition, some statistical models such as GMM (Gaussian Mixed Model) [6] and HMM (Hidden Markov Models) [7] are used to identify and classify encrypted traffic. Although classical machine learning approach can solve many issues that port and payload based methods cannot solve, it still has some limitations: (1) It is hard to obtain handcrafted flow features, which always depend on the domain experts' experience. Therefore, the features cannot be automatically extracted and selected, which leads to great uncertainty and confusion of classic machine learning methods when applying ML to mobile services traffic classification. (2) Flow characteristics are prone to be out of date rapidly and need to be continuously updated frequently. (3) How to combine a large readily-obtainable unlabeled dataset with a few costly labeled dataset for traffic classifier to reduce the need of labeled data is a very crucial research topic. (4) Class imbalance is a non-trivial problem for traffic classification tasks, however, current data augmentation methods can not accurately generate samples as close to original data distribution as possible.

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. 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.

In this paper, a general framework of DL-based mobile services traffic classification is proposed. Moreover, we further provide a thorough survey of the state-of-the-art approaches to traffic classification focusing on mobile services encrypted traffic classification based on deep learning and discuss some noteworthy issues and challenges about DL-based traffic classification.

The rest of this paper is organized as follows. Section II introduces the preliminaries of mobile services traffic classification. Section III proposes a general framework of DL-based mobile services encrypted traffic classification. Section IV discusses some noteworthy issues and challenges about DL-based traffic classification. Section V concludes our work.

II. PRELIMINARIES

A. MOBILE SERVICES ENCRYPTED TRAFFIC

With the repaid growth of E-commerce, search engine and SNS mobile applications, privacy has become more and more important not only for mobile internet users, but also service providers. Therefore, several important security protocols aiming at privacy preservation have been brought forward, such as SSH, PKI, SET and SSL etc. As the most popular encrypted tunnel, Virtual Private Network (VPN) has been always used in data transmission to keep the data security and availability. HTTPS as the protocol of HTTP over SSL is another security protocol widely used in web or mobile application, such as e-shopping, search engine and SNS etc. Encryption methods as all above mentioned make the traffic data more and more secure, consequently, it naturally becomes a big challenge for traffic classification, especially application-level.

B. TRAFFIC CLASSIFICATION (TC)

Traffic classification has been tasks of crucial importance in the network management domain, especially QoS. In summary, there are three approaches about Internet traffic classification.

1) PORT-BASED APPROACH

This approach is the oldest method for traffic classification, which uses the association of the ports in the TCP/UDP header with well-known TCP/UDP port numbers assigned by the IANA [8]. Apparently, it is very simple and fast, nevertheless, not all protocols can be classified by ports because of dynamic ports or tunnels and Network Address Port Translation (NAPT) [9], [10].

2) PAYLOAD-BASED APPROACH

This approach identifies applications by inspecting the packet headers or even payload. It is often called Deep Packet Inspection (DPI). This method generally provides high accuracy with low false negative rates. However, it has very high computational resources consumption and also is expensive to develop and maintain the packet signature library up to date [11]. Moreover, it is useless to encrypted traffic classification.

3) STATISTICAL APPROACH

Statistical classification methods use payload-independant parameters such as packet length, inter-arrival time and flow duration to circumvent the problem of payload encrypted and user's privacy [12]. Many work was carried out using Machine Learning (ML) algorithms. In general, there are two learning strategies are used: one is the supervised methods like decision tree, SVM and Naive Bayes, the other is unsupervised approaches like k-means and PCA [13]. Nevertheless, its poor accuracy and handcrafted feature selection still cannot meet the fine grained traffic classification requirements of rapid growth of Smart Home.

C. DEEP LEARNING (DL)

1) INTRODUCTION OF ARTIFICIAL NEURAL NETWORK (ANN) AND DEEP LEARNING

ANN is a network of simple elements called neurons, which receive input, change their internal state activation according to that input, and produce output depending on the input and activation. Neural network models can be viewed as simple mathematical defining a function $f : X \longrightarrow Y$. Besides, a neuron's network function $f(x)$ is defined as a composition of other functions $g_i(x)$, which is widely used as the nonlinear weighted sum, that is

$$f(x) = K\left(\sum_{i=1}^{n} \omega_i g_i(x)\right) \tag{1}$$

where K is an activation function, such as *sigmoid* function or *softmax* function or *rectifier* function. The important characteristic of the activation function is that it provides a smooth transition as input values change.

Deep learning, also known as deep structured learning or hierarchical learning, is part of a broader family of machine learning methods based on learning data representations, as opposed to task-specific algorithms. Learning can be supervised, semi-supervised or unsupervised [14], [15].

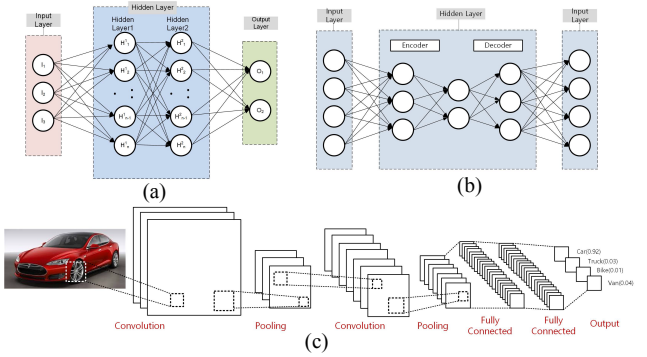


FIGURE 1. Some common neural network of deep learning. (a) FC (fully-connected neural network). (b) AE (autoencoder). (c) CNN (convolutional neural network).

2) MULTILAYER PERCEPTRON (MLP)

A multilayer perceptron (MLP) is a class of feedforward artificial neural network as shown in Fig. 1(a). A MLP consists of at least three layers of nodes. Except for the input nodes, each node is a neuron that uses a nonlinear activation function. MLP utilizes a supervised learning technique called backpropagation for training [16]. Since MLPs are fully connected, each node in one layer connects with a certain weight ω_{ij} to every node in the following layer. For example of supervised learning, the node weights are adjusted based on backpropagation that minimize the error in the entire output, given by

$$\varepsilon(n) = \frac{1}{2}\sum_{j} e_j^2(n) \tag{2}$$

Using gradient descent, the change in each weight is

$$\Delta\,\omega_{ji}(n) = -\,\eta\,\frac{\partial\varepsilon(n)}{\partial\,v_j(n)}\,y_i(n) \tag{3}$$

where y_i is the output of the previous neuron and η is the learning rate.

3) STACKED AUTOENCODER (SAE)

An autoencoder is an unsupervised learning algorithm which is one approach to automatically learn features from unlabeled data as shown in Fig. 1(b). It is usually used for dimensionality reduction or feature extraction. The network mainly consists an encoder function and a decoder function, given by $\phi : X \longrightarrow F$ and $\psi : F \longrightarrow X$. Autoencoders are trained to minimize reconstruction errors like following:

$$\phi,\ \psi = argmin \parallel X - (\phi \circ \psi)\,X \parallel^2 \tag{4}$$

Autoencoders can be thought as a special case of feedforward networks, and it can be trained with the same techniques, such as backpropagation algorithm. In order to obtain a better performance, SAE stacks several autoencoders, in which the output of one autoencoder is the input of the next autoencoder [17].

4) CONVOLUTIONAL NEURAL NETWORK (CNN)

CNN is another type of deep learning network as shown in Fig. 1(c). For example, each neuron in the lth layer is connected to a small region of the neurons in the $(l-1)$th layer. The local filters are used to complete the mapping which can be viewed as convolutional functions. In addition, the replicated units share the same weight vectors and bias. It increases learning efficiency by reducing the number of parameters being learnt greatly. Another important operation of a CNN is pooling which is used for down-sampling. For example, the max pooling outputs the maximum value within a rectangular sub-region. Other popular pooling functions include the average pooling, the L2 norm pooling. The pooling provides a form of translation invariance and reduces computation for upper layers [18].

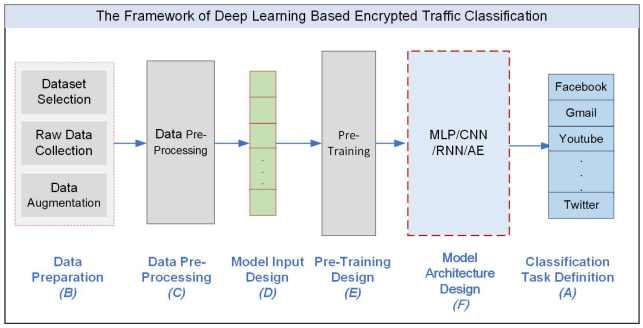


FIGURE 2. The framework of deep learning based mobile services encrypted traffic classification.

III. THE FRAMEWORK OF DEEP LEARNING IN MOBILE SERVICES ENCRYPTED TRAFFIC CLASSIFICATION

This section will outline the framework of DL in mobile services encrypted traffic classification and present a thorough survey of existing relevant research work. Fig. 2 illustrates a general framework for DL-based mobile services

TABLE 1. Summary of deep learning based traffic classification methods.

Work	DL Algorithms	Model Input	Dataset	Classification Task	Granularity	Year
Z.Wang [19]	MLP,SAE	raw packet	private	traffic classification	application	2015
Datanet [20]	MLP,SAE,CNN	raw packet	ISCX2012	encrypted traffic classification	application	2018
DeepPacket [21]	MLP,SAE,CNN	raw packet	ISCX2012	encrypted traffic classification	application	2018
Wang-2D-CNN [22]	CNN	raw data	USTC-TFC2016	malware classification	application	2017
Wang-1D-CNN [23]	CNN	raw data	ISCX2012		encrypted traffic classification	2017
Lopez-Martin [24]	CNN,LSTM	packet-level features	private dataset	traffic classification	application	2017
Aceto [25]	MLP,SAE,CNN,LSTM	raw data	private dataset	mobile traffic classification	application	2018
HAST-IDS [26]	CNN,LSTM	raw data	ISCX2012	intrusion detection	application	2018
Rezaei [27]	CNN	packet-level features	private dataset	QUIC encrypted traffic classification	application	2018
Van [28]	CNN,RF	packet-level features	private dataset	QUIC encrypted traffic classification	application	2018
Y.Wang [29]	CNN	flow features	Moore	traffic classification	services group	2017
Seq2Img [30]	CNN	raw data	private dataset	traffic classification	protocol/application	2017
Li [31]	MLP,VAE	raw data	IMTD17	mobile traffic classification	application	2017
Vu [32]	GAN	raw data	NIMS	encrypted traffic classification	protocol	2017

traffic classification, which is composed of six steps. We will discuss these steps in this section and examine the corresponding parts related to the existing work. Furthermore, we demonstrate some potential challenges of these steps and share our ideas and solutions to these problems. The summary of recent research work is shown in Table 1.

A. CLASSIFICATION TASK DEFINITION

Defining explicit classification task is a primary step before designing traffic classifier. Substantially, classification task is composed of three parts, which are aims, granularity and performance requirements.

1) AIMS OF CLASSIFICATION

Generally, mobile services traffic classification aims always include three parts, which are mobile network management, security and personalized recommendation. There are Mobile Network Resource Scheduling, QoS Provisioning and Content based Billing in network management related scenarios. Intrusion Detection, Malware Detection and BotNet Detection are the typical scenarios of the network security. Moreover, Mobile Service Providers or Content Providers can push their own recommendations based on subscribers' preferences by means of fine-grained traffic classification, such as mobile internet users behavior analysis.

2) GRANULARITY OF CLASSIFICATION

To meet the different requirements of aforementioned classification aims, it is strongly necessary to define the granularity of traffic classification tasks based on following items:

1) Binary Classification (e.g. Normal or Anomaly, clear or encrypted, VPN or non-VPN), mainly used for Intrusion Detection, Malware Detection and Botnet Detection.
2) Protocols (e.g. TCP, UDP, HTTP or SMTP), mainly used for network resource scheduling, planning and allocation.
3) Services Group (e.g. Streaming, Browsing or Downloading), the same with aboved.
4) Applications (e.g. Facebook, Youtube or Skype), the same with aboved.
5) Websites (e.g. Search Engine, e-Shopping or social network websites) mainly used for recommendation based on internet user preference analysis.
6) User Specific Behaviors in Applications (e.g. Adding items to shopping cart of Amazon.com, Posting a picture on Twitter or voice call in Skype), the same with aboved.
7) Smart Devices (e.g. iPhone, iPad, TV Box), ISP can provide specific QoS based on different Smart Devices [33].
8) Application Identity (e.g. Mobile Phone Number [34], Facebook Account Name, Twitter User Name), mainly used for security audit and information forensics.

3) PERFORMANCE REQUIREMENTS OF CLASSIFICATION

As for performance requirements of classification, it is important to take two factors into account, one is realtime ability,

the other is light-weight ability. From the perspective of realtime ability of classification, one can divide the classifiers into two types: online and offline. Online classifiers always are used in realtime scenarios, such as Network Resource Scheduling, Intrusion Detection. In contrast, offline classifiers usually are used for user behaviors analysis, billing based on applications or content etc. Furthermore, some classifiers should be light-weight in some specific scenarios, especially in some simple hardware, such as home gateway or edge router [35]. Apparently, with the rapid growth of fog computing, light-weight classifiers arise more and more focus not only academic research but also network operating [36].

TABLE 2. The summary of dataset used in existing work.

Work	Dataset	Samples	Numbers of App	Encrypted	Balanced
Z.Wang [19]	private	300K	58	Yes	N/A
Datanet [20]	ISCX2012	73K	15	Yes	Yes
DeepPacket [21]	ISCX2012	22293K	17	Yes	Yes
Wang-2D-CNN [22]	USTC-TFC2016	752k	20	Yes	N/A
Wang-1D-CNN [23]	ISCX2012	N/A	12	Yes	N/A
Lopez-Martin [24]	private	266k	15	Yes	No
Aceto [25]	private	77.5K	49	Yes	N/A
HAST-IDS [26]	ISCX2012	1525K	5	Yes	No
Y.Wang [29]	Moore	377K	12	No	No
Seq2Img [30]	private	22K-pro/282K-app	5-pro/5-app	Yes	Yes
Li [31]	IMTD17	61K	12	Yes	N/A
Van [28]	private	20K	5	Yes	Yes

B. DATA PREPARATION

It is essential for training a deep learning model to acquire a large, balanced and representative dataset. There are three ways for data preparation, including selecting existing dataset, collecting raw data and generating synthesized samples. The description in detail is as follows:

1) ***Dataset Selection***. In Table 2, there are a summary of datasets used in recent existing work. Apparently, most work selected public datasets like ISCX2012 and Moore. Moreover, some work collected raw data from ISP's network or research lab to create their own dataset like USTC-TFC2016 and IMTD17. From the perspective of numbers of samples, we can see that most work selected 70K-1500K records for training, in which most work included encrypted traffic samples. While most work selected 5-17 applications or protocols as their classification task. In addition, it is noteworthy that some datasets used in existing work were imbalanced. Nevertheless, as we all know, class imbalance problem is non-trivial. In summary, there is no publicly accepted dataset for research because of following reasons:(1) no dataset can contain all types of application traffic because of huge numbers of traffic types and frequent update of applications. (2) It is very hard, time-consuming and costly to cover all the network scenarios like broadband and radio access, PC and mobile devices access etc. In one word, it is very difficult to create a dataset that is able to characterize the traffic data distribution accurately and extensively.
2) ***Raw Data Collection***. In Table 2, Wang [19], Lopez-Martin [24] and Seq2Img [30] collected raw data from internal company or ISP's network. They collected raw packets by some packet capture tools like Tcpdump [37], besides, some work use flow tools to collect traffic records like NetFlow [38].
3) ***Data Augmentation***. As we mentioned aboved, class imbalance is a non-trivial problem when facing to the traffic classification. As a useful method of handling class balance, data augmentation usually refers to generate synthesized samples to keep the samples of major and minor classes balanced. We will illustrate the topic in following Section IV-A later.

C. DATA PRE-PROCESSING

Generally speaking, traffic data in dataset can be categorized into three types: raw packet data, PCAP files and statistical features. The first two types of data usually need to be pre-processed because of three reasons:(1) raw packet data always contains some irrelevant packets, such as ARP, DHCP, ICMP. (2) packet-level feature distribution may be distorted by some unexpected network conditions, such as retransmission packets, out of order packets. (3) PCAP files contain some unnecessary information like PCAP files header. Therefore, some data pre-processed measures like packets filtering, header removal are needed [20], [21], [23]. Zero-padding and truncation at a fixed length are required in most cases of raw packet dataset, because Deep Neural Network (DNN) is always fed a fixed-size input, while the frame length of packets from dataset varies a lot from 54 to 1514 bytes when taking TCP as an example. In addition, data normalization is crucial to the performance of deep leanring, which always normalize the traffic data from dataset to a value in the range of $[-1,+1]$ or $[0,1]$. This facilitates the classification task to converge faster during the training of model.

D. MODEL INPUT DESIGN

As an important component, the input of a deep learning model has a strong impact on the performance of the model during training and testing. Generally speaking, the input of a traffic classification model based on DL can be categorized into three types: raw packet data, traffic features and combination with raw data and features. The summary of model input of the existing work is shown in Table 3.

1) ***Raw Packet Data***. Most work chose raw packets data as the input of model, such as [**?**], [19]–[22], [26]. As aforementioned in Section III-C, zero-padding and truncation are always needed. From Table 3, we can see that the range of zero-padding and truncation length is usually from 700 to 1500 bytes.

TABLE 3. The summary of model input of existing work.

Work	Data Type	Zero-Padding Len	Features	First Packets
Z.Wang [19]	raw data	1000	–	–
Datanet [20]	raw data	1480	–	–
DeepPacket [21]	raw data	1500	–	–
Wang-2D-CNN [22]	raw data	784	–	–
Wang-1D-CNN [23]	raw data	784	–	–
Lopez-Martin [24]	features	–	port, payload bytes, TCP window size, interarrival time, direction	20
HAST-IDS [26]	raw data	600	–	–
Y.Wang [29]	features	–	249 features from Moore dataset	–
Seq2Img [30]	features	–	raw features-28, statistical features-10	10
Rezaei [27]	features	–	interarrival time, packet length, direction	–
Van [28]	raw data and features	1400	percentage of small, medium, large packets, average payload length	10

2) ***Traffic Features***. Generally speaking, traffic features can be categorized into three types: packet-level features(such as packet length and inter-arrival time of packets), flow-level features(such as flow duration, total packet in the flow) and statistical features(such as average packet length and average bytes sent or received per second). In [24], packet-level features were widely used, such as source and destination port, payload bytes, TCP window size, inter-arrival time of packets and packet direction. In [27], the authors only selected three packet-level features like packet length, inter-arrival time of packets and direction of packet. While in [29], [30], packet-level, flow-level features and statistical features were all used for input of model. In addition, how many first packets of a flow to capture for extracting features has very strong impact on performance of traffic classifier, especially real-time classification. Apparently, the more first packets are collected, the more flow features are intact and comprehensive. Meanwhile, the complexity and computational load are increasing accordingly which will lead to the classifier's performance degradation. From Table 3, we can see some literatures have shown that they took first 10 [28], [30] or 20 [24] packets for flow features.

3) ***Combination with raw data and features***. In [28], they chose the combination of raw packet data and features extracted from netflow to identify the applications of Google using QUIC, which is a new encryption protocol.

E. PRE-TRAINING DESIGN

As we all know, deep learning requires a large amount of labeled data during training, however, collecting and labeling a large dataset is very time-consuming and costly. Traffic dataset is without exception, especially encrypted traffic, because current traffic labeling tools like DPI cannot handle encrypted traffic. On the contrary, unlabeled traffic data is abundant and readily available. Therefore, some researchers began to explore how to use easily-obtainable unlabeled traffic data combined with a few labeled traffic data for accurate traffic classification [27], [31]. Actually, this is a typical semi-supervised learning, by which one can pre-train a model D_u with a large unlabeled traffic data and then transfer it to a new architecture and retrain the model with D_l.

Furthermore, pre-training can also be used for dimension reduction by which model will become light-weight that is very important in some scenarios, such as online or simple hardware as aforementioned in Section III-A.3. Besides, large dataset will consume enormous computing and memory resources. In [20], [21], they designed a SAE model used in the pre-training process for dimension reduction, while VAE was used in [27], [31] for semi-supervised learning to overcome the labeling problem. A summary of pre-training desigh of existing work is shown in Table. 4.

TABLE 4. The summary of pre-training design of existing work.

Work	Purpose	Model
Datanet [20]	dimension reduction	SAE
DeepPacket [21]	dimension reduction	SAE
Rezaei [27]	semi-supervised	CNN
Li [31]	semi-supervised	VAE

F. MODEL ARCHITECTURE DESIGN

Model architecture is the most critical factor for traffic classification. In this subsection, we will present a thorough review of model architecture design of existing work.

1) MLP

In 2015, A first attempt using DL for traffic classification is proposed by Wang in [19]. MLP model (ANN called in this paper) was used for traffic records about 0.3 million, which were collected from their internal network. Besides, there were 58 protocol types in these private dataset including regular and encrypted applications. The experimental results show that both precision and recall can achieve more than 90% on the top 25 popular protocols. In 2018, P. Wang *et al.* introduced a DL based encrypted traffic classification method called DataNet, which was embeded in SDN Home Gateway for fine-grained network resource allocation [20]. MLP model was one of the TC methods of DataNet. The MLP model was composed of one input layer, two hidden layers and one output layer. The input layer has 1480 neurons and the two hidden layers were composed of 6 and 6 neurons respectively. The output layer was composed of 15 neurons with ***Softmax*** as classifier. They used VPN-nonVPN traffic dataset of ISCX2012 [39], which is composed of 15 encrypted applications and 73,392 packets.

The experimental evaluation results show that ***Precision, Recall and F1-Score*** were all more than 92%.

2) CNN

MLP cannot handle high dimensional input because the number of model parameters in hidden layers is too large, however, CNN improves this limitation by adopting convolution layers which use a set of kernels with a number of learned parameters. After convolution and pooling, the number of model parameters are notably reduced. Finally, classification task can be achieved by combining CNN with several fully-connected layers and softmax layer.

There are a few literatures focusing on traffic classification based on CNN. Referring to the CNN's successful applications in images and computer vision, most of existing work applied existing classical CNN models (VGG and ResNet) for traffic classification by converting packets sequences into images. In [22], Wang *et al.* has used 2D-CNN for malware traffic classification with all packet bytes of bidirectional flow and acquired an outstanding accuracy of classification. Whereafter, they proposed a 1D-CNN architecture to classify the traffic from ISCX2012 dataset and showed a significant improvement over C4.5 ML methods and a slight increasing over 2D-CNN they proposed previously [23]. Wang *et al.* proposed a LeNet-5 based CNN model for classification over Morre dataset after preprocessing traffic data using Min-Max Normalization method [29]. Chen *et al.* [30] presented a CNN model used Reproducing Kernel Hilbert Space (RKHS) embedding and converted the early time series data into 2D images and outperformed some classical ML methods.

3) RNN

As aforementioned aboved, CNN has very excellent learning ability of spatial characteristic, however, it cannot extract temporal features like time series data which are also strongly helpful to classification. In [26], the authors proposed a novel IDS based on a combination with CNN and LSTM, in which the low-level spatial features of network traffic were learned by CNN and high-level temporal features by LSTM. The automatically learned traffic features effectively reduced the False Alarm Rate(FAR). In [24], Lopez-Martin *et al.* proposed a time-series flow features based classification method using the combination with CNN and LSTM and this particular combination gave the best results compared with other alternative approaches. Different to aboved work, [40] proposed a novel data augmentation approach based on LSTM for generating traffic flow patterns to improve the class imbalance problem.

4) AE

As described in Section II-C.3, auto-encoder (AE) is an unsupervised learning algorithm which is an approach to automatically learn features from unlabeled data and usually used for dimensionality reduction or feature extraction. AE is widely used to initialize the parameters of DNN. As a basic architecture, AE has some other variations, such as Stacked Auto-encoder (SAE), Variational Auto-encoder (VAE) and Denoizing Auto-encoder (DAE). The existing work focusing on AE and the variations of AE are as follows:

1) ***AE***. In [41], an AE model was used to reconstruct the input traffic data and a sofmax classifier was combined with the encoded internal representation of the AE. The experimental evaluation has shown a moderate accuracy. In addition, AE is extensively used in IDS and network anormaly detection [42], [43].
2) ***SAE***. SAE is a stacked AEs architecture trained by a greedy layer-wise style which is a way that the output of each AE layer is the input of next AE layer. In [19]–[21], SAE models proposed were all used for traffic classification combined with Softmax and all experimental results have shown a very outstanding performance compared to classical ML algorithms. Moreover, the node numbers of the latent layer described in Datanet of [20] and Deeppacket of [21] were 32 and 50 respectively. Apparently, SAE model can effectively reduce the dimension of input data and reconstruct the input accurately.
3) ***VAE***. Compared to other AEs, the latent features learned by VAE conform to a probability distribution rather than a specific value. Therefore, it can reduce the rigid constraint of the parameters and improve the capacity to tolerate the error of complex input data [44]. In [31], the authors identified traffic through a two-stage learning, which included unsupervised feature extraction and supervised category mapping. In the first stage, VAE extracted latent features from massive unlabeled samples and mapped the features to certain categories with small-scale labeled samples. The experimental results have shown a very good performance.
4) ***DAE***. Although AEs have a lot of advantages about feature selection and extraction, there is still a serious problem that when there are more nodes in the hidden layer than input layer, AEs are risking to learn the so-called 'Identity Function', also called 'Null Function', meaning that the output equals the input, making the AEs useless. DAE solves this problem by corrupting the data on purpose by randomly turning some of the input values to zero. When calculating the loss function, it is important to compare the output values with the original input, not with the corrupted input. In this way, the risk of learning the identity function instead of extracting features is eliminated. Regrettably, there are little literatures about traffic classification using DAE, while some work used DAE for IDS and Network Anormaly Detection [45], [46].

5) GAN

Generative adversarial network(GAN) has been considered as a promising technique since proposed by Goodfellow *et al.* in 2014 [47], 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 [48]. GAN has shown its state-of-the-art advance in the generation of images, sound and texts [49]–[51]. 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 [52]–[56].

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 imbalanced property of network data. In [32], 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 label accordingly. The experimental results has shown that their proposed method achieved better performance compared to other methods like SMOTE. In addition, GAN has been applied in IDS and Malware detection to generate adversarial attacks to deceive and evade the detection systems [52], [54].

IV. NOTEWORTHY PROBLEMS OF DEEP LEARNING IN ENCRYPTED TRAFFIC CLASSIFICATION

A. CLASS IMBALANCE

As aforementioned in Section III-B, class imbalance is a non-trivial problem of traffic classification. This section will outline some techniques for addressing imbalanced data related traffic classification. According to the description in the literature [57] and [58], there are usually three methods: modifying the objective cost function, under-sampling and over-sampling, and generating artificial data to handle the imbalance problem. In table 2, DeepPacket [21] and Datanet [20] adopted under-sampling method to randomly remove the major classes' samples until the classes were fairly balanced, whereas, Lopez-Martin *et al.* [24] did not take the imbalanced problem into account. As a method of generating artificial data, SMOTE has been adopted in some work like [59]. With the rapid growth of deep generative model, recent work has shown some new ideas of data augmentation about traffic data. In [40], the authors proposed a data augmentation approach based on the use of LSTM for generating traffic flow patterns and Kernel Density Estimation(KDE) for replicating the numerical features of each class. In [60], an augmentation method has been proposed by using an Auxiliary Classifier Generative Adversarial Network (AC-GAN) to generate two classes of network, which were SSH and none-SSH.

B. SEMI-SUPERVISED LEARING BASED TRAFFIC CLASSIFICATION

Most existing work mentioned aboved is based on supervised learning way, which depends on large quantities of labeled data. As we all know, labeled data are always difficult and costly to obtain, however, unlabeled data are readily-obtainable. Obviously, it is a promising research direction that how to combine the large unlabeled traffic data with a few labeled ones to complete the classification task in a semi-supervised way, which will dramatically obviate the dependence of large labeled datasets. Generally speaking, a semi-supervised approach is composed of two stages: one is the stage of pre-training on large unlabeled traffic datasets in unsupervised way, the other is the stage of re-training on a small labeled traffic datasets in supervised way. In [27], the authors first pre-trained a model on a large unlabeled dataset where the input is the time series features of a few sampled packets, then transferred the learned weights to a new model to further re-train on a small labeled dataset generated from the more challenging QUIC protocol. The experimental results showed that the proposed semi-supervised approach achieved almost the same accuracy as a fully-supervised method.

V. CONCLUSION

In the paper, we proposed a general framework of DL based mobile services encrypted traffic classification and review most recent existing work according to classification task definition, data preparation, pre-processing, model input design, pre-training design and model architecture. Furthermore, some noteworthy issues are also illustrated and discussed. In the future work, we will continue to study and follow related progress about mobile services traffic classification.

REFERENCES

[1] M. Finsterbusch, C. Richter, E. Rocha, J.-A. Muller, and K. Hanssgen, "A survey of payload-based traffic classification approaches," *IEEE Commun. Surveys Tuts.*, vol. 16, no. 2, pp. 1135–1156, 2nd Quart., 2014.

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

[3] G.-L. Sun, Y. Xue, Y. Dong, D. Wang, and C. Li, "An novel hybrid method for effectively classifying encrypted traffic," in *Proc. Global Commun. Conf.*, Dec. 2010, pp. 1–5.

[4] P. Velan, M. Čermák, P. Čeleda, and M. Drašar, "A survey of methods for encrypted traffic classification and analysis," *Int. J. Netw. Manage.*, vol. 25, no. 5, pp. 355–374, Sep. 2015. doi: 10.1002/nem.1901.

[5] D. J. Arndt and A. N. Zincir-Heywood, "A comparison of three machine learning techniques for encrypted network traffic analysis," in *Proc. IEEE Symp. Comput. Intell. Secur. Defense Appl. (CISDA)*, Apr. 2011, pp. 107–114.

[6] M. Shen, M. Wei, L. Zhu, and M. Wang, "Classification of encrypted traffic with second-order Markov chains and application attribute bigrams," *IEEE Trans. Inf. Forensics Security*, vol. 12, no. 8, pp. 1830–1843, Aug. 2017.

[7] C. V. Wright, F. Monrose, and G. M. Masson, "On inferring application protocol behaviors in encrypted network traffic," *J. Mach. Learn. Res.*, vol. 7, pp. 2745–2769, Dec. 2006.

[8] *Service Name and Transport Protocol Port Number Registry*. [Online]. Available: https://www.iana.org/assignments/service-names-port-numbers

[9] A. W. Moore and K. Papagiannaki, "Toward the accurate identification of network applications," in *Passive and Active Network Measurement*, C. Dovrolis, Ed. Berlin, Germany: Springer, 2005, pp. 41–54.

[10] A. Madhukar and C. Williamson, "A longitudinal study of P2P traffic classification," in *Proc. 14th IEEE Int. Symp. Modeling Anal., Simulation*, Sep. 2006, pp. 179–188.

[11] M. Finsterbusch, C. Richter, E. Rocha, J. A. Muller, and K. Hanssgen, "A survey of payload-based traffic classification approaches," *IEEE Commun. Surveys Tuts.*, vol. 16, no. 2, pp. 1135–1156, 2nd Quart., 2014.

[12] D. C. Sicker, P. Ohm, and D. Grunwald, "Legal issues surrounding monitoring during network research," in *Proc. 7th ACM SIGCOMM Conf. Internet Meas. (IMC)*, New York, NY, USA, 2007, pp. 141–148. doi: 10.1145/1298306.1298307.

[13] T. T. T. Nguyen and G. Armitage, "A survey of techniques for Internet traffic classification using machine learning," *IEEE Commun. Surveys Tuts.*, vol. 10, no. 4, pp. 56–76, 4th Quart., 2008.

[14] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1798–1828, Jun. 2012.

[15] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural Netw.*, vol. 61, pp. 85–117, Jan. 2014.

[16] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning internal representations by error propagation," in *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, vol. 1, D. E. Rumelhart, J. L. McClelland, and C. PDP Research Group, Eds. Cambridge, MA, USA: MIT Press, 1986, pp. 318–362. [Online]. Available: http://dl.acm.org/citation.cfm?id=104279.104293

[17] S. Yadav and S. Subramanian, "Detection of application layer DDoS attack by feature learning using stacked autoencoder," in *Proc. Int. Conf. Comput. Techn. Inf. Commun. Technol. (ICCTICT)*, Mar. 2016, pp. 361–366.

[18] V. Sze, Y.-H. Chen, T.-J. Yang, and J. S. Emer, "Efficient processing of deep neural networks: A tutorial and survey," *Proc. IEEE*, vol. 105, no. 12, pp. 2295–2329, Dec. 2017.

[19] Z. Wang. (2015). *The Application of Deep Learning on Traffic Identification*. [Online]. Available: http://www.blackhat.com

[20] 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. 55380–55391, 2018.

[21] M. Lotfollahi, R. S. H. Zade, M. J. Siavoshani, and M. Saberian. (2017). "Deep packet: A novel approach for encrypted traffic classification using deep learning." [Online]. Available: https://arxiv.org/abs/1709.02656

[22] W. Wang, M. Zhu, X. Zeng, X. Ye, and Y. Sheng, "Malware traffic classification using convolutional neural network for representation learning," in *Proc. Int. Conf. Inf. Netw.*, Jan. 2017, pp. 712–717.

[23] W. Wang, M. Zhu, J. Wang, X. Zeng, and Z. Yang, "End-to-end encrypted traffic classification with one-dimensional convolution neural networks," in *Proc. IEEE Int. Conf. Intell. Secur. Inform. (ISI)*, Jul. 2017, pp. 43–48.

[24] 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. 18042–18050, 2017.

[25] G. Aceto, D. Ciuonzo, A. Montieri, and A. Pescapé, "Mobile encrypted traffic classification using deep learning," in *Proc. Netw. Traffic Meas. Anal. Conf. (TMA)*, Jun. 2018, pp. 1–8.

[26] W. Wang *et al.*, "HAST-IDS: Learning hierarchical spatial-temporal features using deep neural networks to improve intrusion detection," *IEEE Access*, vol. 6, pp. 1792–1806, 2018.

[27] S. Rezaei and X. Liu, "How to achieve high classification accuracy with just a few labels: A semi-supervised approach using sampled packets," *CoRR*, Dec. 2018.

[28] V. Tong, H.-A. Tran, S. Souihi, and A. Mellouk, "A novel QUIC traffic classifier based on convolutional neural networks," 2018, pp. 1–6.

[29] H. Zhou, Y. Wang, X. Lei, and Y. Liu, "A method of improved CNN traffic classification," in *Proc. 13th Int. Conf. Comput. Intell. Secur. (CIS)*, Dec. 2017, pp. 177–181.

[30] Z. Chen, K. He, J. Li, and Y. Geng, "Seq2img: A sequence-to-image based approach towards IP traffic classification using convolutional neural networks," in *Proc. IEEE Int. Conf. Big Data (Big Data)*, Dec. 2017, pp. 1271–1276.

[31] D. Li, Y. Zhu, and W. Lin, "Traffic identification of mobile apps based on variational autoencoder network," in *Proc. 13th Int. Conf. Comput. Intell. Secur. (CIS)*, Dec. 2017, pp. 287–291.

[32] L. Vu, C. T. Bui, and U. Nguyen, "A deep learning based method for handling imbalanced problem in network traffic classification," in *Proc. 8th Int. Symp. Inf. Commun. Technol.*, 2017, pp. 333–339.

[33] P. Wang, F. Ye, and X. Chen, "Smart devices information extraction in home Wi-Fi networks," *Internet Technol. Lett.*, vol. 1, no. 3, p. e42. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/itl2.42

[34] 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. 30483–30490, 2018.

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

[36] P. Wang, X. Chen, and Z. Sun, "Performance modeling and suitability assessment of data center based on fog computing in smart systems," *IEEE Access*, vol. 6, pp. 29587–29593, 2018.

[37] P. Goyal and A. Goyal, "Comparative study of two most popular packet sniffing tools-Tcpdump and Wireshark," in *Proc. 9th Int. Conf. Comput. Intell. Commun. Netw. (CICN)*, Sep. 2017, pp. 77–81.

[38] R. Hofstede *et al.*, "Flow monitoring explained: From packet capture to data analysis with NetFlow and IPFIX," *IEEE Commun. Surveys Tuts.*, vol. 16, no. 4, pp. 2037–2064, 4th Quart., 2014.

[39] G. Draper-Gil, A. H. Lashkari, M. S. I. Mamun, and A. Ghorbani, "Characterization of encrypted and VPN traffic using time-related features," in *Proc. ICISSP*, 2016, pp. 407–414.

[40] R. Hasibi, M. Shokri, and M. Dehghan, "Augmentation scheme for dealing with imbalanced network traffic classification using deep learning," *CoRR*, Jan. 2019.

[41] J. Höchst, L. Baumgärtner, M. Hollick, and B. Freisleben, "Unsupervised traffic flow classification using a neural autoencoder," in *Proc. IEEE 42nd Conf. Local Comput. Netw. (LCN)*, Oct. 2017, pp. 523–526.

[42] Y. Mirsky, T. Doitshman, Y. Elovici, and A. Shabtai, "Kitsune: An ensemble of autoencoders for online network intrusion detection," *CoRR*, Feb. 2018.

[43] G. Kathareios, A. Anghel, A. Mate, R. Clauberg, and M. Gusat, "Catch it if you can: Real-time network anomaly detection with low false alarm rates," in *Proc. ICMLA*, Dec. 2017, pp. 924–929.

[44] N. Dilokthanakul *et al.*, "Deep unsupervised clustering with Gaussian mixture variational autoencoders," *CoRR*, Jan. 2016.

[45] H. Zhang, C. Wu, S. Gao, Z. Wang, Y. Xu, and Y. Liu, "An effective deep learning based scheme for network intrusion detection," in *Proc. ICPR*, 2018, pp. 682–687.

[46] R. C. Aygun and A. G. Yavuz, "Network anomaly detection with stochastically improved autoencoder based models," in *Proc. IEEE 4th Int. Conf. Cyber Secur. Cloud Comput. (CSCloud)*, Jun. 2017, pp. 193–198.

[47] I. Goodfellow *et al.*, "Generative adversarial nets," in *Advances in Neural Information Processing Systems*, Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, Eds. Curran Associates, 2014, pp. 2672–2680. [Online]. Available: http://papers.nips.cc/paper/5423-generative-adversarial-nets.pdf

[48] H. Lee, S. Han, and J. Lee, "Generative adversarial trainer: Defense to adversarial perturbations with GAN," *CoRR*, May 2017.

[49] C. Ledig *et al.*, "Photo-realistic single image super-resolution using a generative adversarial network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2017, pp. 105–114.

[50] H.-W. Dong, W.-Y. Hsiao, L.-C. Yang, and Y.-H. Yang. (2017). "MuseGAN: Multi-track sequential generative adversarial networks for symbolic music generation and accompaniment." [Online]. Available: https://arxiv.org/abs/1709.06298

[51] O. Olabiyi, A. Salimov, A. Khazane, and E. T. Mueller, "Multi-turn dialogue response generation in an adversarial learning framework," *CoRR*, Dec. 2018.

[52] W. Hu and Y. Tan, "Generating adversarial malware examples for black-box attacks based on GAN," *CoRR*, Feb. 2017.

[53] J.-Y. Kim, S.-J. Bu, and S.-B. Cho, "Malware detection using deep transferred generative adversarial networks," in *Proc. Int. Conf. Neural Inf. Process.*, 2017, pp. 556–564.

[54] Z. Lin, Y. Shi, and Z. Xue, "IDSGAN: Generative adversarial networks for attack generation against intrusion detection," *CoRR*, Sep. 2018.

[55] M. Salem, S. Taheri, and J. S. Yuan, "Anomaly generation using generative adversarial networks in host based intrusion detection," *CoRR*, Dec. 2018.

[56] M. Rigaki and S. Garcia, "Bringing a GAN to a knife-fight: Adapting malware communication to avoid detection," in *Proc. IEEE Secur. Privacy Workshops (SPW)*, May 2018, pp. 70–75.

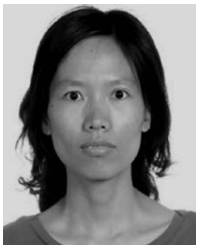
[57] L. Vu, D. Van Tra, and U. Nguyen, "Learning from imbalanced data for encrypted traffic identification problem," *Proc. 7th Symp. Inf. Commun. Technol.*, 2016, pp. 147–152.

[58] S. E. Gómez, L. Hernández-Callejo, B. C. Martínez, and A. J. Sánchez-Esguevillas, "Exploratory study on class imbalance and solutions for network traffic classification," *Neurocomputing*, vol. 343, pp. 100–119, 2019. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S092523121930164X

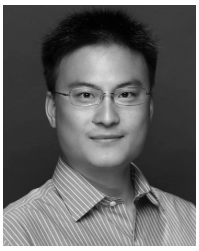
[59] B. Yan, G. Han, Y. Huang, and X. Wang, "New traffic classification method for imbalanced network data," *J. Comput. Appl.*, vol. 38, no. 1, pp. 20–25, 2018. [Online]. Available: http://www.joca.cn/EN/abstract/article_21447.shtml

[60] L. Vu, C. T. Bui, and Q. U. Nguyen, "A deep learning based method for handling imbalanced problem in network traffic classification," in *Proc. 8th Int. Symp. Inf. Commun. Technol. (SoICT)*, New York, NY, USA, 2017, pp. 333–339. doi: 10.1145/3155133.3155175.

PAN WANG (M'18) received the B.S. degree from the Department of Communication Engineering, Nanjing University of Posts and Telecommunications, Nanjing, China, in 2001, and the Ph.D. degree in electrical and computer engineering from the Nanjing University of Posts and Telecommunications, Nanjing, in 2013, where he is currently an Associate Professor with the School of Modern Posts. From 2017 to 2018, he was a Visiting Scholar with the Department of Electrical and Computer Engineering, University of Dayton (UD). His research interests include cyber security and communication network security, network measurements, quality of service, deep packet inspection, SDN, and big data analytics and applications.



XUEJIAO CHEN (M'18) received the B.S. degree from the Department of Communication Engineering, Nanjing University of Posts and Telecommunications, Nanjing, China, in 2001, and the master's degree in electrical and computer engineering from the Nanjing University of Posts and Telecommunications, in 2006. From 2017 to 2018, she was a Visiting Scholar with the Department of Electrical and Computer Engineering, University of Dayton (UD). She is currently an Assistant Professor with the Department of Communication Engineering, Nanjing College of Information Technology, Nanjing. Her research interests include wireless communications and networks, cyber security and communication network security, network measurements, quality of service, and deep packet inspection.



FENG YE (S'12–M'15) received the B.S. degree from the Department of Electronics Engineering, Shanghai Jiaotong University, Shanghai, China, in 2011, and the Ph.D. degree in Electrical and Computer Engineering from the University of Nebraska-Lincoln (UNL), NE, USA, in 2015. He is currently an Assistant Professor with the Department of Electrical and Computer Engineering, University of Dayton (UD), Dayton, OH, USA. Prior to joining UD, he was with the Department of ECE, UNL, as an Instructor and a Researcher, from 2015 to 2016. His research interests include cyber security and communication network security, wireless communications and networks, green ICT, smart grid communications and energy optimization, and big data analytics and applications. He serves as a TPC member for numerous international conferences, including INFOCOM, GLOBECOM, VTC, and ICC. He serves as the Co-Chair of ICNC'19 Signal Processing for Communications Symposium; the Publicity Co-Chair of IEEE CBDCom 2018; and the Co-Chair of Cognitive Radio and Networking Symposium, and IEEE ICC 2018. He was a recipient of the 2015 Top Reviewer of the IEEE Vehicular Technology Society. He is also a reviewer for several IEEE journals, including the IEEE TRANSACTIONS ON BIG DATA, the IEEE TRANSACTIONS ON GREEN COMMUNICATIONS AND NETWORKING, the IEEE TRANSACTIONS ON SMART GRID, the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, and the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS. He serves as the Secretary of the IEEE Technical Committee on Green Communications and Computing (TCGCC). He is currently an Associate Editor of *Security and Privacy* (Wiley), and *China Communications*.

ZHIXIN SUN was born in Xuancheng, China, in 1964. He received the Ph.D. degree from the Nanjing University of Aeronautics and Astronautics, Nanjing, China, in 1998. From 2001 to 2002, he held a postdoctoral position with the School of Engineering, Seoul National University, South Korea. He is currently a Professor and the Dean of the School of Modern Posts, Nanjing University of Posts and Telecommunications. His research interests include cloud computing, cryptography, and traffic identification.