Hindawi
Security and Communication Networks
Volume 2020, Article ID 8872923, 17 pages
https://doi.org/10.1155/2020/8872923

Review Article

Network Attacks Detection Methods Based on Deep Learning Techniques: A Survey

Yirui Wu, Dabao Wei, and Jun Feng

College of Computer and Information, Hohai University, Nanjing, China

Correspondence should be addressed to Jun Feng; fengjun@hhu.edu.cn

Received 7 May 2020; Revised 26 June 2020; Accepted 20 July 2020; Published 28 August 2020

Academic Editor: Xiaolong Xu

With the development of the fifth-generation networks and artificial intelligence technologies, new threats and challenges have emerged to wireless communication system, especially in cybersecurity. In this paper, we offer a review on attack detection methods involving strength of deep learning techniques. Specifically, we firstly summarize fundamental problems of network security and attack detection and introduce several successful related applications using deep learning structure. On the basis of categorization on deep learning methods, we pay special attention to attack detection methods built on different kinds of architectures, such as autoencoders, generative adversarial network, recurrent neural network, and convolutional neural network. Afterwards, we present some benchmark datasets with descriptions and compare the performance of representing approaches to show the current working state of attack detection methods with deep learning structures. Finally, we summarize this paper and discuss some ways to improve the performance of attack detection under thoughts of utilizing deep learning structures.

1. Introduction

The continuous development and extensive usage of Internet benefit numerous network users from a quantity of aspects. Meanwhile, network security becomes much more important with wide usage of network. Network security is closely related to computers, networks, programs, various data, and so forth, where the purpose of defense is to prevent unauthorized access and modification [1]. However, the growing number of internet-connected systems in finance, E-commerce, and military makes them become targets of network attacks, resulting in large quantity of risk and damage. Essentially, it is necessary to provide effective strategies to detect and defend attacks and maintain network security. Furthermore, different kinds of attacks are usually required to be processed in different ways. How to identify different kinds of network attacks thus becomes the main challenge in domain of network security to be solved, especially those attacks never seen before.

Over the past several years, researchers have used various kinds of machine learning methods to classify network attacks without prior knowledge of their detailed characteristics. However, traditional machine learning methods are not capable of providing distinctive feature descriptors to describe the problem of attack detection, due to their limitations in model complexity. Recently, machine learning has made a great breakthrough by simulating human brain with structure of neural networks, which are named deep learning methods for their general architecture of deep layers to solve complicated problems. Among these successful applications, Google's AlphaGo is one of the most outstanding trials for the game of "go," involving the strength of a typical kind of deep learning structure, that is, convolutional neural networks.

Since deep learning is complex in its original structures and domain-oriented applications, this paper is written to explain so for those who aim to study in the field of network security by utilizing deep learning methods. Essentially, there exists a quantity of previous work focusing on attack detection using deep learning techniques. Among them, several literature reviews [2–8] have been conducted to get ideas from applying deep learning on attack detection, which

is the foundation of our paper. For example, Berman et al. [5] provide a quantity of reading resources to describe the basic knowledge and development history of deep learning methods and their corresponding applications in attack detection. Different from a complete view on this specific domain brought by Berman et al. [5], Apruzzese et al. [4] focus on explaining attack detection methods related to intrusion detection, malware analysis, and spam detection. In work of Wickramasinghe et al. [7], they mainly review deep learning methods on securing under the usage of Internet of Things technologies, which offers a clear view on variant kinds of cyberattacks and the corresponding techniques used in detection. Afterwards, Aleesa et al. [3] review and analyze the research status of intrusion detection system based on deep learning technology among four major databases. Meanwhile, they offer a systematic literature review of the relevant articles using the keywords "deep learning", "invasion", and "attack" selection, which provide a wide range of resource background for the researchers. By regarding dataset as significantly important to intrusion detection, Ferrag et al. [6] describe 35 well-known network datasets and divide them into seven categories. They introduce seven presentative models for each category, where they evaluate and compare the efficiency via accuracy and false alarm rate based on real traffic datasets, that is, CSE-CIC-IDS2018 and Bot-IoT.

In fact, all the above review papers have their own emphases, such as security applications, attacks type, datasets, or databases. Unlike former methods, we intend to build our paper on the basis of deep learning models, thus paying special attention to attack detection methods built on different kinds of deep learning architectures. Furthermore, we offer a fair comparison and our own specified analysis on performance of representing approaches based on benchmark datasets. We believe our paper could offer a more understandable reading resource for readers, who are interested in how different deep learning architectures affect the area of attack detection.

In the paper, we attempt to build up basics for future research through a thorough literature review of deep learning related approaches in the field of attack detection. More specially, firstly, we summarize the fundamental problems, classify the previous methods, and review the useful methods for beginners. Then, we briefly introduce the great progress on deep learning techniques in cybersecurity. By replacing traditional machine learning methods with deep learning structures, researchers have proposed a quantity of novel algorithms to greatly improve the performance referring to higher detection rate and lower false alarm rate. Afterwards, we compare and analyze the performance of some representative deep learning approaches on benchmark dataset. Finally, we make a summary of the problems to be solved and future direction of deep learning method to improve attack detection.

We organize the rest of our work as follows. Section 2 focuses on concepts of attack detection and cyber applications via research background introduction. Section 3 offers overviews on different deep learning methods for attack detection, which are categorized as unsupervised and supervised methods with different structures. Section 4 presents datasets and analyzes the performance comparisons of a quantity of deep learning methods. Section 5 provides discussion and conclusion based on the current foundations and presents several ideas for future research.

2. Brief Introduction to Attack Detection

In order to provide an overview of effective attack detection based on deep learning techniques, it is essential to introduce background knowledge. We thus first give a brief introduction to the concepts of attack detection, which could offer a basic recognition for new learners. Afterwards, we make a brief representation of successful applications for cybersecurity.

2.1. Developing Process of Attack Detection. Attacks could be recognized as the attempts to bypass security policies of the system, which gives attackers easier access to obtain or modify information, even destroying the system. With technologies developing on wireless communication systems, serious threats to network security, especially security of wireless communication systems, have been proposed by more frequent network attack activities, due to openness characteristics of wireless channels. Since we are now in machine learning and big data epoch [9], cybersecurity in wireless communication systems is important for users to protect network, computer, and data from attacks. There exist variant kinds of attacks for cyber systems, such as flooding, distributed denial of service, abnormal packet attack, and spoofing.

To deal with such attack threads to cybersecurity, researchers have proposed many solutions [10]. Among the solutions, attack detection is one of the most effective ways, which offers a complete and dynamic security mechanism to monitor, prevent, and resist attacks. Specifically, attack detection would collect information by monitoring network, system status, behavior, and the usage of system, which could automatically detect unauthorized usage of system users and attacks of external attackers on the system.

In recent years, machine learning is developing with incredible speed. Among different machine learning methods, deep learning structures construct artificial neural networks to simulate interconnecting neurons of human brains, which brings distinctive power to solve complicated problems. Researchers thus adopt various deep learning methods to operate attack detection, resulting in significant achievements. However, there are still many unsolved problems due to the limitation of deep learning methods. It is essential to make a summarization of how former methods use deep learning methods to detect attacks, which could bring new ideas for future developments.

2.2. Applications of Attack Detection Using Deep Learning Structures. Since deep learning shows great potential in constructing security applications, it has been widely used in cybersecurity [11]. There are numerous related applications such as malware, intrusion, phishing, spam detection, and

traffic analysis [12]. We believe these successful application examples could help analyze users' requirements with the innovation brought by deep learning structures. Thus, we provide some typical applications to present practicability of deep learning method, where we believe these applications can be implemented in domains of multimedia handling [13], signal processing [14], and so on [15].

2.2.1. Intrusion Detection. Intrusion detection system could detect malicious activities by collecting and analyzing network behavior, security log, and other information available on the network and among connected computers [16]. Essentially, intrusion detection system checks existence of abnormal behaviors against system security policy and signs of being attacked in the system, which is capable of protecting the system with real-time responses. Under traditional system settings, intrusion detection system works as a reasonable, active, and efficient supplement to firewall, which actually acts as a passive defense mean to attacks.

Traditional intrusion detection system is firstly built on misuse of intrusion detection technology, which mainly extracts characteristics or rules of intrusion behavior. After appearance of abnormal behavior detection technology with traditional machine learning models, intrusion detection system evolutes to carry out probability statistical modeling for normal behaviors, which could analyze and alarm abnormal behaviors with large deviation. However, such system may have unsatisfactory results, due to low capability in problem space defining and complexity in modeling malicious activities.

To further overcome shortcomings brought by traditional machine learning methods, deep learning technology is performed to analyze network packets, which progressively changes the mainstream idea of intrusion detection from blacklist to white model. A new NIDS deep learning model is proposed by Shone et al. [17], which is helpful to analyze the network traffic under the symmetrical deep autoencoder technology. On the basis of LSTM algorithm, Vinayakumar et al. [18] design a system call modeling approach with integration method for anomaly intrusion detection system. System call modeling helps capture the semantics of each call and relationship on the network. The integration method mainly focuses on the false alarm rate in accordance with IDS design. Currently, a mature intrusion detection system could detect many kinds of attacks with the strength of deep learning structures.

2.2.2. Malware Detection. Malware is designed to reduce performance and vulnerability of a computer, server, or computer network. Under extreme situations, Malware will result in destruction of the entire system. Malware requires to be implanted into the target computer at first. Afterwards, it could execute code, script, active content, and other software automatically or following orders from planters. It is noted that such software or codes could be categorized in forms of computer viruses, worms, Trojans, spyware, advertising software, and malicious codes.

We divide the malware detection methods into two categories, that is, signature-based and anomaly-based detection. Traditional antivirus software can be included in the first category, which detects malicious files based on file signature. However, slightly deformed malicious codes could be bypassed, leading to a large number of false positives. Later, technologies of sandbox and virtual machine appear to detect dynamic behaviors of virus, which can be regarded as big progress from static detection to dynamic analysis, greatly improving the ability to detect unknown malicious code.

For example, in [19], Saxe discusses the deep learning of a four-layer network application. In order to get appropriate computing feature text extraction technology, PE Metadata Features can be used. The author proposes eXpose neural network, where their network takes the original short strings as input and extracts features to classify with character-level embeddings. Because of the feature design of self-extraction, the method of express is better than the baseline method based on manual feature extraction. Pascanu et al.'s [20] echo state network is helpful to extract all information by random time projection technology, the max pooling is used for nonlinear sampling of data, and the logistic regression is used for final classification of data.

2.2.3. Domain Generate Algorithms. DGAs are popular to be used as malware tools to create a great quantity of domain names for tracking communication with C2 server. Different domain names make it difficult to use standard technologies like blacklist or sink-holing to prevent malicious domain names. DGAs are often used in various network attacks, such as spam, personal data theft, and DDoS attacks.

By applying deep learning technologies, DGA is capable of detecting domain names from the perspective of syntax analysis. Specifically, such novel algorithms could not only compare word frequency with normal domain names by n-gram methods but also compare the probabilities of character combination with normal domain names by HMM method. Moreover, it is capable of analyzing the entropy, consonant letter, and other characteristics of domain names, which are utilized in LSTM for abnormal classification. Due to the slow speed and poor performance of traditional technology, Feng et al. [21] provide a deep learning method which helps to distinguish DGA domains from non-DGA domains. In [22], the advantages of featureless extraction of raw domain names as an input in LSTM network are also discussed.

3. Deep Learning Methods for Attack Detection

Considering the current deep learning methods for attack detection [23] and following the categorization of the previous works [24, 25], we roughly divide them into three categories as well, that is, unsupervised (e.g., autoencoder (AE), deep belief network (DBN), and generative adversarial network (GAN)), supervised (e.g., deep neural network (DNN), convolutional neural network (CNN), and recurrent neural network (RNN)), and other hybrid methods; we show

the details of categorization in Figure 1. Essentially, there exist other classification criterions. For example, Berman et al. [5] review the related deep learning methods according to attacks type and focus on how deep learning is used for various attacks. Moreover, Al-Garadi et al. [2] offer a comprehensive view of deep learning methods based on the applications of cybersecurity.

Adopting different kinds of deep learning algorithms could bring variant advantages for attack detection methods. Supervised learning based methods often result in high accuracy, due to quantity of information provided by manually labeled samples. Without sufficient knowledge from labeled data, unsupervised learning based methods are generally low in performance. However, manually labelling is a time-consuming task, especially for complex attacks. There even exist cases that cannot be described by a simple label, due to the inherent complexity of real-world network attacks. Therefore, unsupervised learning based methods could perform well without prior knowledge of attacks, which is an obvious advantage. Hybrid methods decrease the number of training samples and maintain a relatively high performance, which is suitable to deal with variant attack situations. However, it is generally complex in structure and high in computing time, which prevents its wide usage.

3.1. Unsupervised Learning for Attack Detection

3.1.1. Autoencoder Based Methods for Attack Detection. Let us first introduce the architecture of AE, which can be regarded as a data compression algorithm with neural network structure. In fact, it is capable of firstly compressing the input into feature space representation and then reconstructing representation into the output. Since AE can be regarded as a typical representing learning algorithm, it is widely used for dimension reduction and outlier detection. Researchers in cybersecurity also adopt AE to represent abnormal behaviors in its compressed feature space, which brings the advantage of dynamical representation for unknown category of attacks.

To extract informative feature descriptors from original network traffic data, Zhang et al. [26] propose to detect network intrusion by stacking dilated convolutional AE (DCAEs), which is a successful combination of self-taught and representation learning. Specifically, original network traffic data is firstly transformed into a vector through the preprocessing step. During unsupervised training, DCAEs learn the hierarchical structure of feature representation from a large number of unlabeled samples. Afterwards, use backpropagation algorithm and a few labeled instances to fine-tune and improve feature description ability learned from the unlabeled instances. In fact, using original network traffic and unsupervised pretraining makes their model more adaptive and flexible to deal with complicated raw data.

Following the idea to facilitate intrusion detection with AE models, Shone et al. [17] propose nonsymmetric deep AE (NDAE) for unsupervised feature blearing, which successfully reduces computations cost of analysis by combining AE with shallow learning. Specifically, NDAE has an additional coding stage comparing with typical AE, which could reduce complexity and improve the accuracy of the model. We show such structure in Figure 2, where we can observe its hierarchical feature extractor. At the end of their proposed NDAE, they apply the structure of random forest to recognize abnormal situations with the help of feature representation learned from NDAEs. To evaluate their model, the authors have implemented their codes in GPU and evaluated with KDDCup 99 and NSL-KDD, achieving promising results comparing with others.

Since AE is capable of learning potential representation of unknown attacks, Yousefi-Azar et al. [27] propose to learn feature representation with AE structure for different cybersecurity applications, which consists of two training stages, that is, pretraining and fine-tuning. The former stage is designed to search for an appropriate starting point for the fine-tuning stage. After determining the parameters in the pretraining stage, fine-tune stage will coverage offering feature description for input data. Their proposed feature learning scheme can greatly reduce feature dimensions, thus significantly minimizing storage requirements. Experiment results show their feature representation can be used in many domains and could achieve remarkable results comparing with previous works.

Since collected network raw data can be unbalanced in distribution, Farahnakian et al. [28] utilize deep stacked autoencoder to focus on important and informative feature representations, thus constructing classification models to detect abnormal behaviors. Specifically, their proposed network consists of 4 AEs in sequential order, which will be trained in a greedy layerwise fashion. Experimental results on KDDCup 99 dataset show it could achieve high accuracy for abnormal detection, that is, 94.71%, even under the situation of unbalanced data.

In order to construct a flexible system for detecting intrusion attacks, Javaid et al. [29] utilize sparse AE and softmax-regression layer for construction and self-taught learning (STL) for the training process. Specifically, their proposed STL could be divided into two steps, where sparse AE is used for unsupervised feature learning at first and softmax-regression is used for classification after feature extraction. In fact, usage of STL could largely improve the learning ability of constructed network facing unknown attacks, where new categories of attacks can be incrementally analyzed during runtime without troubles of training from scratch.

Following such idea, Papamartzivanos et al. [30] present a more powerful approach with MAPE-K framework, which could construct a misuse intrusion detection system with scalable, self-adaptive, and autonomous characteristics. It could extract generalized features for problem reconstruction, even facing unknown environment and using unlabeled data. They believe their proposed method could work well by grasping the nature of variant attacks, where they further design experiments to show that their method could deal with new situations without updating the training set manually.

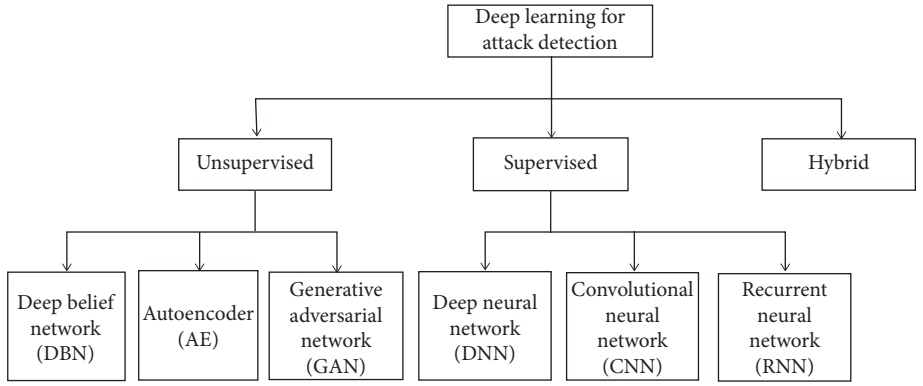

Figure 1: Categorization of the current deep learning methods for attack detection.

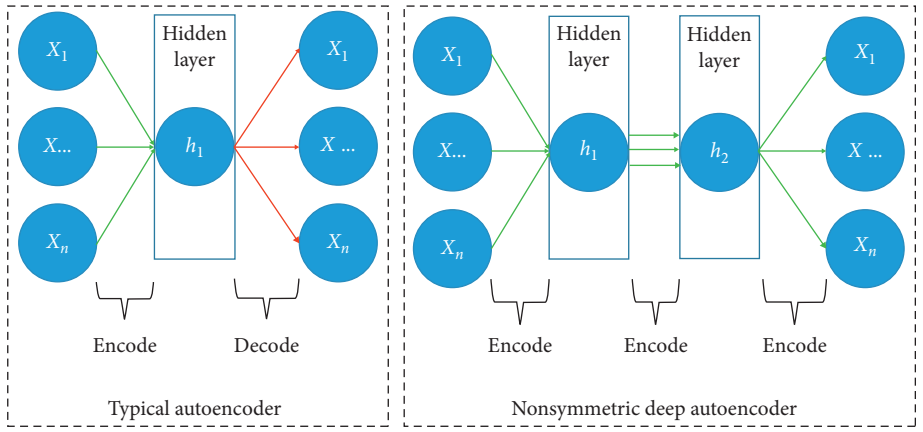

Figure 2: Network structure of Shone et al. [17], which is a novel structure of AE designed with nonsymmetrical multiple hidden layers.

Feature extraction is one of the major issues to address for attack detection. Regarding AE as a structure for information compressing and feature generation, utilizing AE brings advantages of automatical and dynamical feature construction, resulting in high accuracy for detecting predefined attacks existing in datasets. Facing variant and unknown attacks which are the main characteristics in cybersecurity, researchers have emphasized self-learning strategies to make AE more powerful.

3.1.2. Deep Belief Network Based Methods for Attack Detection. Deep belief network (DBN) could be divided into two categories, that is, restricted Boltzmann machines (RBM) with several layers of unsupervised learning networks and backpropagation neural network (BPNN or BP) with one such layer. Essentially, RBM is a random structure of generating neural network, which is undirected graph model composed of different layers constructed by visible neurons and hidden neurons. Due to the natural characteristics of RBM, it is effective for DBN to train layer by layer.

Early, Gao et al. [31] focus on dealing with big raw data and apply deep belief network to construct such intrusion detection system. In their paper, they try different DBN models by adjusting parameters like number of layers and hidden layers. They find the best parameter settings for DBN is a four-layer DBN model, which could achieve better performance than other machine learning methods on KDDCup 99 dataset.

Afterwards, Ding et al. [32] represent malware as opcode sequences and use DBN to detect malware, where they use unsupervised learning to pretrain a multilayer generative model to help DBNs solve the overfitting problem. We show its structure in Figure 3, where we can observe DBN works as classier in the whole workflow with steps of RBM training and BP fine-tuning. With the help of additional unlabeled data, their proposed DBN could achieve accuracy as high as 96%, which outperforms three other traditional artificial intelligence models, that is, SVM, kNN, and decision tree. However, their methods are not justified by other metrics.

Since behavioral characteristics of ad hoc networks have brought great challenges to network security, Tan et al. [33] propose a deep belief network based on ad hoc network intrusion detection structure. Their proposed DBN model contains 6 modules: wireless monitoring node for data fetching, data fusion module to fuse useful data and remove redundancy, DBN training module and DBN intrusion module to train and identify whether there is intrusion, respectively, and response module that expresses results of the proposed model to users. Experimental results show their proposed method can reach 97.6% in accuracy, leading

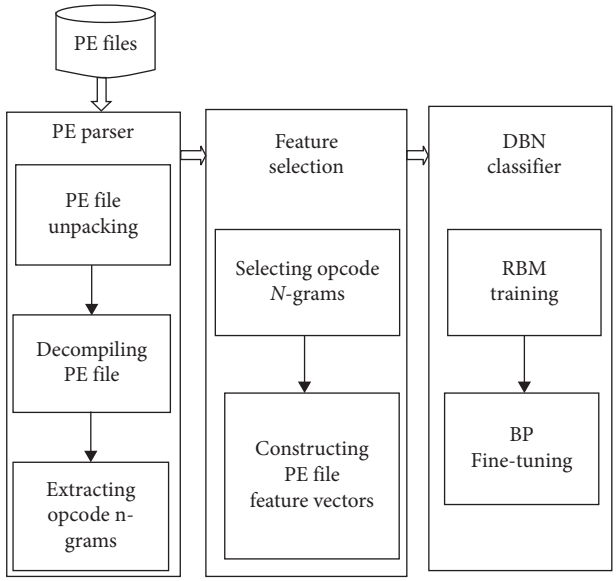

Figure 3: Workflow of opcode malware detection approach proposed by Ding et al. [32], which consists of three major components: PE (Portable Executable) parser, feature extractor, and malware detection module. It is noted that DBN is the core classier of malware detection module.

it to be fit with implementation in intrusion detection applications.

To explore the capabilities of DBN for detecting intrusion attacks, Alom et al. [34] propose an effective platform to explain intrusion attempts in network traffics. Their constructed system firstly uses digital encoding and standardized method to select features and then uses DBN to classify network intrusion by assigning class label to each feature vector. According to their experiments and analysis, their constructed system can not only detect attacks, but also accurately identify and classify network activities according to limited, incomplete, and nonlinear data sources.

Many trials have been applied in using DBN for intrusion detection. However, there still exist many unsolved problems, such as redundant information, easy to trap into local maximal. To solve these problems, Zhao et al. [35] propose to detect intrusion attacks by involving strength of DBN and probabilistic neural network (PNN). Firstly, they rescale original input data to low-dimensional by utilizing nonlinear describing capability of DBN. Meanwhile, DBN could maintain basic characteristics of original data in representation. Secondly, particle swarm optimization algorithm is used to reduce the size of hidden nodes of every layer. Thirdly, PNN is introduced to classify low-dimensional information. Their experiments on KDDCup 99 dataset show they have solved the above problem to a certain extent.

Regarding real-time attacks detection as the biggest challenge of intrusion detection, Alrawashdeh and Purdy [36] propose an anomaly detection method based on DBN, which only consists of one-hidden layer RBM and a fine-tuning layer constructed by logistic regression classifier. Their simplest design of DBN achieves instant running and best performance (reported as 97.7% in accuracy and 8s CPU time for each instance) when testing with KDDCup 99 dataset. Their method offers possibility to implement deep learning methods for attack detection on low computation resource platforms like drones, cell phones, and personal computers, which greatly expands usage scenarios of such methods.

Because the traditional intrusion detection approaches face difficulties dealing with high-speed network data and cannot detect the unknown attacks at present, Zhang et al. [37] propose a network attack detection model integrating flow calculation and deep learning, which comprises two parts: real-time detection algorithm based on frequent patterns and a classification algorithm based on the DBN and SVM. Sliding window stream data processing can realize real-time detection, and the DBN-SVM algorithm can improve classification accuracy. Based on the CICIDS2017 dataset, several groups of comparative experiments are carried out. The method's real-time detection efficiency is higher than that of traditional algorithms.

3.1.3. Generative Adversarial Network Based Methods for Attack Detection. Due to property of discovering inherent pattern of data to generate new samples, generative adversarial network (GAN) is one of the most promising unsupervised learning methods proposed in recent years. The main inspiration of GAN comes from the idea of zero-sum game. When it is applied to deep neural network, it keeps playing games between generator G and discriminator D, and finally G is capable of learning distribution representation of actual data. G is to imitate, model, and learn distribution characteristics of real data as much as possible, while the task of D is to distinguish whether an input data comes from real data or output of G. Through the continuous competition between these internal models, the generation ability and discrimination ability of both G and D can be greatly improved.

Even though GAN is new in conception and hard in the training process, researchers successfully build several attack detection applications by regarding it as basic structure. For instance, Erpek et al. [38] propose a GAN-based approach to detect jamming attacks on wireless communications and defend it based on collected information of attacks. Specifically, their model consists of a transmitter, a receiver, and a jammer. A pretrained classifier is adopted by the transmitter to predict the current channel state and decide whether to send based on the latest sensing results, while the jammer collects the channel state and ACKs to construct a classifier, which could predict next transmission and block it successfully. The jammer uses classification score to control the power under the average power constraint. Afterward, a GAN is designed to perform as a jammer, which can cut down collection time by adding synthetic samples.

Utilizing machine learning technology to perform phishing detection, that is, URL of fake web address, is popular, due to its high effectiveness and real-time response. However, adversary may bypass URL classification algorithm by modifying components. To solve this problem,

AlEroud and Karabatis [39] propose to generate URL-based phishing examples by using generator of GAN, which are then shipped to discriminator, that is, black-box phishing detector. In their proposed GAN model where its structure is shown in Figure 4, generator network could generate disturbed versions of real phishing examples and convert them into adversary examples. Discriminator network learns to classify both generated examples and real ones working as a phishing detector, where the generator parameters and weights are updated with information passing from the discriminator. After testing with a public phishing dataset, their experimental results show that their proposed GAN is successful by avoiding a large number of unknown phishing examples.

GAN is not often used for attack detection field. In fact, GAN is in fast developing in terms of structures, algorithms, and so forth. At present, GAN have shown promising results in many domains, which lead us to believe this proposing new technique to synthesize attempts is quite significant in creating a defensive mechanism. Such novel defensive mechanism can further complete quantity of tasks, such as preventing zero-day phishing attempts, performing opinion spam, and detecting intrusion attacks. Therefore, we think there exists a broad research space to connect GAN structure with attack detection filed.

3.2. Supervised Learning for Attack Detection

3.2.1. Deep Neural Network Based Methods for Attack Detection. DNN is recognized as multilayer perceptron due to characteristic of multiple hidden layers. Such multilayer feature brings advantage to express complex functions with fewer parameters, which makes DNN capable of facilitating tasks of feature extraction and representation learning. Essentially, there exist three categories of layers in DNN. Generally speaking, we regard the first layer as input layer, the last layer as output layer, and middle layers as hidden layers.

To provide a solution to network security problem, Tang et al. [40] propose a DNN model to perform flow based anomaly detection. Their first attempt in applying DNN for network security results in a relatively simple DNN, which is composed of one input layer, three hidden layers, and one output layer. Some experiments are carried out on NSL-KDD dataset, where the proposed DNN model is proven to detect zero-day attack and behaves better than the other machine learning methods.

To enhance ability of DNN, Li et al. [41] propose a novel network structure called HashTran-DNN to classify Android malware. We show its architecture design in Figure 5, where we can observe their most innovation point lies in transforming input samples by using hash functions to preserve locality characteristics. After transforming input data, HashTran-DNN uses AE to perform denoising task, so that DNN classifier can obtain locality information in the potential space for better performance. After analyzing the experimental results, we can observe that HashTran-DNN can effectively defend against four special testing attacks, where standard DNN fail to detect all these attacks.

Challenges arise motivated by the fact that malicious attacks are constantly varying and occur on very large volumes which require scalable solutions. To meet this challenge, a DNN structure with a scalable and hybrid design is proposed by Vinayakumar et al. [18], which can watch network traffic and host level events in real-time, actively warning possible network attacks. Specifically, their proposed framework adopts scalable computing architecture, text representation method, and DNNs to meet the requirement to process big data, where DNN could help improve the performance of their model with functions of nonlinear activation.

For network administrator, it is an urgent task to prevent the invasion of malicious network hackers and keep the network system and computer in a safe and normal operation state. Peng et al. [42] propose a network intrusion detection method based on deep learning, which uses deep neural network to extract features of network monitoring data, and BP neural network is used to classify intrusion types. The method is evaluated by KDDCup 99 dataset. The results show that the method achieves the accuracy of 95.45%, and it has a significant improvement while compared with the traditional machine learning method.

3.2.2. Convolutional Neural Network Based Methods for Attack Detection. CNN involves convolution computation and depth structure, which is a representative and commonly used techniques in deep learning domain. Specifically, CNN uses multilayer perception variant design requiring minimal preprocessing. The basic structure of CNN is composed of input and output layers and multiple hidden layers which include convolution, pooling, and full connection layer. Compared with other classification algorithms, CNN uses relatively less preprocessing and is independent of feature design containing prior knowledge, which are its main advantages.

Convolutional neural network has been applied to network security field with much promising progress. For example, Kolosnjaji et al. [43] attempt to construct a neural network with convolutional and recursive network layers, which obtains classification features to model malware detection system. Through their proposed method, they obtain a hierarchical feature extraction architecture, which combines advantages of convolution operation from convolutional layer and sequence modeling from recursive network layer. Afterwards, Kolosnjaji et al. [44] further develop it to involve with feature derived from headers of Portable Executable files, which achieves quite remarkable accuracy and recall rate under cases of fusing data.

To detect attack indicators in advance, Saxe and Berlin [19] propose eXpose neural network, where their network takes the original short strings as input and extracts features to classify with character-level embeddings. It is noted that their original inputs are a wide and complicated range for algorithms to deal with. Owing to the self-extracted feature design, eXpose is superior to baseline methods based on

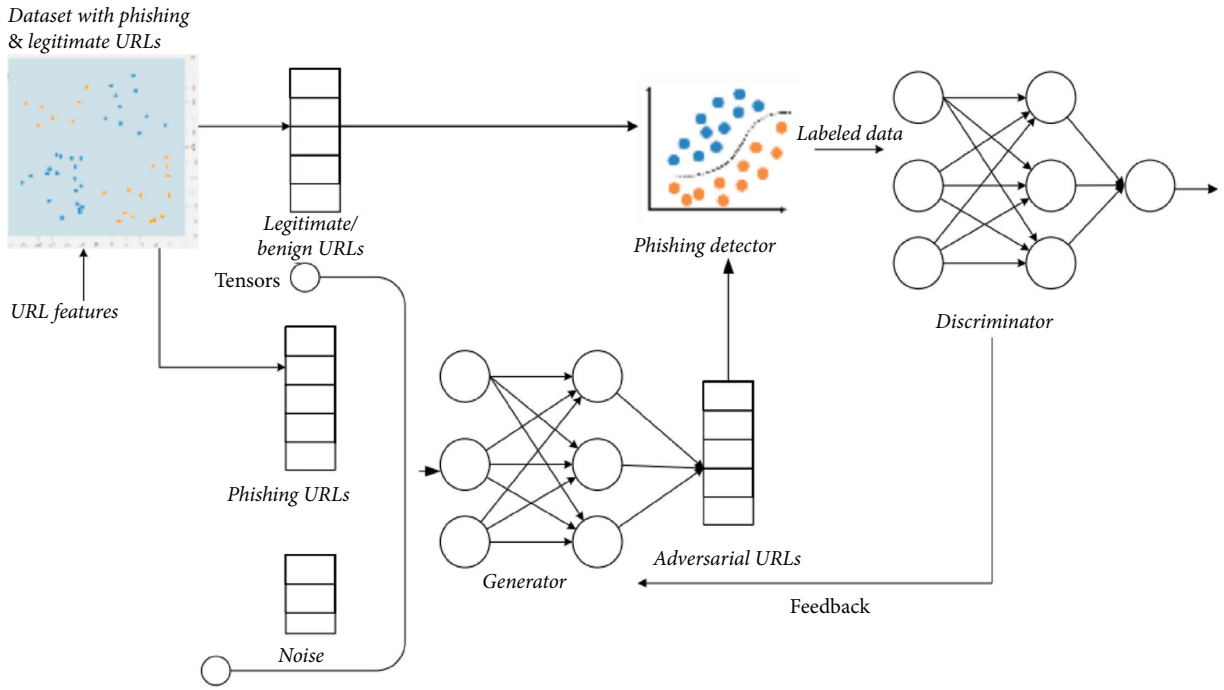

Figure 4: Overview of steps for the GAN model proposed by AlEroud and Karabatis [39].

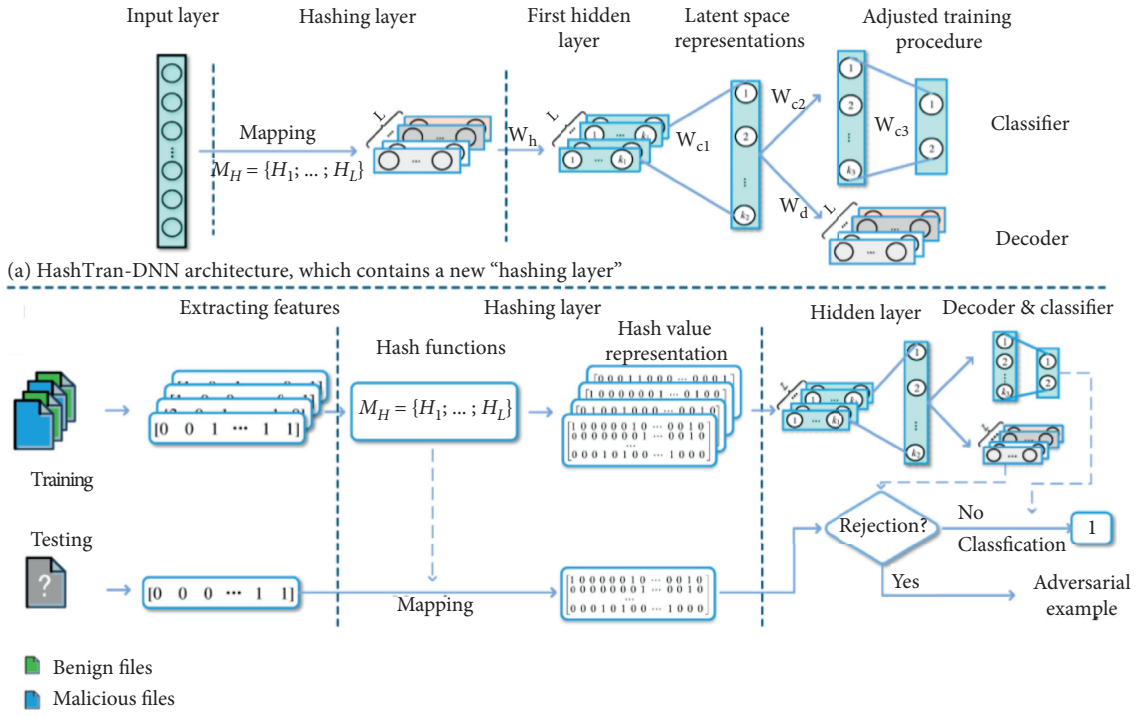

Figure 5: Architecture design of HashTran-DNN model proposed by Li et al. [41].

manual feature extraction. However, it achieves a decrease in false alarm rate compared with these baselines, which proves automatically feature extraction process in CNN is not robust and reasonable enough with introducing extra or even noise information from original inputs.

Malicious web shell detection is an important means to protect network security. Aiming at analysis of HTTP requests, Zhang et al. [46] propose a word2vec representing and CNN-based malicious detection approach, which is the first attempt to combine "word2vec" and CNN in malicious detection domain. Specifically, they first introduce the "word2vec" tool to represent each word obtained from HTTP by features. Then, they represent the web request as a fixed-size matrix by concatenating features. Finally, they

build up the shell classification model based on CNN structure. Several groups of experiment are carried out, and the proposed method performs the best when comparing with relevant classical classifiers.

To achieve robust performance in attack detection with CNN structure, an end-to-end encrypted traffic classification method based on one-dimensional CNN is presented by Wang et al. [45], in which feature extraction, selection, and classifier are integrated into an end-to-end framework. We show its detailed network design in Figure 6, where their proposed 1D-CNN as learning algorithm directly learns relationship between automatical extracted features and outputs with predicted labels in training phase. In their experiment, they adopt ISCX VPN-nonVPN traffic dataset for verification, where they achieve better performance than the latest methods in 11 of 12 evaluation measurements. Such promising results are remarkable, due to robust and informative traffic data representation and fine-tuning steps to improve model ability. Regarding network traffic data as two-dimensional image, a new traffic analysis approach based on CNN is further proposed by Wang et al. [47]. They test their algorithm on USTC-TRC2016 flow dataset to show average classification accuracy is as high as 99%.

To solve the diversity attack of wireless network traffic and improve the detection ability of malicious intrusion in wireless network, an intrusion detection method based on improved convolutional neural network is proposed by Yang and Wang [48], namely, ICNN-Based Wireless Network Intrusion Detection Model. Preprocess the network traffic data, and then model the data using CNN. CNN abstracts low-level intrusion traffic data into high-level features, automatically extracts sample features and optimizes network parameters through random gradient descent algorithm to converge the model. The results on the KDDTest + show that the detection accuracy is 8.82% and 0.51% higher than that of LeNet-5 and DBN, while the false positive rate is also lower. It also has a big advantage compared to the previous methods.

Low rate denial of service (LDOS) attacks reduce the performance of network services, and it is difficult to distinguish the attack behavior from the normal traffic. Thus, a new detection method of LDOS attack based on multifeature fusion and convolutional neural network (CNN) is proposed by Tang et al. [49]. They calculate features and fuse them into a feature map to describe the state of the network. The CNN model is used to distinguish and detect feature maps including LDOS attacks. Experiments are carried out on NS2 simulation platform and test-bed and results show that the proposed method can effectively detect LDOS attacks with accuracy of 97.1%.

3.2.3. Recurrent Neural Network Based Methods for Attack Detection. Since the output of DNN and CNN only considers the influence of the current input without considering information from the previous and future time, they could achieve significant performance on the classification or recognition tasks without time-varying characteristics. Involving time-dependent data, RNN is proposed as a special

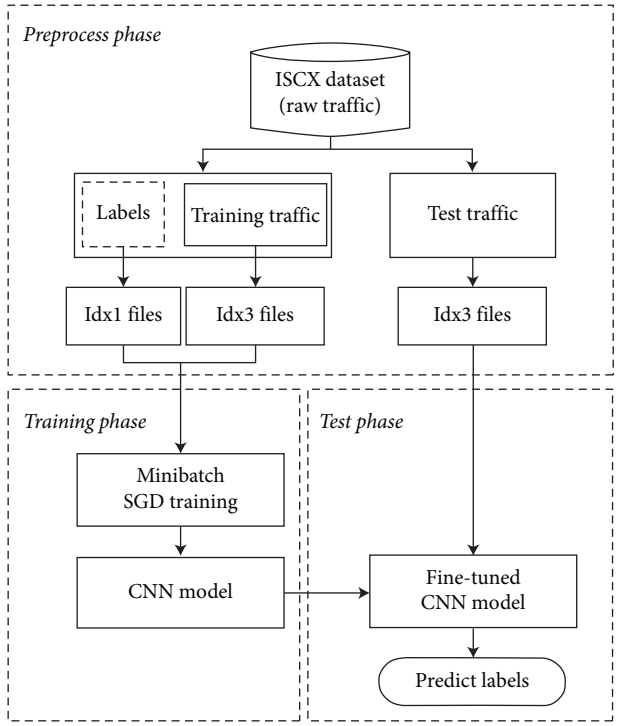

Figure 6: Workflow of the traffic analysis approach proposed by Wang et al. [45], which consists of three parts: preprocess, training, and testing phase.

category of neural network structures, which is designed with "memory" function to maintain previous content. In fact, such design feature coincides with the idea that "human cognition is based on the past experience and memory." RNN is thus good at dealing with time-series information. However, there are still some problems in structure design of RNN like gradient disappearance or gradient explosion, which leads failure to remember or model long-time dependence. Therefore, researchers develop LSTM and GRU with gates design and memory cell, which successfully keep long-time relationship unforgotten by passing through important components of information flow.

Early, Staudemeyer [50] proposes to consider time-series characteristics of known malicious behavior and network traffic, which may improve accuracy performance of attacks detection algorithms. To confirm this, they implement LSTM for intrusion detection based on excellent property of LSTM to model long-time dependant relationship. They design a four-memory blocks network, each of which contains two cells. The network is capable of keeping balance in both computational cost and detection performance. Their experimental results indicate that the proposed LSTM model is better than previously published methods since LSTM could learn to backtrack and correlate continuous connection records in a time-varying manner.

Later, Krishnan and Raajan [51] apply RNN to perform task of attack classification, where their anticipated model is constructed as a sawy self-erudition based Intrusion Detection System by RNN structure. During the experiments, their proposed intrusion detection system could filter

attacks, but fail to identify false positives. Comparing with the baseline methods, their proposed method has improved in measurements, such as classification accuracy and time-consuming.

Similarly, Yin et al. [52] explore utilizing RNN for intrusion detection named RNN-IDS, where they evaluate RNN-IDS with forms of binary classification and multiclass classification. In fact, RNN model has one-way information flow from the first units to the hidden, also from the previous hidden unit to the current one, where the hidden units could be regarded as storage units to store end-to-end and useful information for classification. They have tested whether the parameters, such as number of the neurons, have impact on the RNN-IDS using NSL-KDD dataset. When comparing with previous works such as ANN, random forest, and SVM, RNN-IDS has an advantage in classification performance with high accuracy.

Since LSTM solves the long-term dependency problem and overcomes the vanishing gradient drop during training, Kim et al. [54] apply LSTM architecture for intrusion detection, where the size of hidden layer and the learning rate are settled as 80 and 0.01 after experiments. Comparing with Staudemeyer [50], the constructed LSTM model has a higher false detection rate when training with the KDDCup 99 dataset. Following the trend of applying LSTM on attack detection, Le et al. [55] build a LSTM classifier to detect intrusion as well. They aim is to find the most suitable optimizer for gradient descent optimization of LSTM, where they compare six widely used optimization methods, that is, Adagrad, Adadelta, RMSprop, Adam, Adamax, and Nadam, and find the most effective one is LSTM with Nadam optimizer.

To reduce high false alarm rate achieved by the former methods, a system-call analysis method is proposed by Kim et al. [53], which is developed for anomaly-based host intrusion detection system. As shown in Figure 7, their method consists of two modules: the front-end module, that is, system call language models, which is used to model time-varying characteristics of system calls with LSTM structure in various environments, and the back-end module which is used to predict exceptions based on information passing from the front-end module by a set of ensemble and threshold-based classifiers.

GRU is a variant of LSTM, in which softmax function is used as the final output layer. Moreover, GRU uses cross-entropy function to calculate its losses. Based on GRU structure, Agarap [56] proposes a novel network for binary classification in the attack detection field, which regards a total of 21 features as model inputs. Specifically, linear support vector machine (SVM) is introduced to replace softmax function of the proposed GRU model, which could achieve relatively better effects than the traditional GRU-softmax network on public datasets, due to fast convergence and better ability in classification.

3.3. Other Deep Learning Methods for Attack Detection. In this subsection, we aim to emphasize on the hybrid category of methods on attack detection, which are designed with the idea of integrating advantages of different deep learning structures.

Early in 2015, Li et al. [57] apply a AE and DNN based hybrid deep learning method for malicious code detection. Specifically, they adopt AE to reduce dimensions of original data and focus on the main and important features. Afterward, they use a DBN-based learning model to do the detection of malicious code, which consists of multilayer RBM and a layer of BPNN. Defining each layer of RBM as unsupervised trained and BP as supervised trained, their optimal hybrid model is finally obtained by fine-tuning the whole network. Experiments show that detection accuracy of their hybrid network is higher than other previous DBN-based networks.

Later in 2017, Ludwig [58] employs an ensemble network to classify various types of attacks. In fact, the neural network learning classifies targets with multiple classifiers and merges their results to form robust outputs. To distinguish between normal and abnormal behaviors, their proposed method fuses AE, BNN, DNN, and extreme learning machine for better performance. Their proposed ensemble method brings promising results, which achieve more accurate performance than utilizing single classifier for detection task.

Following the idea of fusing classifiers to obtain better results, Li et al. [59] propose an ensemble structure to enhance the robustness of neural networks for malware detection in 2018; the network is shown in Figure 8. More specifically, a group of neural networks are trained in the training stage and each classifier keeps its counter such as input conversion and semantic preservation. In the test stage, the labeled samples are determined by voting of different classifiers. Their proposed ensemble framework is applied to the challenge of AICS 2019 and has received a good performance in both accuracy and recall.

In order to detect network attacks effectively, Liu et al. [60] propose an end-to-end detection method in 2019. Based on the deep learning model, the author proposes two payload classification models: PL-CNN and PL-RNN. The model learns feature representation from the original payload without feature engineering and end-to-end detection. At the same time, they design a data preprocessing method, which can keep enough information while keeping efficiency. The accuracy of the proposed methods is 99.36% and 99.98%, respectively, when applied to DARPA 1998 dataset. The proposed methods support the use of network data flow for effective end-to-end attack detection, so as to solve the practical problem.

Most recently in 2019, Zhang et al. [61] do not design the characteristics of the flow but directly extract the original data information for analysis. At the same time of learning the temporal and spatial characteristics of flow, a new network intrusion detection model called deep network is proposed, which integrates the improved leNet-5 and LSTM neural network structure. The CICIDS2017 dataset and the CTU dataset are used to evaluate the performance of the network. The amount of traffic is large, and the type of attack is relatively new. The experimental results show that the performance of the network model is better than other

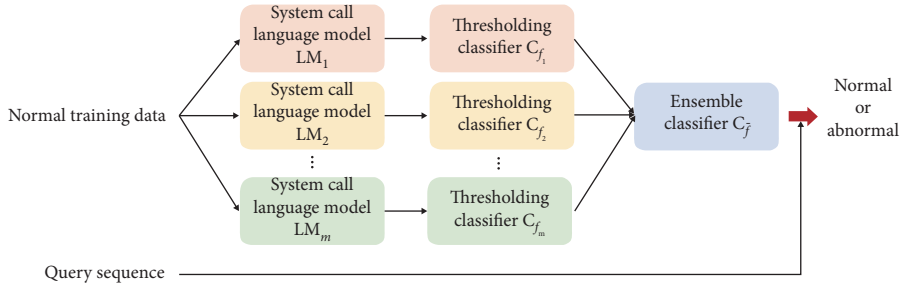

Figure 7: Structure design of Kim et al. [53] for intrusion detection system.

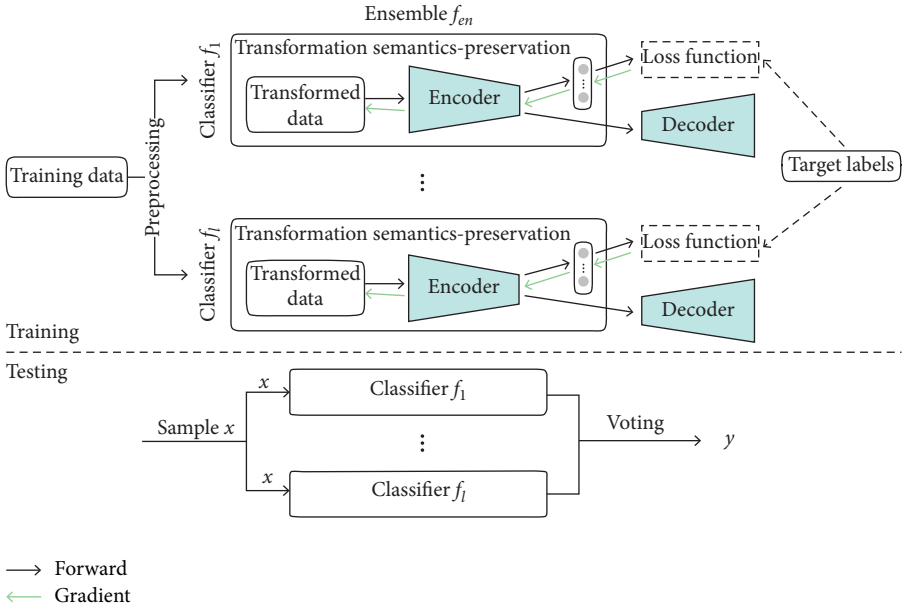

Figure 8: Workflow of the hybrid model proposed by [59].

network intrusion detection models, and it can achieve the best detection accuracy.

4. Comparisons and Analysis

4.1. Public Datasets. Many public datasets are popular to prove and compare efficiency and effectiveness among different attack detection methods. Among them, we list two famous benchmark datasets, that is, KDDCup 99 and NSL-KDD, which are widely used in the academic research to evaluate the ability to detect attacks.

4.1.1. KDDCup 99 Dataset. Despite the fact that there exist some drawbacks like containing a great deal of redundant training and testing data, KDDCup 99 dataset is famous in the field of cybersecurity. It includes both labeled training data and unlabeled test data, which correspond to seven and two weeks of data originated from DARPA′98 IDS evaluation program [62].

Five categories of labels are contained in the dataset which are normal, DoS, Probe, R2L and U2R, that is, short for DoS, Probe, R2L, and U2R, where normal refers to normal traffic instances, Dos is an attack in which the attacker tries to make the target machine stop providing service or resource access to system, Probe represents surveillance and probing, and R2L refers to the unauthorized access while there is an illegal access from the remote machine to local one and represents that there is an unauthorized access to local superuser privileges by local unprivileged user. In Table 1, we display 22 different attacks in training and test data, which could be categorized into these four attack types.

In KDDCup 99 dataset, each record has 41 features in total including basic features, content features, and traffic features as shown in Table 2, where the basic features are obtained from TCP/IP connections including basic characteristics of connection. The content features are extracted from data content, which can be used in the detection of U2R and R2L attacks, which are usually hidden in the packets data without abnormal appearance in single packet and normal connection. Meanwhile, traffic features refer to accumulated values in a time window with 100 connections. It is noted that 7 features and 34 features are symbolic and continuous in data type, respectively.

4.1.2. NSL-KDD Dataset. NSL-KDD is famous as a new development of KDDCup 99 dataset, which comes out to

Table 1: Category of 22 different attacks contained by KDDCup 99.

Class label	Attack name
DoS	back, land, neptune, pod, smurf, teardrop.
Probe	ipsweep, nmap, portsweep, satan.
R2L	ftp_write, guess_passwd, imap, multihop, phf, spy, warezclient, warezmaster.
U2R	buffer_overflow, loadmodule, perl, rootkit.

Table 2: Feature set for each instance in KDDCup 99 dataset.

No.	Features	Types
1	Duration	Continuous
2	protocol_type	Symbolic
3	Service	Symbolic
4	Flag	Symbolic
5	src_bytes	Continuous
6	dst_bytes	Continuous
7	Land	Symbolic
8	wrong_fragment	Continuous
9	Urgent	Continuous
10	Hot	Continuous
11	num_failed_logins	Continuous
12	logged_in	Symbolic
13	num_compromised	Continuous
14	root_shell	Continuous
15	su_attempted	Continuous
16	num_root	Continuous
17	num_file_creations	Continuous
18	num_shells	Continuous
19	num_access_files	Continuous
20	num_outbound_cmds	Continuous
21	is_hosts_login	Symbolic
22	is_guest_login	Symbolic
23	Count	Continuous
24	srv_count	Continuous
25	serror_rate	Continuous
26	srv_serror_rate	Continuous
27	rerror_rate	Continuous
28	srv_rerror_rate	Continuous
29	same_srv_rate	Continuous
30	diff_srv_rate	Continuous
31	drv_diff_host_rate	Continuous
32	dst_host_count	Continuous
33	dst_host_srv_count	Continuous
34	dst_host_same_srv_count	Continuous
35	dst_host_diff_srv_rate	Continuous
36	dst_host_same_src_port_count	Continuous
37	dst_host_srv_diff_host_rate	Continuous
38	dst_host_serror_rate	Continuous
39	dst_host_srv_serror_rate	Continuous
40	dst_host_serror_rate	Continuous
41	dst_host_srv_rerror_rate	Continuous

reduce shortcomings of the previous dataset. Specifically, it not only removes redundant data from the training and test data to achieve more accurate detection rate but also officially sets the number of records in both training and test data. Moreover, different difficulty level group has different number of records, which is inversely proportional to the percentage of that in the primary KDD dataset. Hence, evaluations and comparisons among different learning technologies become more effective and obvious.

NSL-KDD and KDDCup 99 dataset are similar in structure, where both of them are divided into four attack types as mentioned before. NSL-KDD dataset is divided into two parts: KDDTrain+ and KDDTest+, where we show the specific numbers corresponding to each attack type in Table 3. It is noted that there are 17 attack types in KDDTest+, which do not appear in KDDTrain+. This interesting setting makes NSL-KDD more challenging than KDDCup 99 dataset, which imitates real-life network environment with unknown attacks. We believe only these learning methods built on realistic theoretical basis, that is, analyzing inherent characteristics of attack behaviors, would achieve promising results on NSL-KDD.

4.2. Measurements. In this subsection, we describe 7 measurements including accuracy (ACC), precision (PR), true positive rate (TPR), recall (RE), false positive rate (FPR), true negative rate (TNR), and F1-score. Firstly, we define several items, where true positive (TP) and false negative (FN) refer to attack data correctly classified or not, respectively, and false positive (FP) and true negative (TN) are normal data which are classified as normal or attack, respectively. Afterwards, we define measurements as follows:

$$\begin{aligned}
\mathrm{ACC} &= \frac{(\mathrm{TP}+\mathrm{TN})}{(\mathrm{TP}+\mathrm{FN}+\mathrm{TN}+\mathrm{FP})},\\
\mathrm{PR} &= \frac{\mathrm{TP}}{(\mathrm{TP}+\mathrm{FP})},\\
\mathrm{RE} &= \frac{\mathrm{TP}}{(\mathrm{TP}+\mathrm{FN})},\\
\mathrm{FNR} &= \frac{\mathrm{FN}}{(\mathrm{TP}+\mathrm{FN})},\\
\mathrm{FPR} &= \frac{\mathrm{FP}}{(\mathrm{FP}+\mathrm{TN})},\\
\mathrm{TNR} &= \frac{\mathrm{TN}}{(\mathrm{TN}+\mathrm{FP})},\\
\mathrm{FS} &= \frac{(2*\mathrm{PR}*\mathrm{RE})}{(\mathrm{PR}+\mathrm{RE})},
\end{aligned} \tag{1}$$

where ACC shows the proportion of the amount of data that are correctly classified to whole data, PR calculates the proportion of the amount of attack data that are correctly classified to all attack data representing how many attacks predicted are actual attacks, TPR or RE shows the proportion of predicted attacks to all attacks, FNR estimates the percentage of the number of misclassified normal data to all

Table 3: Records distribution in training and test data [63].

Class	KDDTrain+	KDDTest+
Dos	45927	74588
Probe	11656	2421
R2L	995	2754
U2R	52	200

normal data, FPR called FAR measures the proportion of the benign events that are incorrectly classified as attack, TNR is recognized as the proportion of attack data that are correctly classified to the whole attack data, and F1-score is the weighted average of PR and RE and representing balance performance in both precision and recall.

4.3. Comparisons and Performance Analysis. In Table 4, we offer detailed statics on attack detection results achieved by various methods listed in Section 3, where most of the listed deep learning methods are designed to perform network intrusion detection and malware detection. Among quantity of measurements, we select accuracy, precision, F1-score, and FPR as evaluations, since most of the listed methods use these measurements for experiments. We must emphasize that there exist imbalances in performance comparisons since different authors adopt different datasets, measurements, and settings. However, Table 4 can still provide much information by roughly comparing different deep learning methods for attack detection.

From Table 4, we could notice that the mean performances of different categories of attack detection methods are variant. In the authors' opinion, DBN, LSTM, CNN, and AE achieve the detection performance in descending order. Meanwhile, hybrid methods are inconsistent, since their performances are highly related with ensemble classifiers. DBN is the highest in performance, due to its inherent property of multiple layers in dealing with quantity of unlabeled data. LSTM may achieve higher performance than CNN by involving temporal property for more precise modeling. AE may suffer from large unlabeled data without enough prior knowledge or enough layers to describe the complexity embedded.

Essentially, it is interesting to point out that RBMs and AEs are popular in intrusion detection because we can pretrain the RBMs and AEs with unlabeled data and fine-tune with only a small number of labeled data. Regarding ACC values achieve by listed methods as the first evaluation index due to its completeness, we can find the best performance achieved by attack detection methods on KDDCup 99 dataset, that is, 99.8% achieved by Kim et al. [53], is larger than that on NSL-KDD dataset, that is, 98.3% achieved by Javaid et al. [29], which proves that NSL-KDD dataset is much more difficult than KDDCup 99 dataset due to settings of unknown instances in testing dataset. Another interesting point is that all CNN-based methods abandon the usage of KDDCup 99 and NSL-KDD datasets since their small number of samples could not support showing distinguished power of CNN for generating feature descriptors with abundant information. Meanwhile, other deep learning methods, especially unsupervised learning methods, could bare the shortage of sufficient training samples.

We can observe that performance of AE-based methods is uneven, where most of the improved AE-based methods obviously perform better than traditional AE-based methods. This is due to the fact that the structure of AE might lose important information during compression process. Meanwhile, improved AE could better capture important and informative parts of input data with additional designs. Similarly, LSTM-based and GRU-based methods outperform RNN-based methods, due to their features in structure design of gates and memory cells. In fact, such intelligent designs bring advantage of capability of maintaining long-term information, thus better modeling long-time relationship.

Due to the large number of DBN- and RNN-based (e.g., LSTM and GRU) methods for attack detection proposed by researchers, we would like to regard DBN- and RNN-based methods as typical unsupervised and supervised algorithms, respectively, where we further compare them to show the advantages and disadvantages of both groups.

Essentially, RNN could remember information of the last several moments and then apply it in the calculation for the current unit, which introduces temporal information to help more accurate classification. However, RNN can be powerful structure with sufficient training instances, where attack data especially those unknown attacks are hard to be achieved. Meanwhile, DBN is capable of automatically discovering feature pattern from input data. Moreover, the unsupervised DBN network is less likely to be overfitting than those supervised methods due to its pretraining procedure, where DBN could learn inherent descriptions on abnormal behaviors or attacks by learning from unlabeled data. This feature of generated ability makes DBN, that is, a typical unsupervised learning method, fit with real environment of network security. Last but not least, DBN is easy to be trained, fast to be converged, and low in running time, due to less hidden layers compared with deep structures of CNN or so. Therefore, we think unsupervised learning methods could produce better classification results than supervised learning methods, especially when facing small, imbalanced, or redundant dataset.

5. Summary

Deep learning uses cascaded layers in a hierarchy structure to perform data processing, which results in significant results in domains of unsupervised feature learning and pattern recognition. Inspired by performance of deep learning methods, we believe deep learning is important for field of network security, so as to review the current deep learning methods for attack detection. We analyze recent methods, classify them according to different deep learning techniques, and compress the performance of the most representative methods.

Over the past few years, research on how to apply deep learning methods on attack detection has made a great progress. However, many problems still exist. Firstly, it is challenging to modify deep learning methods as real-time

Table 4: Quantitative evaluation of listed attack detection methods using different deep learning structures, where ID, MD, and TI represent network intrusion detection, malware detection, and traffic identification, respectively.

DL	Method	Usage	Dataset	ACC (%)	PR (%)	FPR (%)	FS
Convolutional AE	Yu et al. [26]	ID	CTU-UNB	—	98.44	—	0.980
Sparse AE	Javaid et al. [29]	ID	NSL-KDD	98.30	—	—	0.990
AE	Pamartzivanos et al. [30]	ID	KDDCup 99	77.99	80.00	—	—
SAE	Farahnakian and Heikkonen [28]	ID	KDDCup 99	94.71	94.53	0.42	—
AE	Shone et al. [17]	ID	NSL-KDD	89.22	92.97	10.78	0.910
Sparse AE	Shone et al. [17]	ID	KDDCup 99	97.85	99.99	2.15	0.980
AE	Aygun and Yavuz [64]	ID	NSL-KDD	93.62	91.39	—	0.938
Denoising AE	Aygun and Yavuz [64]	ID	NSL-KDD	94.35	94.26	—	0.940
Sparse AE	Gharic et al. [65]	ID	NSL-KDD	96.45	95.56	—	0.965
AE	Yousefi-Azar et al. [27]	ID, MD	NSL-KDD	83.34	—	—	—
DBN	Gao et al. [31]	ID	KDDCup 99	93.49	92.33	0.76	—
DBN	Ding et al. [32]	MD	Netflow	96.10	—	—	—
DBN	Qu et al. [66]	ID	NSL-KDD	95.25	—	—	—
DBN	Tan et al. [33]	ID	Netflow	97.60	—	0.90	—
DBN	Alom et al. [34]	ID	40% NSL-KDD	97.50	—	—	—
DBN	Zhao et al. [35]	ID	KDDCup 99	99.14	93.25	0.62	—
DBN	Alrawashdeh and Purdy [36]	ID	10% KDDCup 99	97.90	97.81	2.10	0.975
DNN	Tang et al. [40]	ID	NSL-KDD	91.70	83.00	—	—
DNN	Vinayakumar et al. [18]	ID, MD	KDDCup 99	93.00	99.00	0.95	
DNN	Wang et al. [42]	ID	KDDCup 99	95.45	—	—	—
CNN	Kolosnjaji et al. [43]	MD	Netflow	—	93.00	—	0.920
CNN	Saxe and Berlin [19]	MD	Netflow	92.00	—	0.10	—
CNN	Wang et al. [45]	ID	ISCX	—	97.30	—	0.960
CNN	Wang et al. [47]	TI	Netflow	99.41	—	—	—
CNN	Tang et al. [49]	ID	NS2 simulation	97.1	—	—	—
CNN	Yang and Wang [48]	ID	KDDCup 99	95.36	95.55	0.76	0.930
LSTM	Staudemyer [50]	ID	10% KDDCup 99	93.85	—	1.62	—
RNN	Krishnan and Raajan [51]	ID	KDDCup 99	77.55	84.60	—	0.730
RNN	Yin et al. [52]	ID	NSL-KDD	83.28	—	—	—
LSTM	Kim et al. [54]	ID	10% KDDCup 99	96.93	98.80	10.00	—
LSTM	Le et al. [55]	ID	KDDCup 99	97.54	98.95	9.98	—
LSTM	Kim et al. [53]	ID	KDDCup 99	99.80	—	5.50	—
GRU	Agarap [56]	ID	Netflow	84.15	—	—	—
Ensemble	Ludwig [58]	ID	NSL-KDD	92.50	93.00	0.92	—
AE, DBN	Li et al. [57]	ID	KDDCup 99	92.10	—	1.58	—
DCNN	Naseer et al. [67]	ID	NSL-KDD	85.00	—	—	0.980
PL-CNN	Liu et al. [60]	ID	DARPA1998	99.36	90.56	—	0.910
PL-RNN	Liu et al. [60]	ID	DARPA1998	99.98	99.98	—	0.990

classifiers for attack detection. In most of the previous works, they only reduce feature dimension for less computation cost during phase of feature extraction. Secondly, most of the deep learning techniques are appropriate for analysis of image and pattern recognition. Thus, how to conduct the classification of network traffic reasonably with deep learning techniques will be an interesting issue. Thirdly, with more data involving the experiments, the classification results will be better [68]. However, most of the attack detection problems are short of sufficient data. Therefore, combining supervised and unsupervised learning may provide better performance, which has been proved by many trials. Moreover, with the development of IoT [69], fog, cloud [70], and big data technologies, how to involve them to help improve effectiveness of attack detection methods using deep learning remains an open and interesting question. According to the above analysis, we hold a belief that this overview is a benefit for those who have ideas to improve the performance of attack detection in terms of accuracy; our review will provide guidance and dictionaries for further research in this field.

Data Availability

The data used to support the findings of this study were supplied by Dabao Wei under license and so cannot be made freely available. Requests for access to these data should be made to Yirui Wu (wuyirui@hhu.edu.cn).

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by National Key R&D Program of China under Grant 2018YFC0407901, the Fundamental Research Funds for the Central Universities under Grant

B200202177, the Natural Science Foundation of China under Grant 61702160, and the Natural Science Foundation of Jiangsu Province under Grant BK20170892.

References

[1] S. Aftergood, "Cybersecurity: the cold war online," *Nature*, vol. 547, no. 7661, pp. 30-31, 2017.

[2] M. A. Al-Garadi, A. Mohamed, A. Al-Ali, X. Du, and M. Guizani, "A survey of machine and deep learning methods for internet of things (iot) security," 2018, http://arxiv.org/abs/ 11023.

[3] A. Aleesa, B. Zaidan, A. Zaidan, and N. M. Sahar, *Review of Intrusion Detection Systems Based on Deep Learning Techniques: Coherent Taxonomy, Challenges, Motivations, Recommendations, Substantial Analysis and Future Directions. Neural Computing and Applications*, pp. 1–32, Springer, Berlin, Germany, 2019.

[4] G. Apruzzese, M. Colajanni, L. Ferretti, A. Guido, and M. Marchetti, "On the effectiveness of machine and deep learning for cyber security," in *Proceedings of 2018 10th International Conference on Cyber Conflict (CyCon)*, IEEE, Tallinn, Estonia, pp. 371–390, June 2018.

[5] D. Berman, A. Buczak, J. Chavis, and C. Corbett, "A survey of deep learning methods for cyber security," *Information*, vol. 10, no. 4, p. 122, 2019.

[6] M. A. Ferrag, L. Maglaras, S. Moschoyiannis, and H. Janicke, "Deep learning for cyber security intrusion detection: approaches, datasets, and comparative study," *Journal of Information Security and Applications*, vol. 50, p. 102419, 2020.

[7] C. S. Wickramasinghe, D. L. Marino, K. Amarasinghe, and M. Manic, "Generalization of deep learning for cyber-physical system security: a survey," in *Proceedings of IECON 2018-44th Annual Conference of the IEEE Industrial Electronics Society*, IEEE, Washington, DC, USA, pp. 745–751, October 2018.

[8] Y. Xin, L. Kong, Z. Liu et al., "Machine learning and deep learning methods for cybersecurity," *IEEE Access*, vol. 6, pp. 35365–35381, 2018.

[9] X. Xu, C. He, Z. Xu, L. Qi, S. Wan, and M. Z. A. Bhuiyan, "Joint optimization of offloading utility and privacy for edge computing enabled iot," *IEEE Internet of Things Journal*, vol. 7, no. 4, pp. 2622–2629, 2020.

[10] X. Xu, Q. Liu, X. Zhang, J. Zhang, L. Qi, and W. Dou, "A blockchain-powered crowdsourcing method with privacy preservation in mobile environment," *IEEE Transactions on Computational Social Systems*, vol. 6, no. 6, pp. 1407–1419, 2019.

[11] X. Xu, X. Liu, Z. Xu, F. Dai, X. Zhang, and L. Qi, "Trust-oriented iot service placement for smart cities in edge computing," *IEEE Internet of Things Journal*, vol. 7, 2019.

[12] X. Xu, Y. Chen, X. Zhang, Q. Liu, X. Liu, and L. Qi, *A Blockchain-Based Computation Offloading Method for Edge Computing in 5g Networks*, John and Wiley, Hoboken, NJ, USA, 2019.

[13] C. Wang, Z. Chen, K. Shang, and H. Wu, "Label-removed generative adversarial networks incorporating with k-means," *Neurocomputing*, vol. 361, pp. 126–136, 2019.

[14] T. Meng, K. Wolter, H. Wu, and Q. Wang, "A secure and cost-efficient offloading policy for mobile cloud computing against timing attacks," *Pervasive and Mobile Computing*, vol. 45, pp. 4–18, 2018.

[15] X. Li and H. Wu, "Spatio-emporal representation with deepneural recurrent network in MIMO CSI feedback," CoRRabs/1908.07934, 2019.

[16] R. Vinayakumar, K. Soman, and P. Poornachandran, "Evaluating effectiveness of shallow and deep networks to intrusion detection system," in *Proceedings of 2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, IEEE, Udupi, India, pp. 1282–1289, September 2017.

[17] N. Shone, T. N. Ngoc, V. D. Phai, and Q. Shi, "A deep learning approach to network intrusion detection," *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 2, no. 1, pp. 41–50, 2018.

[18] R. Vinayakumar, M. Alazab, K. P. Soman, P. Poornachandran, A. Al-Nemrat, and S. Venkatraman, "Deep learning approach for intelligent intrusion detection system," *IEEE Access*, vol. 7, pp. 41525–41550, 2019.

[19] J. Saxe and K. Berlin, "expose: a character-level convolutional neural network with embeddings for detecting malicious urls, file paths and registry keys," 2017, http://arxiv.org/abs/ 1702.08568.

[20] R. Pascanu, J. W. Stokes, H. Sanossian, M. Marinescu, and A. Thomas, "Malware classification with recurrent networks," in *Proceedings of 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1916–1920, Queensland, Australia, April 2015.

[21] Z. Feng, C. Shuo, and W. Xiaochuan, "Classification for dga-based malicious domain names with deep learning architectures," in *Proceedings of 2017 Second International Conference on Applied Mathematics and Information Technology*, London, UK, January 2017.

[22] J. Woodbridge, H. S. Anderson, A. Ahuja, and D. Grant, "Predicting domain generation algorithms with long short-term memory networks," 2016, http://arxiv.org/abs/ 1611.00791.

[23] M. Z. Alom, T. M. Taha, C. Yakopcic et al., "The history began from alexnet: a comprehensive survey on deep learning approaches,"2018 pages, CoRR abs/1803.01164.

[24] E. Aminanto and K. Kim, "Deep learning in intrusion detection system: an overview," in *Proceedings of 2016 International Research Conference on Engineering and Technology (2016 IRCET)*, Higher Education Forum, Seoul, South Korea, January 2016.

[25] L. Deng, "A tutorial survey of architectures, algorithms, and applications for deep learning," *APSIPA Transactions on Signal and Information Processing*, vol. 3, 2014.

[26] Y. Yu, J. Long, and Z. Cai, "Network intrusion detection through stacking dilated convolutional autoencoders," *Security and Communication Networks*, vol. 2017, Article ID 4184196, 10 pages, 2017.

[27] M. Yousefi-Azar, V. Varadharajan, L. Hamey, and U. Tupakula, "Autoencoder-based feature learning for cyber security applications," in *Proceedings of 2017 International Joint Conference on Neural Networks (IJCNN)*, IEEE, San Diego, CA, USA, pp. 3854–3861, June 2017.

[28] F. Farahnakian and J. Heikkonen, "A deep auto-encoder based approach for intrusion detection system," in *Proceedings of 2018 20th International Conference on Advanced Communication Technology (ICACT)*, IEEE, Chuncheon, South Korea, pp. 178–183, July 2018.

[29] A. Javaid, Q. Niyaz, W. Sun, and M. Alam, "A deep learning approach for network intrusion detection system," in *Proceedings of the 9th EAI International Conference on Bio-Inspired Information and Communications Technologies (formerly BIONETICS)*, pp. 21–26, New York, NY, USA, December 2016.

[30] D. Papamartzivanos, F. Gomez Marmol, and G. Kambourakis, *Introducing Deep Learning Self-Adaptive Misuse Network*

Intrusion Detection Systems, IEEE Access, Piscataway, NJ, USA, 2019.

[31] N. Gao, L. Gao, Q. Gao, and H. Wang, "An intrusion detection model based on deep belief networks," in *Proceedings of 2014 Second International Conference on Advanced Cloud and Big Data*, IEEE, Huangshan, China, pp. 247–252, November 2014.

[32] Y. Ding, S. Chen, and J. Xu, "Application of deep belief networks for opcode based malware detection," in *Proceedings of 2016 International Joint Conference on Neural Networks (IJCNN)*, pp. 3901–3908, Vancouver, British, July 2016.

[33] Q. . s. Tan, W. Huang, and Q. Li, "An intrusion detection method based on dbn in ad hoc networks," in *Proceedings of Wireless Communication and Sensor Network: International Conference on Wireless Communication and Sensor Network (WCSN*, World Scientific, Wuhan, China, pp. 477–485, December 2015.

[34] M. Z. Alom, V. Bontupalli, and T. M. Taha, "Intrusion detection using deep belief networks," in *Proceedings of 2015 National Aerospace and Electronics Conference (NAECON)*, pp. 339–344, Dayton, OH, USA, June 2015.

[35] G. Zhao, C. Zhang, and L. Zheng, "Intrusion detection using deep belief network and probabilistic neural network," in *Proceedings of 2017 IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC)*, Taipei, Taiwan, December 2017.

[36] K. Alrawashdeh and C. Purdy, "Toward an online anomaly intrusion detection system based on deep learning," in *Proceedings of 2016 15th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pp. 195–200, Anaheim, CA, USA, December 2016.

[37] H. Zhang, Y. Li, Z. Lv, A. K. Sangaiah, and T. Huang, "A real-time and ubiquitous network attack detection based on deep belief network and support vector machine," *IEEE/CAA Journal of Automatica Sinica*, vol. 7, no. 3, pp. 790–799, 2020.

[38] T. Erpek, Y. E. Sagduyu, and Y. Shi, "Deep learning for launching and mitigating wireless jamming attacks," *IEEE Transactions on Cognitive Communications and Networking*, vol. 5, no. 1, pp. 2–14, 2018.

[39] A. AlEroud and G. Karabatis, "Bypassing detection of url-based phishing attacks using generative adversarial deep neural networks," in *Proceedings of the Sixth International Workshop on Security and Privacy Analytics*, pp. 53–60, New Orleans, LA, USA, March 2020.

[40] T. A. Tang, L. Mhamdi, D. McLernon, S. A. R. Zaidi, and M. Ghogho, "Deep learning approach for network intrusion detection in software defined networking," in *Proceedings of 2016 International Conference on Wireless Networks and Mobile Communications (WINCOM)*, IEEE, Reims, France, pp. 258–263, October 2016.

[41] D. Li, R. Baral, T. Li, H. Wang, Q. Li, and S. Xu, "Hashtran-dnn: a framework for enhancing robustness of deep neural networks against adversarial malware samples," 2018, http://arxiv.org/abs/ 1809.06498.

[42] W. Peng, X. Kong, G. Peng, X. Li, and Z. Wang, "Network intrusion detection based on deep learning," in *Proceedings of 2019 International Conference on Communications, Information System and Computer Engineering (CISCE)*, pp. 431–435, Haikou, China, July 2019.

[43] B. Kolosnjaji, A. Zarras, G. Webster, and C. Eckert, "Deep learning for classification of malware system call sequences," in *Proceedings of Australasian Joint Conference on Artificial Intelligence*, pp. 137–149, Springer, Hobart, Australia, December 2016.

[44] B. Kolosnjaji, G. Eraisha, G. Webster, A. Zarras, and C. Eckert, "Empowering convolutional networks for malware classification and analysis," in *Proceedings of 2017 International Joint Conference on Neural Networks (IJCNN)*, pp. 3838–3845, San Diego, CA, USA, June 2017.

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

[46] M. Zhang, B. Xu, S. Bai, S. Lu, and Z. Lin, "A deep learning method to detect web attacks using a specially designed CNN," in *Proceedings of 24th International Conference on Neural Information Processing*, pp. 828–836, Guangzhou, China, November 2017.

[47] W. Wang, M. Zhu, X. Zeng, X. Ye, and Y. Sheng, "Malware traffic classification using convolutional neural network for representation learning," in *Proceedings of 2017 International Conference on Information Networking*, ICOIN), Da Nang, Vietnam, January 2017.

[48] H. Yang and F. Wang, "Wireless network intrusion detection based on improved convolutional neural network," *IEEE Access*, vol. 7, pp. 64366–64374, 2019.

[49] D. Tang, L. Tang, W. Shi, S. Zhan, and Q. Yang, *Mf-cnn: A New Approach for Ldos Attack Detection Based on Multi-Feature Fusion and Cnn. Mobile Networks and Applications*, pp. 1–18, Springer, Berlin, Germany, 2020.

[50] R. C. Staudemeyer, "Applying long short-term memory recurrent neural networks to intrusion detection," *South African Computer Journal*, vol. 56, no. 1, pp. 136–154, 2015.

[51] R. B. Krishnan and N. Raajan, "An intellectual intrusion detection system model for attacks classification using rnn," *International Journal of Pharmaceutical Technology and Biotechnology*, vol. 8, no. 4, pp. 23157–23164, 2016.

[52] C. Yin, Y. Zhu, J. Fei, and X. He, "A deep learning approach for intrusion detection using recurrent neural networks," *IEEE Access*, vol. 5, pp. 21954–21961, 2017.

[53] G. Kim, H. Yi, J. Lee, Y. Paek, and S. Yoon, "Lstm-based system-call language modeling and robust ensemble method for designing host-based intrusion detection systems," 2016, http://arxiv.org/abs/ 1611.01726.

[54] J. Kim, J. Kim, H. L. T. Thu, and H. Kim, "Long short term memory recurrent neural network classifier for intrusion detection," in *Proceedings of 2016 International Conference on Platform Technology and Service (PlatCon)*, pp. 1–5, Jeju, Korea, February 2016.

[55] T. Le, J. Kim, and H. Kim, "An effective intrusion detection classifier using long short-term memory with gradient descent optimization," in *Proceedings of 2017 International Conference on Platform Technology and Service (PlatCon)*, pp. 1–6, Jeju, Korea, February 2017.

[56] A. F. M. Agarap, "A neural network architecture combining gated recurrent unit (gru) and support vector machine (svm) for intrusion detection in network traffic data," in *Proceedings of the 2018 10th International Conference on Machine Learning and Computing*, ACM, Macau, China, February 2018.

[57] Y. Li, R. Ma, and R. Jiao, "A hybrid malicious code detection method based on deep learning," *International Journal of Security and Its Applications*, vol. 9, no. 5, pp. 205–216, 2015.

[58] S. A. Ludwig, "Intrusion detection of multiple attack classes using a deep neural net ensemble," in *Proceedings of 2017 IEEE Symposium Series on Computational Intelligence (SSCI)*, IEEE, Honolulu,HI, USA, November 2017.

[59] D. Li, Q. Li, Y. Ye, and S. Xu, "Enhancing robustness of deep neural networks against adversarial malware samples: principles, framework, and aics'2019 challenge," 2018, http://arxiv.org/abs/ 1812.08108.

[60] H. Liu, B. Lang, M. Liu, and H. Yan, "Cnn and rnn based payload classification methods for attack detection," *Knowledge-Based Systems*, vol. 163, pp. 332–341, 2019.

[61] Y. Zhang, X. Chen, L. Jin, X. Wang, and D. Guo, "Network intrusion detection: based on deep hierarchical network and original flow data," *IEEE Access*, vol. 7, pp. 37004–37016, 2019.

[62] R. Lippmann, J. W. Haines, D. J. Fried, J. Korba, and K. Das, "The 1999 darpa off-line intrusion detection evaluation," *Computer Networks*, vol. 34, no. 4, pp. 579–595, 2000.

[63] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, "A detailed analysis of the kdd cup 99 data set," in *Proceedings of 2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications*, IEEE, Ottawa, Canada, pp. 1–6, July 2009.

[64] R. C. Aygun and A. G. Yavuz, "Network anomaly detection with stochastically improved autoencoder based models," in *Proceedings of 2017 IEEE 4th International Conference on Cyber Security and Cloud Computing (CSCloud)*, pp. 193–198, New York, NY, USA, June 2017.

[65] M. Gharib, B. Mohammadi, S. H. Dastgerdi, and M. Sabokrou, "Autoids: auto-encoder based method for intrusion detection system," 2019, http://arxiv.org/abs/ 1911.03306.

[66] F. Qu, J. Zhang, Z. Shao, and S. Qi, "An intrusion detection model based on deep belief network," in *Proceedings of the 2017 VI International Conference on Network, Communication and Computing*, pp. 97–101, Kunming, China, December 2017.

[67] S. Naseer, Y. Saleem, S. Khalid et al., "Enhanced network anomaly detection based on deep neural networks," *IEEE Access*, vol. 6, pp. 48231–48246, 2018.

[68] N. Jones, "Computer science: the learning machines," *Nature*, vol. 505, no. 7482, pp. 146–148, 2014.

[69] X. Xu, X. Zhang, H. Gao, Y. Xue, L. Qi, and W. Dou, "Become: blockchain-enabled computation offloading for iot in mobile edge computing," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 6, pp. 4187–4195, 2020.

[70] X. Xu, R. Mo, F. Dai, W. Lin, S. Wan, and W. Dou, "Dynamic resource provisioning with fault tolerance for data-intensive meteorological workflows in cloud," *IEEE Transactions on Industrial Informatics*, vol. 16, 2019.